



**HAL**  
open science

# Optimization of ITS-G5 network resource management to support C-ITS services

Abdenmour Rachedi

► **To cite this version:**

Abdenmour Rachedi. Optimization of ITS-G5 network resource management to support C-ITS services. Artificial Intelligence [cs.AI]. Université de Bordeaux, 2024. English. NNT : 2024BORD0070 . tel-04589598

**HAL Id: tel-04589598**

**<https://theses.hal.science/tel-04589598>**

Submitted on 27 May 2024

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

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

THÈSE PRÉSENTÉE

POUR OBTENIR LE GRADE DE

**DOCTEUR DE  
L'UNIVERSITÉ DE BORDEAUX**

ÉCOLE DOCTORALE  
DE MATHÉMATIQUES ET D'INFORMATIQUE  
SPECIALITÉ : INFORMATIQUE

PRÉPARÉE À : BORDEAUX INP

Par Abdenmour RACHEDI

**Optimization of ITS-G5 network resource management to  
support C-ITS services**

**Optimisation de la gestion des ressources des réseaux ITS-G5  
pour le support des services C-ITS**

Sous la direction de : Toufik AHMED, Professeurs à Bordeaux INP  
Et la codirection de : Mohamed MOSBAH, Professeurs à Bordeaux INP

Soutenue le : 18/04/2024

Membres du jury :

Gerard Chalhoub	Professeur, Université Clermont Auvergne	Rapporteur
Thierry Gayroud	Professeur, Université Toulouse III	Rapporteur
Houda Labiod	Professeur, Télécom Paristech	Examinatrice
Toufik AHMED	Professeur, Bordeaux INP	Directeur de thèse
Mohamed MOSBAH	Professeur, Bordeaux INP	Codirecteur de thèse

**Title:** Optimizations of ITS-G5 network resource management to support C-ITS services

## Abstract

This thesis unfolds in the dynamic context of Cooperative Intelligent Transport Systems (C-ITS) and Vehicle-to-Everything (V2X) communications, with a particular focus on the integration of emerging technologies such as Artificial Intelligence (AI), Network Slicing, and Multi-access Edge Computing (MEC). These revolutionary technologies are reshaping the way vehicular networks manage traffic safety and efficiency while presenting unique challenges.

The first major challenge addressed in this thesis is the degradation of communication channel quality in congested V2X networks. This common situation in dense traffic environments negatively impacts vehicular communication performance, thus hindering the efficiency of C-ITS. The second challenge is to ensure ultra-low end-to-end (E2E) latency in these congested networks, particularly for services and user groups requiring high priority. This need is especially crucial in scenarios where vehicles, such as emergency services, rely on rapid and reliable communication. The third significant issue tackled is service migration in MEC-enabled vehicular networks, an essential aspect to ensure service continuity in highly mobile environments. The mobile nature of vehicular networks, combined with the limited coverage of edge servers, poses significant challenges in maintaining QoS and minimizing service interruptions.

To address these challenges, the thesis proposes several innovative solutions. A proactive approach for Decentralized Congestion Control (DCC) was developed using Long Short-Term Memory (LSTM) recurrent neural networks. This technique aims to optimize channel performance by forecasting the Channel Busy Ratio (CBR), thus improving network stability and ensuring fair resource allocation. Simulations demonstrated the effectiveness of proactive DCC algorithms, showing faster convergence and better resource management.

Next, we address the innovative aspects of network slicing in ITS-G5 vehicular communications. The second contribution proposes an ITS-G5 RAN slicing architecture, aiming to create slices with varied priorities for efficient and secure traffic, while ensuring isolation and prioritization between slices. This approach aims to maintain performance and security for each slice, even in the presence of conflicting services. In the third contribution, we develop an end-to-end network slicing architecture, aiming to improve latency for specific user groups, particularly in congested areas. Simulations confirmed the effectiveness of these architectures in traffic flow management and latency reduction for high-priority services, demonstrating the importance of these approaches in advancing intelligent and efficient vehicular networks.

Finally, to address service migration in MEC vehicular networks, we formulated the problem as a Markov Decision Process (MDP) and developed an adaptive migration strategy using Deep Reinforcement Learning (DRL), specifically Deep Q Networks (DQN) and Double Deep Q-network (DDQN) approaches. This strategy aims to balance migration costs and latency. Simulation results showed that the DDQN method excels in managing migration costs while maintaining optimal QoS, particularly for latency-sensitive services, and offers an optimal balance for high-priority services.

These contributions, combining technological advances and innovative analytical approaches, provide robust solutions to current and emerging challenges in cooperative intelligent transport systems, paving the way for significant improvements in road safety, traffic efficiency, and user experience in the field of smart mobility.

**Keywords:** C-ITS, V2X, Artificial Intelligence, Network Slicing, MEC, ITS-G5, DCC, Service Migration.

**Titre :** Optimisation de la gestion des ressources des réseaux ITS-G5 pour le support des services C-ITS

## Résumé

Cette thèse s'intéresse à la gestion des ressources dans les Systèmes de Transport Intelligents Coopératifs (C-ITS) utilisant les communications V2X (vehicle-to-everything) ou Véhicule-à-Tout. Nos contributions tirent profit des technologies émergentes telles que l'Intelligence Artificielle (IA), le tranchage de réseau (Slicing) et le Multi-access Edge Computing (MEC). Ces technologies promettent de révolutionner la manière dont les réseaux véhiculaires gèrent la sécurité et l'efficacité du trafic routier et permettent de résoudre certains défis liés à l'optimisation des ressources réseaux.

Le premier défi majeur abordé dans cette thèse concerne la dégradation de la qualité des canaux de communication dans les réseaux V2X congestionnés. Cette situation, fréquente dans les environnements de trafic dense, affecte négativement les performances des communications véhiculaires, entravant ainsi l'efficacité des services C-ITS. Le deuxième défi est de garantir une latence ultra-faible de bout-en-bout dans ces réseaux encombrés, en particulier pour des services et des groupes d'utilisateurs nécessitant une priorité élevée. Ce besoin est particulièrement crucial dans les scénarios où des véhicules, tels que ceux des services d'urgence, exigent une communication fiable et à faible latence. La troisième problématique traitée est la migration de services dans les réseaux véhiculaires équipés de MEC, un aspect essentiel pour assurer la continuité des services dans des environnements à mobilité élevée. La nature mobile des réseaux véhiculaires, combinée à une couverture limitée des serveurs Edge, pose des défis significatifs en termes de maintien d'un bon niveau de Qualité de Service (QoS) et de minimisation des interruptions de service.

Pour relever ces défis, la thèse propose trois contributions majeures. Premièrement, une approche proactive pour le Contrôle de Congestion Décentralisé (DCC) a été développée en utilisant des réseaux neuronaux récurrents avec Long Short-Term Memory (LSTM). Cette technique vise à optimiser les performances du canal en prévoyant en amont le Taux d'Occupation du Canal (CBR) pour améliorer la stabilité du réseau tout en garantissant une allocation équitable des ressources. Les simulations ont démontré l'efficacité des algorithmes DCC proactifs, montrant une convergence plus rapide et une meilleure gestion de l'allocation des ressources.

Ensuite, nous avons abordé le tranchage de réseau dans les communications véhiculaires utilisant la technologies ITS-G5. Pour cela, nous avons proposé une architecture de tranchage de la partie RAN ITS-G5 (Radio Access Network), visant à créer des tranches réseaux avec des

priorités variées, tout en garantissant l'isolation des performances et le respect de niveau de QoS entre les tranches. Cette approche vise à maintenir des niveaux de performances et de sécurité adéquats pour chaque tranche, même en présence de services d'exigences de QoS contradictoires. Au-delà de la partie RAN, nous avons élaboré une architecture de tranchage de réseau de bout-en-bout (RAN et Core Network), visant à garantir des latences faibles pour des groupes d'utilisateurs spécifiques, notamment dans les zones encombrées. Les simulations ont confirmé l'efficacité de cette architecture dans la gestion du flux de trafic et la réduction de la latence pour les services à haute priorité, démontrant l'importance de ces approches dans l'avancement des réseaux véhiculaires intelligents et efficaces.

La dernière contribution aborde la migration de services dans les réseaux véhiculaires équipés de MEC. Nous avons formulé ce problème de migration en tant que Processus Décisionnel Markovien (MDP) et avons développé une stratégie de migration adaptative en utilisant l'apprentissage par renforcement profond (DRL), notamment les approches Deep Q Networks (DQN) et Double Deep Q Networks (DDQN). Cette stratégie vise à trouver l'équilibre optimal entre la latence et le coût de migration. Les résultats de nos simulations ont montré que la méthode DDQN excelle dans la gestion des coûts de migration tout en maintenant un niveau de QoS optimal, en particulier pour les services sensibles à la latence, et offre un équilibre optimal pour les services à haute priorité.

Nos contributions fournissent des solutions robustes aux défis actuels et émergents dans les systèmes de transport intelligents coopératifs et ouvrent la voie à des améliorations significatives en termes de sécurité routière, d'efficacité du trafic et d'expérience utilisateur dans le domaine de la mobilité intelligente.

**Mots clés :** C-ITS, V2X, Intelligence Artificielle, tranchage de réseau, MEC, ITS-G5, DCC, Migration de Services.

## Résumé étendu en français

Dans le paysage en évolution rapide des communications véhiculaires, les Systèmes de Transport Intelligents Coopératifs (C-ITS) se sont imposés comme une technologie clé pour améliorer la sécurité routière et la gestion du trafic. L'intégration des C-ITS dans les réseaux véhiculaires marque un pas important vers des systèmes de transport plus intelligents, sûrs et efficaces. Au cœur de cette avancée se trouve la communication Véhicule-à-Tout (V2X), une technologie fondamentale qui facilite l'échange d'informations entre les véhicules et leur environnement, y compris d'autres véhicules, piétons, infrastructures routières et systèmes de réseau.

La communication V2X, standardisée en Europe sous le nom d'ITS-G5, fonctionne sur les principes de la norme IEEE 802.11p et est spécifiquement conçue pour les Réseaux Ad hoc Véhiculaires (VANET). L'ITS-G5 joue un rôle crucial dans les C-ITS en fournissant le cadre pour divers types de stations de Système de Transport Intelligent (ITS) : véhicules, Unités de Bord de Route (RSU) et piétons, pour contribuer de manière collaborative à l'efficacité et à la sécurité du trafic. Elle permet une gamme de modes de communication, y compris véhicule-à-infrastructure (V2I), véhicule-à-véhicule (V2V) et véhicule-à-piéton (V2P), assurant une couverture et une connectivité complètes. Ce large spectre de modalités de communication est essentiel pour transmettre des informations critiques telles que les Messages de Sensibilisation Coopérative (CAM) et les Messages de Notification d'Événement Décentralisés (DENM), qui sont cruciaux pour la gestion du trafic en temps réel et les scénarios de réponse aux urgences.

La prolifération des applications V2X entraîne des exigences strictes en termes de Qualité de Service (QoS), en particulier en ce qui concerne la latence et la fiabilité. Ces exigences deviennent encore plus prononcées dans les zones à forte densité, où les réseaux véhiculaires sont souvent confrontés à des défis tels que la congestion de la bande passante et les interférences entre les liens sans fil. Pour relever ces défis, des solutions technologiques innovantes telles que l'Intelligence Artificielle (IA), le Tranchage de Réseau et le Multi-accès Edge Computing (MEC) peuvent être exploitées.

L'Intelligence Artificielle (IA) devient de plus en plus un pilier dans l'évolution des systèmes de communication véhiculaires. L'IA, avec ses vastes capacités, offre des avancées significatives dans la gestion et l'optimisation de la dynamique complexe des réseaux véhiculaires. Parmi les différentes branches de l'IA, l'Apprentissage par Renforcement Profond (DRL) et les réseaux Long Short-Term Memory (LSTM) pourraient être très efficaces dans ce domaine. Les réseaux LSTM contribuent de manière significative à l'analyse et à la prévision des données séquentielles, un aspect vital dans le contexte des communications véhiculaires où les modèles et tendances évoluent dans le temps. Cependant, le DRL est essentiel pour améliorer les processus décisionnels, assurant que les réseaux véhiculaires peuvent s'adapter et

fonctionner efficacement dans des environnements en rapide évolution. La synergie de ces technologies d'IA dans les systèmes C-ITS et V2X conduit à des réseaux véhiculaires plus intelligents, plus réactifs et plus efficaces, sous-tendant l'avancement des systèmes de transport véhiculaires.

Le tranchage de réseau, un nouveau paradigme dans le domaine des réseaux, est sur le point de révolutionner les réseaux véhiculaires en tant que technologie clé pour la 5G et au-delà. Il implique la création de réseaux logiques indépendants de bout-en-bout sur une infrastructure partagée, chacun étant conçu pour supporter des services ou applications spécifiques avec des Accords de Niveau de Service (SLA) prédéfinis. Le tranchage de réseau facilite la gestion et le contrôle des ressources réseau de manière plus efficace, en garantissant que chaque tranche réponde à ses exigences spécifiques de QoS. Cette capacité à séparer les ressources réseau pour différentes applications et groupes d'utilisateurs est particulièrement bénéfique pour gérer divers types de trafic et priorités au sein des réseaux ITS-G5.

Le Multi-access Edge Computing (MEC) émerge comme une technologie complémentaire, offrant des capacités de calcul robustes à la périphérie du réseau. En déployant des serveurs Edge à proximité des utilisateurs des réseaux véhiculaires, le MEC réduit considérablement la latence et améliore la puissance de calcul disponible pour les véhicules. Ceci est particulièrement crucial pour des applications nécessitant de hautes ressources computationnelles, telles que la détection d'objets et l'analyse de flux vidéo, permettant ainsi leur exécution efficace au sein du réseau véhiculaire. Le rôle du MEC dans la minimisation de la latence des services et le soutien de tâches computationnellement intensives est essentiel pour maintenir la performance continue et fiable des réseaux véhiculaires.

Ensemble, ces technologies - l'IA, le tranchage de réseau et le MEC – (et autres) constituent la base des communications modernes C-ITS et V2X. Leur intégration ne se contente pas de répondre aux défis actuels des réseaux véhiculaires, mais ouvre également la voie à de futures innovations dans les systèmes de transport intelligents. Alors que les réseaux véhiculaires continuent d'évoluer et de devenir plus complexes, la synergie de ces technologies avancées sera clé pour réaliser le plein potentiel des C-ITS, en améliorant la sécurité routière, l'efficacité du trafic et l'expérience utilisateur globale à l'ère de la mobilité intelligente.

Cette thèse vise à d'atteindre trois objectifs clés : premièrement, améliorer les communications V2X grâce à une allocation de ressources optimisée et à une équité accrue ; deuxièmement, garantir une QoS de bout-en-bout pour une variété d'applications et de services V2X avancés ; et troisièmement, fournir une solution adaptée aux services sensibles à la latence dans des environnements hautement mobiles tout en optimisant les coûts. Cette approche globale aborde les aspects cruciaux des systèmes de communication véhiculaire, cherchant un équilibre entre performance, qualité et rentabilité. Ce qui soulève les contributions suivantes.



## **1. Proactive C-ITS Decentralized Congestion Control Using LSTM**

Dans le domaine des réseaux véhiculaires, notamment dans le cadre de la technologie ITS-G5 européenne, la gestion de la densité des véhicules et de l'utilisation des canaux présente des défis significatifs. Ces réseaux, caractérisés par leurs environnements dynamiques et complexes, sont souvent confrontés à des problèmes de congestion des canaux et de dégradation de la qualité, principalement en raison du volume élevé de trafic généré par les véhicules, en particulier les Messages de Sensibilisation Coopérative (CAM). Pour relever ces défis, des techniques de Contrôle de Congestion Décentralisé (DCC) telles que le Contrôle de Taux de Transmission réactif (TRC) et l'algorithme Dual  $\alpha$  adaptatif ont été proposés.

Cependant, ces méthodes existantes présentent des limites dans les conditions de réseau véhiculaire en rapide évolution. Ce premier travail introduit une nouvelle approche du DCC, en utilisant une technique proactive qui utilise un Réseau Neuronal Récurrent (RNN) Long Short-Term Memory (LSTM). L'objectif principal est d'optimiser les performances du canal en prévoyant le Taux d'Occupation du Canal (CBR) à l'aide de valeurs CBR lissées calculées par un agent LSTM. Ces prévisions sont ensuite intégrées dans les algorithmes DCC pour permettre une convergence plus rapide de la charge du canal, améliorant ainsi la stabilité du réseau, limitant la congestion du canal et garantissant une allocation équitable des ressources.

Ce travail a consisté à réaliser des simulations pour démontrer l'efficacité des algorithmes DCC proactifs proposés, en comparaison avec le TRC réactif et le Dual  $\alpha$  adaptatif. Les principaux paramètres évalués étaient le CBR moyen et la vitesse de convergence du CBR, qui sont indicatifs de la stabilité du canal de transmission et du niveau d'équité atteint parmi les véhicules. Les simulations ont montré que l'algorithme Dual  $\alpha$  proactif basé sur LSTM, en particulier, a atteint une meilleure et plus rapide convergence dans des conditions stables par rapport à l'algorithme Dual  $\alpha$  hérité. De plus, l'utilisation d'algorithmes DCC proactifs n'a pas eu d'impact négatif sur la fiabilité ou la qualité de transmission du canal, comme en témoigne la légère augmentation du Taux de Livraison de Paquets (PDR) dans tous les environnements de simulation.

L'approche proactive, basée sur les prédictions LSTM, a effectivement réduit l'épuisement du canal et accéléré la transition d'états de canal instables à stables, garantissant ainsi une convergence plus rapide du canal, une utilisation optimale des ressources et une équité accrue tant localement que globalement parmi les véhicules. Ces résultats confirment le potentiel d'utilisation d'algorithmes IA avancés pour le DCC, suggérant des orientations futures pour remplacer entièrement les mécanismes DCC traditionnels par des solutions basées sur l'IA.

## **2. A RAN Slicing Architecture for ITS-G5 C-ITS**

Dans cette deuxième contribution, on aborde les besoins évolutifs des C-ITS face aux transformations technologiques dans le marché automobile. L'accent principal est mis sur

l'assurance d'une ultra-faible latence et d'une ultra-haute fiabilité dans des conditions de haute mobilité et de densité, en utilisant la technologie européenne ITS-G5 pour les services V2X. Nous proposons une architecture de tranchage innovante du Réseau d'Accès Radio (RAN) ITS-G5, intégrant de nouveaux modules dans la pile de protocoles ITS. Cette architecture vise à créer plusieurs tranches RAN avec des priorités variées, garantissant un accès sécurisé, une limitation et régulation efficaces du trafic, une priorisation du trafic et une isolation entre les différentes tranches. L'objectif central est de maintenir la performance et la sécurité pour chaque tranche, même en présence de services ayant des exigences de performance conflictuelles, en s'assurant que l'amélioration des performances dans une tranche ne se fait pas au détriment d'une autre.

La mise en œuvre et la validation de l'architecture de tranchage RAN ITS-G5 ont impliqué diverses simulations et scénarios au sein de l'environnement de simulation Artery. Ces simulations ont été conçues pour mettre en évidence les capacités de priorisation du trafic, de Qualité de Service (QoS) et d'isolation du réseau. Plus précisément, les simulations se sont concentrées sur les performances de deux tranches conçues : une tranche de priorité plus élevée (HP) et une tranche de priorité inférieure (LP).

Les résultats de ces simulations ont confirmé l'efficacité de l'architecture dans la gestion du flux de trafic et le maintien d'un taux de messages fixe par station, validant ainsi les mécanismes de façonnage et de régulation du trafic. La tranche HP a constamment maintenu des latences moyennes inférieures à celles de la tranche LP à travers différentes densités de trafic et proportions de trafic HP. Ce résultat souligne la capacité de l'architecture à isoler et prioriser efficacement le trafic. De plus, les valeurs de latence médiane et au 95<sup>e</sup> percentile ont davantage démontré la performance supérieure de la tranche HP par rapport à la tranche LP, confirmant l'efficacité de l'architecture dans la priorisation du trafic et la différenciation de la QoS dans diverses conditions de trafic.

### **3. End-to-End Network Slicing for ITS-G5 Vehicular Communications**

L'objectif de ce travail est de faire progresser les C-ITS dans le contexte de ITS-G5. Les réseaux C-ITS rencontrent des défis significatifs dans les zones encombrées où la performance de la communication diminue considérablement. Ces réseaux nécessitent une performance de communication améliorée, en particulier dans des conditions de haute mobilité et densité, pour répondre aux exigences strictes de très faible latence et de très haute fiabilité. Cette contribution propose une architecture innovante de tranchage de réseau de bout-en-bout pour les communications véhiculaires ITS-G5. Cette architecture, en intégrant de nouveaux modules dans la pile de protocoles ITS, vise à créer plusieurs tranches de bout-en-bout avec des priorités variées pour assurer les exigences de qualité de service (QoS) pour différentes applications. Le mécanisme de tranchage se concentre sur la fourniture d'une latence de bout-en-bout améliorée

pour des types de trafic et des groupes d'utilisateurs spécifiques, en particulier pour le trafic à haute priorité dans les zones encombrées.

Les scénarios de simulation et les résultats présentés valident l'efficacité de l'architecture de tranchage proposée. L'objectif principal de l'approche de tranchage était d'améliorer la latence de bout-en-bout pour le trafic à haute priorité, en particulier pour les Messages de Sensibilisation Coopérative (CAM) dans les zones à forte densité de véhicules. L'architecture a défini deux tranches distinctes pour deux groupes d'utilisateurs : une Tranche à Haute Priorité (HPS) pour les CAM à haute priorité et une Tranche à Priorité Ordinaire (OPS) pour les CAM à priorité ordinaire. Les scénarios ont été conçus pour assurer la transmission de messages à haute priorité avec moins de temps d'attente et plus d'opportunités d'accès au canal de transmission.

Les résultats de ces simulations ont indiqué une amélioration notable de la latence de bout-en-bout pour la HPS par rapport à la OPS, avec des latences pour la HPS restant stables et significativement plus basses que celles de la OPS même dans des scénarios à haute densité. Ces résultats soulignent l'efficacité du mécanisme de tranchage pour prioriser le trafic à haute priorité et maintenir une latence plus faible, ce qui est particulièrement bénéfique pour des applications V2X avancées nécessitant des latences ultra-faibles telles que la conduite autonome et le peloton de véhicules.

#### **4. Dynamic Service Migration Empowered with Deep Q-Learning for C-ITS**

Dans une ère où le Multi-access Edge Computing (MEC) gagne en importance, notamment pour des tâches sensibles au temps et computationnellement intensives dans les réseaux véhiculaires, cette dernière contribution aborde un défi critique : les interruptions de service et la dégradation de la Qualité de Service (QoS) dues à la haute mobilité des véhicules et à la couverture limitée des serveurs Edge. Le potentiel du MEC dans les réseaux véhiculaires est immense, facilitant des applications avancées telles que la détection d'objets et l'analyse de flux vidéo. Cependant, ce potentiel est souvent entravé par la haute mobilité des véhicules, entraînant des topologies de communication dynamiques et des liens de communication peu fiables.

Pour relever ces défis, ce travail étudie le problème de la migration de services dans les réseaux véhiculaires dotés de MEC. L'objectif est d'optimiser la continuité du service en minimisant à la fois la latence du service et les coûts de migration. Ce problème complexe est formulé comme un Processus Décisionnel Markovien (MDP). Ensuite, nous proposons une stratégie de migration adaptative inédite qui prend en compte des facteurs tels que la mobilité des véhicules, la charge du serveur Edge et différents profils de service. Cette stratégie est élaborée à l'aide de l'Apprentissage par Renforcement Profond (DRL), spécifiquement les

Réseaux Q Profonds (DQN), en se concentrant sur l'algorithme Double Deep Q-Network (DDQN) pour atteindre des compromis optimaux entre latence et coûts de migration.

Cette approche est évaluée en profondeur grâce à des simulations qui modélisent les mouvements et communications véhiculaires au sein d'un réseau doté de MEC, fournissant un scénario réaliste pour tester l'algorithme DDQN. La performance du DDQN est illustrée par l'amélioration des récompenses moyennes par épisode, démontrant l'efficacité d'apprentissage et la convergence de l'algorithme autour de 190 épisodes. Cela indique la capacité de l'agent MEC à explorer et à s'adapter à de meilleures actions. Une analyse comparative de la méthode DDQN proposée par rapport à d'autres approches de migration de service souligne sa supériorité. Spécifiquement, la méthode DDQN excelle dans la gestion des coûts de migration, en particulier pour les profils de service moins sensibles à la latence (SP1 et SP2), tout en maintenant une QoS optimale. Pour les services à haute priorité (SP0), elle surpasse les autres méthodes, établissant un équilibre entre la minimisation des coûts de migration et le maintien d'une faible latence.

Ces résultats montrent comment le modèle DDQN s'adapte habilement aux exigences de chaque profil de service pour atteindre un équilibre optimal entre les coûts de migration et la latence, un facteur essentiel dans les applications V2X avancées où une faible latence est cruciale.

# Acknowledgments

I would like to express my deepest gratitude to my thesis advisors, Prof. Toufik Ahmed and Prof. Mohamed Mosbah, for their invaluable guidance, unwavering support, and insightful feedback throughout the journey of this thesis. Their expertise and mentorship have been pivotal in shaping my research and academic growth.

Following them, I extend my heartfelt thanks to my family, especially my parents and my wife, for their constant support and encouragement. Their unwavering belief in me and their endless love have been the foundation of my strength and resilience throughout this academic pursuit.

I also wish to express my sincere appreciation to the committee members for their time and expertise, which have greatly enriched my work. Their perspectives and constructive critiques have been instrumental in refining my research.

A special note of gratitude goes to the entire team at LaBRI. The collaborative environment and shared knowledge within the team have immensely contributed to my personal and professional development.

Lastly, I am immensely grateful to my friends and brothers, Sabri Khamari, Badreddine Yacine Yacheur, and Imed Ghnaya. Their constant encouragement, thoughtful discussions, and enduring friendship have been a tremendous source of strength and inspiration during the challenges of this academic endeavour.

My journey would not have been the same without the support and encouragement of each one of you. Thank you for being an integral part of this journey and for contributing to its success.

# List of Figures

---

Figure 1. Cooperative intelligent transport system.....	9
Figure 2. ETSI ITS-S reference architecture .....	9
Figure 3. Illustration of V2X communications.....	12
Figure 4. C-V2X operation modes .....	15
Figure 5. LTE-V2X communication mode 3: In coverage & mode 4: Out of coverage....	15
Figure 6. WAVE spectrum and channel types .....	18
Figure 7. Channel allocation for the 5GHz range.....	19
Figure 8. External networks involved in the ITS architecture and their interconnections ..	21
Figure 9. High-level ITS network architecture: Ad hoc-centric.....	22
Figure 10. ITS-G5 protocol stack including DCC functionalities.....	23
Figure 11. Example of state-based DCC algorithm.....	24
Figure 12. An overview of network slicing in enabling smart services .....	27
Figure 13. Life-cycle management of a network slice instance. ....	31
Figure 14. MEC-enabled vehicular network architecture .....	35
Figure 15. LSTM block components.....	42
Figure 16. ML modelling for proactive DCC algorithms.....	43
Figure 17. Average CBR for both DCC algorithms .....	48
Figure 18. Average CBR variation for DCC Dual $\alpha$ .....	49
Figure 19. Average CBR variation for DCC TRC .....	49
Figure 20. Network slicing architecture and slice instantiation for autonomous driving ...	52
Figure 21. Architecture for PKI infrastructure. ....	55
Figure 22. Integration of the solution modules through the ITS station protocol stack .....	57
Figure 23. New PKI architecture.....	58
Figure 24. The process of PC pool attribution to an ITS-S .....	59
Figure 25. Message rate shaping of ITS-Ss belonging to HP and LP slices respectively ...	61
Figure 26. Average latencies for a density of 20 v/km/lane.....	62
Figure 27. Average latencies for a density of 50 v/km/lane.....	62
Figure 28. Latency variations for a density of 20 v/km/lane.....	62
Figure 29. Latency variations for a density of 50 v/km/lane.....	62
Figure 30. ITS-G5 global system architecture overview.....	65
Figure 31. ITS station protocol stack + EMS .....	67
Figure 32. Average RAN latency (vehicle number: 450).....	69
Figure 33. Average RAN latency (vehicle number: 600).....	69
Figure 34. Average RAN latency (vehicle number: 600, HPS: 5%).....	70
Figure 35. Overview of the E2E deployed architecture .....	71

Figure 36. Average E2E latency (vehicle number: 600, HPS: 10%) .....	72
Figure 37. Illustration of the system architecture .....	77
Figure 38. Simulation scenario .....	86
Figure 39. Average reward of MEC agent .....	87
Figure 40. Migration cost results .....	89
Figure 41. Latency results .....	90

# List of tables

---

Table 1: Connectivity requirements of V2X applications .....	14
Table 2: IEEE 802.11p MCSs and corresponding values .....	17
Table 3: Channel allocation in Europe .....	20
Table 4: Default values for AIFS and CW at the access layer .....	23
Table 5: Packet rate settings for a 3-active states TRC .....	24
Table 6: Parameter values of the Dual $\alpha$ algorithm .....	26
Table 7: Summary of V2X communication technologies features.....	38
Table 8: Reactive TRC states and corresponding parameters. ....	46
Table 9: Parameter value of the Dual $\alpha$ algorithm .....	46
Table 10: Simulation parameters .....	47
Table 11: LSTM parameter settings .....	48
Table 12: Average PDR.....	50
Table 13: Data rate simulation parameters. ....	60
Table 14: Traffic prioritization simulation parameters.....	61
Table 15: Median and 95th percentile values .....	63
Table 16: Simulation parameters .....	67
Table 17: Link parameters configuration .....	71
Table 18: System notations .....	78
Table 19: Simulation parameters .....	86



# Acronyms

---

<i>Acronym</i>	<i>Description</i>
3GPP	3rd Generation Partnership Project
5G	5 <sup>th</sup> generation of cellular networks
AA	Authorization Authority
AAA	Authentication, Authorization, and Accounting
AI	Artificial Intelligence
AIFS	Arbitration Inter-Frame Space
AM	Always Migration
AMF	Access and Mobility Management Function
AR	Augmented Reality
BE	Best Effort
BK	Background
BPSK	Binary Phase-Shift Keying
BS	Burst Size
BSS	Basic Service Set
BTP	Basic Transport Protocol
CAM	Cooperative Awareness Messages
CCAM	Cooperative, Connected, and Automated Mobility
CBR	Channel Busy Ratio
CCH	Control Channel
C-ITS	Cooperative Intelligent Transport Systems
CN	Core Network
CPM	Collective Perception Message
C-RAN	Cloud RAN
CriC	Critical Communications
CSMA/CA	Carrier-Sense Multiple Access with Collision Avoidance
C-V2X	Cellular V2X
CW	Contention Window
DCC	Decentralized Congestion Control
DCN	Dedicated Core Network
DENM	Decentralized Environmental Notification Message
DDQN	Double Deep Q-Network
DL	Down Link
DP	DCC Profile
DQL	Deep Q Learning
DQN	Deep Q Networks
DRL	Deep Reinforcement Learning
DSRC	Dedicated Short-Range Communication
E2E	End to End

EA	Enrolment Authority
EC	Enrolment Certificate
EDCA	Enhanced Distributed Channel Access
eMBB	enhanced Mobile BroadBand
EMS	Element Management System
eNB	E-UTRAN NodeB
EPC	Evolved Packet Core
ETSI	European Telecommunications Standards Institute
EWMA	Exponentially Weighted Moving Average
FCC	Federal Communication Commission
FMC	Follow-Me Cloud
FMF	Follow Me Fog
GeoNet	GeoNetworking
gNB	Next Generation NodeB
HARQ	Hybrid Automatic Receive reQuest
HMI	Human-Machine Interface
HP	High Priority
HPS	High Priority Slice
HSS	Home Subscriber Server
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IR	Information Rate
ITS	Intelligent Transport System
ITS-S	ITS Station
IVIM	In-Vehicle Information Message
KPI	Key Performance Indicator
LDPC	Low-Density Parity-Check
LDM	Local Dynamic Map
LIMERIC	Linear Message Rate Integrated Control
LSTM	Long Short-Term Memory
LTE	Long Term Evolution
LP	Low Priority
MAC	Medium Access Control
MANO	Management and Network Orchestration
MAPEM	Map Data Extended Message
MCS	Modulation and Coding Scheme
MDP	Markov Decision Process
MEC	Multi-access Edge Computing
MIB	Management Information Base
MIMO	Multiple-Input Multiple-Output
MME	Mobility Management Entity
mMTC	massive Machine Type Communications

ML	Machine Learning
NF	Network Function
NFV	Network Functions Virtualization
NM	Never Migrate
NRF	NF Repository Function
NR-V2X	New Radio V2X
NSI	Network Slice Instance
OBU	On-Board Unit
OCB	Outside the Context of a BSS
OFDM	Orthogonal Frequency Division Multiplexing
OPS	Ordinary Priority Slice
OS	operating system
OSI	Open Systems Interconnection
PC	Pseudonym Certificate
PCF	Policy Control Function
PCRF	Policy and Charging Rules Function
PDR	Packet Delivery Ratio
PGW	Packet data network GateWay
PHY	Physical
PKI	Public Key Infrastructure
PLMN	Public Land Mobile Network
PNF	Physical Network Functions
QAM	Quadrature Amplitude Modulation
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RAT	Radio Access Technology
RLAN	Radio Local Area Network
RM	Random Migration
RNN	Recurrent Neural Network
RSU	RoadSide Unit
SA-DB	Slice Access Database
SAE	Society of Automotive Engineers
SDN	Software-Defined Networking
SD-RAN	Software Defined RAN
SGW	Serving GateWay
SHC	Service Channel
SINR	Signal to Interference & Noise Ratio
SL	Side Link
SLA	Service Level Agreement
SMF	Session Management Function
SPATEM	Signal Phase and Timing Extended Message

SP	Service Profile
SPS	Semi-Persistent Scheduling
SUMO	Simulation of Urban Mobility
TDC	Transmit Data Rate Control
TC	Traffic Class
TCP	Transmission Control Protocol
TPC	Transmit Power Control
TRC	Transmit Rate Control
UDM	Unified Data Management
UD	User Device
UDP	User Datagram Protocol
UE	User Equipment
UL	Up Link
UPF	User Plane Function
URLLC	Ultra-Reliable Low Latency Communication
V2C	Vehicle-to-Cloud
V2I	Vehicle-to-Infrastructure
V2P	Vehicle-to-Pedestrian
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VANET	Vehicular Ad hoc Networks
VI	VIdeo
VO	VOice
VoIP	Voice over IP
VNF	Virtual Network Function
WAN	Wide Area Network
WAVE	Wireless Access in Vehicular Environments
WLAN	Wireless Local Area Network

## List of Publications

1. Abdennour, R., Ahmed, T., & Mosbah, M. (2021, June). End-to-End Network Slicing for ITS-G5 Vehicular Communications. In *2021 International Wireless Communications and Mobile Computing (IWCMC)* (pp. 266-271). IEEE.
2. Moussaoui, M., Abdennour, R., Ahmed, T., & Mosbah, M. (2022, January). A RAN Slicing Architecture for ITS-G5 Cooperative Intelligent Transport System. In *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)* (pp. 727-728). IEEE.
3. Rachedi, A., Ahmed, T., & Mosbah, M. (2022, May). Proactive C-ITS Decentralized Congestion Control Using LSTM. In *ICC 2022-IEEE International Conference on Communications* (pp. 4342-4347). IEEE.
4. Khamari, S., Rachedi, A., Ahmed, T., & Mosbah, M. (2022, October). Green edge servers placement for intelligent transport systems. In *2022 13th International Conference on Network of the Future (NoF)* (pp. 1-8). IEEE.
5. Khamari, S., Rachedi, A., Ahmed, T., & Mosbah, M. (2023, July). Adaptive Deep Reinforcement Learning Approach for Service Migration in MEC-Enabled Vehicular Networks. In *2023 IEEE Symposium on Computers and Communications (ISCC)* (pp. 1075-1079). IEEE.

## Additional Contributions

1. Main contributor and editor on the InDiD report: Assessment of the 5G network to support C-ITS services; State of the Art: New Technologies & Hybridization.

Date	Version	Author(s)	Updates & changes	Diffusion
13/05/2020	0.07	Bordeaux INP (Editor)	Initial version	COCSIC
28/05/2020	0.09	Bordeaux INP (Editor)	Approved subject of consideration of remarks	COCSIC
08/06/2020	0.1	Bordeaux INP (Editor)	Final submitted version	COCSIC
05/10/2021	0.12	Bordeaux INP (Editor)	Integrate updates based on Release 16	GT 2.7.1
10/11/2021	0.15	Bordeaux INP (Editor)	Submitted version	COCSIC

# Table of Content

Abstract.....	II
Résumé .....	IV
Résumé étendu en français .....	VI
Acknowledgments .....	XII
List of Figures.....	XIII
List of tables .....	XV
Acronyms .....	XVI
List of Publications.....	XX
Table of Content .....	XXI
1. Introduction .....	1
1.1 Thesis Scope and Problem Statement .....	4
1.2 Contributions.....	5
1.3 Dissertation Structure.....	6
2. Background Knowledge and State of the Art .....	7
2.1 Introduction.....	7
2.2 C-ITS and V2X Communications.....	8
2.2.1 C-ITS .....	8
2.2.2 ITS Station Reference Architecture.....	9
2.2.3 Vehicle-to-Everything Communication V2X.....	11
2.2.4 V2X Standard Messages.....	12
2.2.5 Advanced V2X Use Cases and Requirements.....	13
2.2.6 C-V2X .....	14
2.3 Evolution of IEEE 802.11 Related Technologies .....	16
2.3.1 IEEE 802.11p.....	16
2.3.2 IEEE WAVE.....	18
2.4 ETSI ITS-G5 Standard For Direct V2X Communications .....	19
2.4.1 Network Architectures for ITS Stations .....	20
2.4.2 ETSI Decentralized Congestion Control (DCC) .....	22
2.4.3 Transmit Rate Control (TRC) DCC.....	24

2.4.4	Dual $\alpha$ DCC .....	24
2.5	Network Slicing .....	26
2.5.1	Key Aspects .....	27
2.5.2	Slicing Scope in 5G Networks.....	28
2.5.3	Network Slice Instance Lifecycle Management.....	30
2.5.4	Network Slicing For V2X Applications .....	32
2.6	Edge Computing .....	33
2.6.1	Introduction to MEC.....	34
2.6.2	MEC for V2X: Architectural Integration and Service Enhancement.....	35
2.7	Toward a New Technology for Vehicular Networks: IEEE 802.11bd.....	36
2.8	Summary of Key Features of IEEE 802.11, LTE-V2X and 5G-V2X .....	38
3.	Proactive C-ITS Decentralized Congestion Control Using LSTM .....	39
3.1	Introduction.....	39
3.1.1	Problem Statement.....	39
3.1.2	Research Objectives .....	40
3.2	Related Works.....	40
3.3	Proactive DCC Approaches Based on LSTM Recurrent Neural Networks.....	42
3.4	Simulations and Results .....	46
3.4.1	Simulation Scenario and Configurations .....	46
3.4.2	Simulation Results.....	48
3.5	Conclusion .....	50
4.	ETSI ITS-G5 Network Slicing .....	50
4.1	Introduction and Problem Statement .....	50
4.2	Literature Overview .....	51
4.3	A RAN Slicing Architecture for ITS-G5 C-ITS .....	53
4.3.1	Research Objectives .....	53
4.3.2	RAN Slicing Advantages.....	54
4.3.3	Security and Privacy in C-ITS.....	54
4.3.4	RAN Slicing Architecture .....	55
4.3.5	Slice Authentication .....	58

4.3.6	Implementation and Simulation.....	59
4.3.7	Conclusion.....	63
4.4	End-to-End Network Slicing for ITS-G5 Vehicular Communications.....	64
4.4.1	Research Objectives .....	64
4.4.2	Proposed Architecture Overview.....	64
4.4.3	Simulation Scenario, Parameters, and Results .....	67
4.4.4	Conclusion.....	72
5.	Dynamic Service Migration Empowered with Deep Q-Learning for C-ITS .....	73
5.1	Introduction and Problem Statement .....	73
5.2	Research Objectives.....	74
5.3	Related Works.....	75
5.4	System Model .....	76
5.4.1	Network Model.....	79
5.4.2	Computational Model.....	79
5.4.3	Migration Cost Model .....	79
5.5	Problem Formulation .....	80
5.5.1	The MDP formulation .....	80
5.6	Proposed DDQN .....	83
5.7	Simulation and Evaluation.....	85
5.7.1	Simulation Environment.....	85
5.7.2	Simulation results and analysis.....	87
5.8	Conclusion .....	91
6.	General Conclusion .....	92
6.1	Summary and Key Contributions.....	92
6.2	Future Directions and Potential Developments .....	93
	References .....	95



# 1. Introduction

In the current digital era, characterized by rapid economic and technological advancements, the need for safer roads and smarter traffic management has become more pressing than ever. The relentless pace of urbanization and the exponential growth in vehicle numbers have escalated the challenges in traffic management, making road safety a critical concern. This situation necessitates a paradigm shift in how traffic systems are managed and controlled. Innovative solutions are required not only to address the burgeoning traffic demands but also to cater to the evolving expectations of efficiency, safety, and sustainability in urban mobility. As the world moves towards a more interconnected and digitalized future, the transportation sector stands at the cusp of a significant transformation, driven by technological innovations and digital solutions aimed at enhancing road security and optimizing traffic flow.

Cooperative Intelligent Transport Systems (C-ITS) have emerged as a beacon of innovation in the realm of modern transportation, garnering increasing attention worldwide. C-ITS represents an intelligent network where crucial components such as vehicles, pedestrians, and road infrastructure are interconnected, enabling seamless information exchange. This collaborative ecosystem is designed to enhance global road safety and traffic efficiency, marking a substantial leap in how vehicular traffic is managed. By harnessing the power of connectivity and data exchange, C-ITS helps mitigate traffic congestion, reducing accident rates, and improving overall transportation experiences. The integration of these systems into the existing transportation infrastructure paves the way for a smarter, safer, and more efficient mode of road travel, aligning with the futuristic vision of smart cities and intelligent mobility.

Vehicle-to-Everything (V2X) communications stand at the forefront of revolutionizing vehicular networks, playing an indispensable role in the functionality of C-ITS. As the cornerstone of modern vehicular communication, V2X facilitates the exchange of data between vehicles and their surrounding environment, including other vehicles, pedestrians, and road infrastructure. This real-time communication capability enables a plethora of applications that are fundamental to road safety and traffic efficiency. From collision avoidance and traffic flow management to dynamic route planning and environmental monitoring, V2X communications are instrumental in making transportation systems more intelligent, responsive, and user-centric. The integration of V2X into the broader landscape of C-ITS not only enhances situational awareness among road users but also fosters a collaborative network, where every element works in synergy for optimal transportation experience.

Within the diverse landscape of V2X standards, including cellular LTE-V2X and the cutting-edge 5G-V2X, ITS-G5 stands out as the foundational technology for V2X communications in Vehicular Ad hoc Networks (VANET) in Europe. Developed by the European Telecommunication Standard Institute (ETSI), ITS-G5 is built upon the IEEE 802.11p standard,

tailored specifically for vehicular communication. This technology epitomizes the European approach to V2X, providing a robust framework for a range of communications, including vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I), among others. ITS-G5 is designed to support a wide range of applications, from basic safety messages to more complex vehicular communication tasks, ensuring reliability and efficiency in high-speed mobility conditions. The adoption and development of ITS-G5 reflect Europe's commitment to advancing vehicular communication technologies, positioning it as a key player in shaping the future of intelligent transportation systems.

In high-density vehicular environments, the volume of traffic generated by vehicles, especially Cooperative Awareness Messages (CAMs), is immense and serves as the primary source of channel congestion. These CAMs, transmitted periodically, are crucial for maintaining situational awareness among vehicles but simultaneously pose a significant challenge in terms of network congestion. To address this issue, the European Telecommunications Standards Institute (ETSI) proposed Decentralized Congestion Control (DCC) as a cross-layer mechanism for managing and controlling network congestion in VANET. DCC plays a vital role in maintaining the efficiency and reliability of communication channels by dynamically adjusting the transmission parameters based on real-time network conditions. This approach helps in balancing the network load and mitigating congestion, thereby ensuring the continuous and effective exchange of critical safety messages among vehicles in dense traffic scenarios.

Many advanced V2X applications and use cases, such as remote driving and autonomous vehicles, have stringent requirements for latency and reliability. These critical applications demand not just rapid but also highly reliable communication channels to function effectively. In scenarios like remote driving, even a slight delay or a minor lapse in communication can have significant implications, underscoring the necessity for ultra-reliable and ultra-low-latency communication solutions. Addressing this strict Quality of Service (QoS) requires exploring new paths and leveraging advanced technologies. The pursuit of meeting these demanding requirements is driving innovation in the field of V2X, leading to the exploration of novel approaches and cutting-edge technologies. These advancements are crucial in ensuring that V2X systems can support the high levels of performance and reliability essential for the safe and efficient operation of advanced vehicular applications.

The advent of Artificial Intelligence (AI) has marked a transformative era across various sectors, offering a multitude of applications and benefits that can significantly enhance diverse fields, including V2X networks. AI's ability to process vast amounts of data, learn from patterns, and make informed decisions would revolutionize the way V2X networks operate. In the context of vehicular communication, AI contributes to improved safety, efficiency, and traffic management by enabling smarter decision-making and predictive analytics. This

technology is instrumental in analysing real-time traffic data, optimizing routing, and enhancing the overall performance of vehicular networks. AI's integration into V2X systems paves the way for advanced features such as autonomous driving, traffic congestion prediction, and dynamic network optimization, showcasing its pivotal role in elevating intelligent transportation systems to new heights.

Furthermore, the emergence of 5G technology has unlocked a plethora of new opportunities for innovation and enhancement in C-ITS. 5G brings to the table high-speed connectivity, lower latency, higher capacity, and improved reliability, all of which are crucial for the effective functioning of C-ITS. Key technologies that play a significant role in this advancement include Network Slicing and Multi-access Edge Computing (MEC). These technologies collectively contribute to the creation of more efficient, scalable, and robust C-ITS. The potential for real-time data processing, enhanced vehicular communication, and support for a wider range of applications become a reality, thereby propelling C-ITS into a new era of technological excellence and paving the way for smarter, more connected transportation systems.

Network slicing stands as a fundamental enabler for 5G networks, offering a novel approach to network management and optimization. This technology involves the creation of multiple independent logical networks, known as slices, on a shared physical infrastructure. Each slice is allocated dedicated resources and tailored virtual network functions to cater to specific users or traffic types. Network slicing allows for customized solutions that meet diverse requirements, ranging from high bandwidth to low latency, ensuring optimal performance for various applications. In the context of C-ITS and V2X communications, network slicing provides a flexible and efficient way to handle different traffic scenarios and service demands, significantly enhancing the capability of vehicular networks to support a broad spectrum of applications with distinct QoS needs.

The Multi-access Edge Computing (MEC) paradigm, characterized by deploying servers at the network's edge, is a significant advancement in vehicular network technology. This strategic placement of edge servers near vehicular networks brings powerful computational capabilities closer to the point of data generation and consumption. By doing so, MEC significantly reduces latency, a crucial factor for applications requiring prompt responses. This efficient deployment enables vehicles to access robust computational resources, essential for processing-intensive tasks like object detection, video stream analytics, and path navigation. The integration of MEC in vehicular networks is a game-changer, especially for advanced applications that demand high computational power and low-latency communication. It facilitates the seamless execution of complex tasks directly within the vehicular network, enhancing performance, and ensuring a more responsive and efficient vehicular environment.

The integration of advanced technologies such as AI, network slicing, and MEC offers a comprehensive solution to a multitude of challenges facing vehicular environments. These

technologies collectively address critical aspects such as stringent QoS requirements, security concerns, and heavy computational loads. By leveraging the capabilities of these technologies, vehicular networks can achieve higher levels of efficiency, reliability, and safety. Together, these technologies, along with others, provide a robust framework to overcome the challenges of vehicular networks, paving the way for advanced and intelligent transportation systems.

## 1.1 Thesis Scope and Problem Statement

In an era where technological advancements are reshaping the landscape of vehicular communications, this thesis explores the intricate interplay of various cutting-edge fields such as AI, Network Slicing, MEC within C-ITS, and V2X communications. The integration of these technologies in the European standard ITS-G5 for VANETs solves many challenges and brings forth a myriad of challenges and opportunities. The scope of this thesis encompasses addressing these critical issues, leading to several research questions:

- How can the challenge of channel congestion in high-density vehicular networks be proactively managed?

Channel congestion, particularly in C-ITS, significantly impedes the efficacy of V2X communications, crucial for safety and traffic management. Traditional DCC strategies often react to congestion after it occurs, leading to delays and inefficiencies. This thesis investigates proactive approaches, integrating AI and advanced algorithms, to predict and manage congestion effectively, thereby enhancing network performance and traffic flow.

- What strategies can be employed to ensure ultra-low latency and high reliability in dynamic traffic environments?

The demand for ultra-low latency and ultra-high reliability in ITS-G5 VANETs, especially for critical applications such as autonomous driving, poses a significant challenge. This thesis explores innovative solutions, namely network slicing, to create dedicated end-to-end (E2E) communication channels that cater to specific user groups and traffic types. These solutions aim to maintain the stringent QoS requirements essential in rapidly changing vehicular environments.

- How can MEC be optimized for computationally intensive applications in high-mobility vehicular networks to ensure cost efficiency and adherence to strict low-latency requirements?

With the emergence of MEC, the frequent movement of vehicles leads to changing communication topologies and unreliable links. This thesis delves into dynamic and responsive strategies within the MEC infrastructure to ensure continuous service and minimal latency

while optimizing costs, particularly for applications like video streaming and augmented reality that demand high computational power and swift data processing.

Through addressing these research questions, the thesis aims to contribute significantly to the advancement of intelligent and efficient vehicular communication systems, paving the way for enhanced road safety, traffic management, and user experiences in the realm of smart mobility.

## 1.2 Contributions

This thesis delves into the development of innovative solutions for addressing the complex challenges faced by C-ITS and V2X communications, particularly within the framework of the European standard ITS-G5 for VANETs. The research is rooted in the application of emerging technologies such as Artificial Intelligence, Network Slicing, and Multi-access Edge Computing, aiming to enhance the efficiency, reliability, and overall performance of vehicular networks. The main contributions of this thesis are summarized as follows:

1. *Proactive C-ITS Decentralized Congestion Control Using LSTM*: This contribution introduces a proactive approach to DCC in vehicular networks. By integrating AI with the DCC Dual  $\alpha$  algorithm, the work presents an innovative method to predict and manage channel congestion before it becomes problematic. This approach significantly improves the handling of data transmission in high-density traffic scenarios, ensuring better resource allocation fairness, and enhanced network efficiency.

2. *A RAN Slicing Architecture for ITS-G5*: We propose a novel Radio Access Network (RAN) slicing architecture specifically tailored for ITS-G5 networks. This architecture facilitates the creation of multiple network slices, each with its own set of resources and priorities. It effectively addresses the diverse communication needs within the vehicular network, ensuring optimized QoS for various traffic types, thereby enhancing the overall network performance and reliability.

3. *End-to-End Network Slicing for ITS-G5 Vehicular Communications*: Building upon the concept of RAN slicing, this contribution extends it to an end-to-end context. We develop an E2E network slicing framework for ITS-G5, enabling more efficient management of vehicular communication from the source to the destination. This approach allows for tailored service provisioning, especially in congested environments, ensuring ultra-low latency and high reliability for critical vehicular applications.

4. *Adaptive Deep Reinforcement Learning Approach for Service Migration in MEC-enabled Vehicular Networks*: This last contribution introduces an adaptive service migration strategy using Deep Reinforcement Learning (DRL) for MEC-enabled vehicular networks. By

employing techniques such as Deep Q Networks (DQN) and Double Deep Q-Networks (DDQN), this work presents a solution that dynamically adapts to the high mobility of vehicles and the shifting communication topologies. This strategy optimizes service continuity, minimizes latency, and reduces migration costs, thereby significantly enhancing the performance and reliability of MEC-enabled vehicular networks.

These contributions collectively address key challenges in C-ITS and V2X communications, offering robust and innovative solutions that significantly advance the field of intelligent transportation systems.

### **1.3 Dissertation Structure**

This introductory chapter sets the stage for the thesis by outlining the context of C-ITS, V2X communications, and Network Slicing, among others. It highlights the key challenges within this domain and introduces the contributions of the thesis. This chapter serves as a foundational overview, introducing the reader to the technological landscape and the pivotal issues addressed throughout the thesis. The second chapter provides a comprehensive exploration of the current state of the art, encompassing a thorough description of the context and theoretical underpinnings related to C-ITS and V2X communications (focusing on ITS-G5), Network Slicing, and Multi-access Edge Computing. It delves into existing technologies, methodologies, and the evolution of vehicular networks, setting the groundwork for the thesis's contributions.

Focusing on the first contribution, the third chapter discusses both adaptive and reactive DCC methods, leading to the elaboration of the proposed proactive Dual  $\alpha$  DCC. It includes an in-depth presentation of the methodology, followed by simulations and the analysis of results, demonstrating the effectiveness of the proposed solution. Chapter four begins with an overview of network slicing literature specific to V2X. It then presents the second contribution, RAN Slicing Architecture for ITS-G5, detailing the proposed architecture, simulation setup, and results. Subsequently, it introduces the third contribution, End-to-End Network Slicing for ITS-G5 Vehicular Communications, again outlining the proposed solution, simulation framework, and key findings. The fifth chapter encompasses the fourth contribution, focusing on Dynamic Service Migration in C-ITS using Deep Q-Learning. This chapter covers the system model, problem formulation, detailed simulations, and a thorough evaluation, highlighting the efficacy of the proposed methodology in MEC-enabled vehicular networks.

The last chapter concludes the manuscript by summarizing the context, challenges, and key contributions and findings of the thesis. It reflects on the implications of the research and outlines potential future works, suggesting directions for further investigation and development within the field of intelligent transportation systems.

## 2. Background Knowledge and State of the Art

### 2.1 Introduction

In 1990, foreseeing the potential of Intelligent Transport Systems (ITS), the U.S. Federal Communication Commission (FCC) designated a 75 MHz bandwidth within the licensed 5.9 GHz band (5.885-5.925 GHz) exclusively for ITS applications. This critical decision, offering the spectrum freely without reservation requirements, significantly propelled vehicular communication technologies forward.

In recent years, C-ITSs have become increasingly prominent. C-ITS represents a smart sophisticated network where connected entities, including vehicles, pedestrians, and road infrastructure, collaboratively share data. The primary objectives of C-ITS are to enhance road safety, optimize traffic flow, reduce emissions, and improve the overall driving experience.

V2X communications are crucial in enabling data exchange for diverse applications such as cooperative autonomous driving, traffic efficiency, infotainment services, and vehicular software updates and diagnostics. The varied nature of these applications demands stringent connectivity requirements, including ultralow latency (below 10 ms), ultrahigh reliability (nearing 100%), and high data throughput (in the order of Gbps). These requirements become increasingly challenging to meet due to the high mobility and density of vehicles.

5G, the fifth generation of cellular networks, is set to transform various vertical industries with its innovative capabilities in communication, automation, sensing, and positioning. This technology promises to revolutionize transportation and enhance the quality of life through its three primary use case categories: enhanced Mobile Broadband (eMBB) for high-speed data; Ultra-Reliable and Low Latency Communications (URLLC) for critical applications requiring fast, reliable connectivity; and massive Machine Type Communications (mMTC) for large-scale IoT deployments. These advancements position 5G as a key driver in the next wave of technological evolution.

Network slicing is an important aspect of 5G networks, as it enables the creation of customized and efficient networks for different use cases and services. It also allows for more efficient use of network resources and improved network performance. For example, a network slice for a self-driving car would have low latency and high reliability to ensure smooth and safe operation. Another slice for a virtual reality gaming service would have high bandwidth and low jitter to provide a seamless gaming experience.

Integrating Edge Computing into this framework, where data processing occurs in proximity to the data source, significantly reduces latency and improves efficiency. Moreover, Service

Migration facilitates dynamic service relocation across network nodes, ensuring continuity and optimal performance in mobile environments like vehicular networks. This combination of Edge Computing and Service Migration is poised to enhance intelligent transportation systems, enabling real-time processing and seamless connectivity essential for applications such as autonomous driving and traffic management.

In this chapter, we begin by introducing C-ITS and the diverse realm of V2X communications. We then delve into the IEEE 802.11p standard, exploring its derivative technologies and their implications. A focal point of our discussion is the ITS-G5 communication standard; we will examine its technical specifications and underscore its pivotal role in enhancing vehicular communication within the European V2X context. Progressing further, we dive into the concept of network slicing, highlighting its importance in developing efficient and flexible networks in the 5G era. Concurrently, we investigate the transformative role of edge computing, emphasizing its significance in expediting data processing at the edge of the network. Following this, we introduce the emerging IEEE 802.11bd standard, underscoring its projected enhancements and the potential impact it holds for the future landscape of V2X communications. The chapter concludes with a comprehensive comparative summary of various V2X communication technologies and their features.

## **2.2 C-ITS and V2X Communications**

### **2.2.1 C-ITS**

Cooperative Intelligent Transport Systems (C-ITS) are a sophisticated blend of technology and communication systems, designed to significantly enhance road safety, traffic efficiency, and driving convenience. C-ITS networks involve vehicles, roadside infrastructure, and other components like pedestrians (Figure 1 [1]), all interconnected to share data and information. These systems utilize various communication technologies, encompassing both Wi-Fi-based (IEEE 802.11) and cellular-based (C-V2X: 4G and 5G) standards, to enable ad hoc networking tailored for road safety and traffic efficiency.

Key features of C-ITS include the ability to disseminate warnings about road hazards, provide real-time information about traffic conditions for speed management, and facilitate navigation. These systems depend on always-on connectivity among vehicles and roadside infrastructure, necessitating frequent data exchange. Additionally, C-ITS enhances driving convenience with internet access and location-based services, such as point-of-interest notifications, road access control, and parking management.





Figure 1. Cooperative intelligent transport system

C-ITS also incorporates specific message sets that have become the current standard in this field. The evolution of C-ITS in Europe can be traced back to the 1980s, with significant milestones like the PROMETHEUS project (1987–1994)[2], which marked the advent of cooperative driving research and development in the region.

### 2.2.2 ITS Station Reference Architecture

The ITS station reference architecture explains the functionalities contained in ITS stations. It is derived from the OSI model, with an extension to include and support ITS applications. Figure 2 presents the reference architecture of an ITS station (ITS-S) [3].

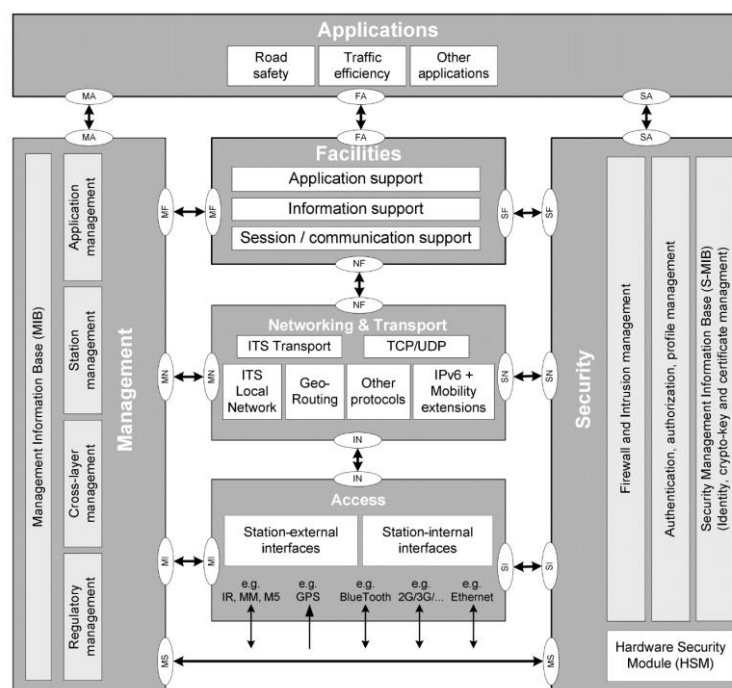


Figure 2. ETSI ITS-S reference architecture

The architecture is stratified into layers and blocks, with each layer representing specific functions and communicating through designated interfaces such as IN, MI, and SF, ensuring seamless interaction within the system [4].

- **Access Layer:** At the foundational level, the Access layer corresponds to layers 1 and 2 of the OSI model, specifying different technologies that provide the physical and data link connections. This includes not only ITS-G5 and Dedicated Short Range Communication (DSRC) for short-range direct communication but also encompasses Wi-Fi and Cellular for wider network access. This layer is essential for establishing the initial connection between ITS stations and for the dissemination and reception of V2X messages.

- **Networking & Transport Layer:** Representing OSI layers 3 and 4, this layer is tasked with the efficient routing and transportation of data packets. It incorporates ITS-specific transport protocols like the Basic Transport Protocol (BTP), tailored for V2X message exchange, and adapts traditional protocols such as TCP/UDP for ITS needs. GeoNetworking is another key component, providing geographical addressing and routing suitable for the dynamic nature of vehicular networks. An ITS management entity within this layer oversees network configuration and operation, distinct from the overarching Management block.

- **Facilities Layer:** Correlating with OSI layers 5, 6, and 7, the Facilities layer provides high-level data processing and support functionalities crucial for ITS applications. It features the Local Dynamic Map (LDM), a dynamic database that maintains real-time information about the vehicle's surroundings, including data received from neighbouring vehicles. This layer also handles the selection of addressing modes, ensures synchronization with accurate positioning and timing sources, and manages application support for user interactions via the Generic Human-Machine Interface (HMI).

- **Security Layer:** This layer weaves throughout the architecture, offering vital services that bolster the security and integrity of V2X communications. It encompasses firewall and intrusion detection systems to safeguard the network, along with mechanisms for authentication and authorization to verify the identity of devices and control access. It also provides encryption and data protection services to maintain the confidentiality and integrity of exchanged information.

- **Management Layer:** This overarching layer plays a critical role in maintaining the health and efficiency of the ITS-S. It interfaces with the Management Information Base (MIB), where operational data is stored and managed. The Management block oversees the entire communication stack, handling system configuration, monitoring performance, and ensuring the compliance of communication protocols.

- **Applications Layer:** At the apex, the Applications layer is where the suite of ITS services is implemented. It's split into sub-categories such as Road Safety and Traffic Efficiency, each utilizing underlying services to deliver specific functionalities to end-users. This layer is the most visible to the users, delivering tangible ITS applications like collision warnings, traffic alerts, and route guidance.

### **2.2.3 Vehicle-to-Everything Communication V2X**

V2X refers to the communication between vehicles and other entities, called ITS stations (ITS-S), such as other vehicles, infrastructure, and devices. In V2X communication, sidelink communication is used for direct communication between ITS stations, whereas uplink/downlink communication is used for communications between an ITS station and a cellular base station [5][6]. Sidelink communication is often used for real-time short-range data sharing, such as speed, location, and braking status. Uplink/downlink is used for longer-range applications such as internet access, phone calls, or sending data to the cloud.

Several V2X communications scenarios have been developed and tested, as shown in Figure 3 [5], including [7]:

1. **Vehicle-to-vehicle (V2V) communication:** Vehicles share information such as speed, location, and braking status with nearby vehicles. This can improve road safety by allowing vehicles to alert each other of potential hazards and coordinate their movements.
2. **Vehicle-to-infrastructure (V2I) communication:** Vehicles communicate with traffic lights, road signs, Roadside Units (RSU), and other infrastructure to receive real-time information about traffic conditions and optimize their routes.
3. **Vehicle-to-pedestrian (V2P) communication:** Vehicles communicate with pedestrians and other vulnerable road users, such as bicyclists, to improve road safety and provide them with relevant information.
4. **Vehicle-to-network (V2N) communication:** Vehicles connect to a cellular network to access the internet and other services. This can provide drivers and passengers with access to navigation, entertainment, and other services.
5. **Vehicle-to-cloud (V2C) communication:** Allows vehicles to communicate with cloud-based services to access real-time data and perform advanced analytics.

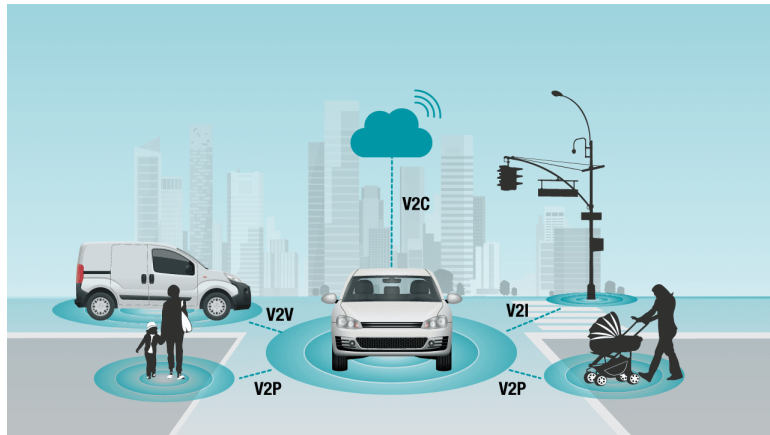


Figure 3. Illustration of V2X communications

## 2.2.4 V2X Standard Messages

In the context of V2X communications, there are several key message types, each designed for specific purposes to enhance road safety, and traffic efficiency, and support various vehicular applications. These message types include [8]:

1. Cooperative Awareness Messages (CAMs): Used to create and maintain situational awareness among vehicles and infrastructure. It includes information about the vehicle's presence, position, dynamics, and other attributes, which helps in assessing situations like potential collision risks.
2. Decentralized Environmental Notification Messages (DENMs): These messages alert other road users about unexpected events that can affect road safety or traffic conditions. Typical contents include information about road hazards, accidents, or abnormal traffic conditions.
3. Collective Perception Message (CPM): Designed to transmit information about locally detected objects, such as non-cooperative traffic participants, obstacles, etc. It enhances situational awareness by providing standardized representations of these detected objects.
4. Map Data Extended Message (MAPEM): Provides information about road and intersection topologies. This can include special permissions for automated driving in specific lanes or designated safe spots for manoeuvres.
5. Signal Phase and Timing Extended Message (SPATEM): Relays information about the state of signalized intersections, including phase and timing details. Useful for managing traffic flow and optimizing signal timings for efficient traffic management.
6. In-Vehicle Information Message (IVIM): Transmits both static and dynamic road sign and message sign information, typically found on highways. These messages are crucial for providing drivers with timely and relevant traffic and navigation information.

## 2.2.5 Advanced V2X Use Cases and Requirements

In 3rd Generation Partnership Project 3GPP systems, it is envisioned to support the following four generic V2X use cases summarized in Table 1 [7][9]:

1. Vehicle platooning enables vehicles to form a group traveling together, and all of them receive periodic data from the leading vehicle. Main vehicles platooning performance requirements include payloads in the ranges of 50-6000 Bytes, end-to-end delays in the ranges of 10-500 ms, reliability in the ranges of 90%-99.99%, data rates in the ranges of 50-65 Mbps, and a minimum required communication range between 80 and 350 meters.

2. Advanced driving where each vehicle shares data obtained from its local sensors with other nearby vehicles or allows vehicles to share their driving intention and coordinate their trajectories or manoeuvres with vehicles in proximity. Main advanced driving performance requirements include payloads in the ranges of 300-12000 Bytes, end-to-end delays in the ranges of 3-100 ms, reliability in the ranges of 90%-99.999%, data rates in the ranges of 10-53 Mbps, and a minimum required communication range between 360 and 500 meters.

3. Extended sensors enable the vehicles to exchange raw or processed data gathered through their local sensors or live video data among other nearby vehicles, Roadside Units, UEs of pedestrians, and even V2X application servers at edge nodes. Main extended sensors performance requirements include payloads of 1600 Bytes, end-to-end delays in the ranges of 3-100 ms, reliability in the ranges of 90-99.999 %, data rates in the ranges of 10-1000 Mbps), and a minimum required communication range between 50 and 1000 meters.

4. Remote driving enables a remote driver or a V2X application to operate a remote vehicle. Main remote driving performance requirements include end-to-end delays less than or equal to 5 ms, reliability equal to or better than 99.999 %, and data rates around 25 Mbps for uplink traffic, and 1 Mbps for downlink traffic.

As vehicular systems evolve towards higher levels of automation, their service requirements, particularly in latency, become more stringent. Advanced driving applications might need latencies of around 100 ms for effective information sharing, whereas remote driving applications demand significantly lower latencies, often as low as 1 ms. While 3GPP's efforts are concentrated on fulfilling these stringent connectivity requirements, these advanced applications also necessitate substantial computing resources. This includes the development of remote vehicle control systems and the processing of extensive sensor data. In this context, the integration of solutions such as edge computing becomes critical, offering the necessary computational power closer to the data source, thereby facilitating rapid data processing and analysis essential for the efficient operation of these highly automated vehicular technologies.

Table 1: Connectivity requirements of V2X applications

Use Case	Communication Mode	Payload (Bytes)	Max. Delay (ms)	Datarate (Mbps)	Minimum Range (m)	Reliability
Advanced driving	V2V, V2I	300-12,000	3-100	10-50	60-500	90-99.999
Remote driving	V2N	--	5	UL:25 DL:1	--	99.999
Vehicle platooning	V2V, V2I	50-6000	10-500	50-65	80-350	90-99.99
Extended sensors	V2V, V2I, V2P	1600	3-100	10-1000	50-1000	90-99.999

## 2.2.6 C-V2X

To keep pace with the evolving landscape of vehicular communications and the diversifying use cases, the 3GPP introduced a V2X standard based on Long-Term Evolution (LTE) technology in 2017. This development, part of the Release 14 (Rel-14) specifications, is commonly known as "cellular V2X" (C-V2X) or LTE-V2X. This designation serves to distinguish it from the earlier 802.11p-based V2X technology, which is Wi-Fi-based.

Following the initial establishment of the LTE-based V2X standard, 3GPP has continued to make significant enhancements to V2X communications. These advancements, incorporated in Release 15/16 and subsequent releases, are designed to support advanced scenarios in Cooperative, Connected, and Automated Mobility (CCAM). This next phase of V2X communication development, utilizing 5G specifications, is referred to as New Radio V2X (NR-V2X) [10]. NR-V2X represents a significant leap forward, bringing improved capabilities and higher performance to meet the increasingly complex and demanding requirements of future vehicular communication systems.

### 2.2.6.1 Operation modes for C-V2X

There are two modes of operation for C-V2X communication, as shown in Figure 4 [3]:

- V2X communication over PC5 interface (aka Sidelink SL): PC5 interface directly connects ITS stations so that over-the-air V2X message from an ITS station is directly received by ITS stations around the transmitter.
- V2X communication over LTE-Uu interface (Uplink/Downlink): The LTE-Uu interface connects ITS stations with eNB (E-UTRAN NodeB), which plays the role of a base station in the LTE networks.

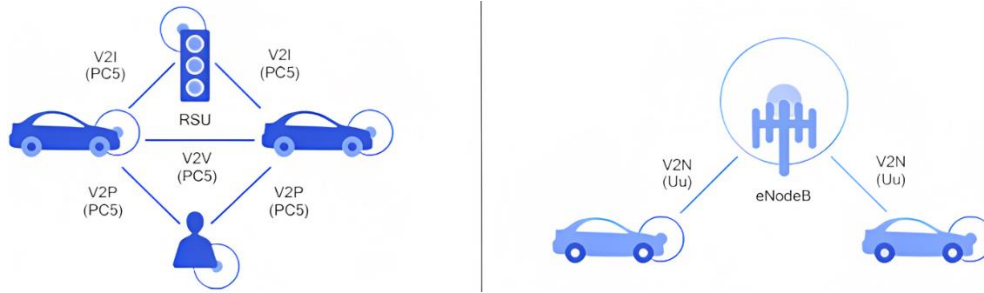


Figure 4. C-V2X operation modes

### 2.2.6.2 LTE-V2X communication modes:

The LTE-V2X PC5 interface includes two modes of communication: mode 3 and mode 4 [11], as illustrated in Figure 5.

**Mode 3:** This mode is also referred to as "base station-scheduled" and "cellular-assisted". In this mode, vehicles need to be in the coverage zone of the base station because the radio resources of UEs are managed and selected under the eNodeB station via control signalling over the Uu interface. Thus, the frequency channel under license by the operator will be used.

In mode 3, each vehicle reports its location and coordinates to assist the eNB in resource scheduling. Mode 3 is only available when vehicles are within network coverage. 3GPP has proposed network architecture enhancements to support V2X. Among these enhancements is the V2X control function, mainly used to manage radio resources and to provide vehicles with the SL V2X configurable parameters.

**Mode 4:** This mode is also known as "pure Ad Hoc V2V" and "autonomously scheduled". In this mode, vehicles do not need to be in the coverage zone of the base station because vehicles implement a mechanism for autonomous radio resource selection based on sensing before transmission with Semi-Persistent Scheduling (SPS). This mode will be used for V2V communications use cases since communication between vehicles cannot depend on cellular coverage. For this purpose, a distributed SPS congestion control mechanism is developed.

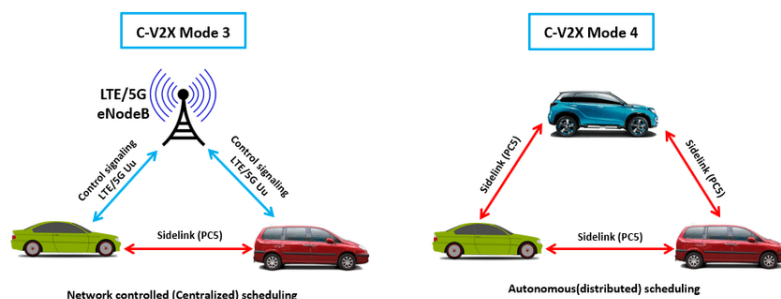


Figure 5. LTE-V2X communication mode 3: In coverage & mode 4: Out of coverage

### 2.2.6.3 5G V2X

New Radio V2X (NR-V2X) in Release 16 marks a pivotal advancement as the first NR sidelink standard, specifically engineered to enhance the reliability, latency, capacity, and flexibility of V2X communications. This release is notable for incorporating a range of communication methods, including unicast, multicast (groupcast), and broadcast, broadening the scope and efficiency of V2X interactions. A key feature of NR-V2X Rel-16 is the integration of the Hybrid Automatic Receive reQuest (HARQ) system, which significantly improves the reliability of sidelink communications by enabling more robust error correction and retransmission strategies. These developments [12] represent a substantial step forward in the evolution of V2X technology, paving the way for more advanced and reliable vehicular connectivity solutions.

Rel. 16 specifies two additional modes, modes 1 and 2 [9], for sub-channel selection in NR V2X SL communications using the NR V2X PC5 interface. These two modes correspond to mode 3 and mode 4 in LTE V2X. NR V2X supports broadcast, group cast, and unicast SL communications, but LTE V2X only supports broadcast SL communications.

- **Mode 1:** Similar to mode 3 in LTE V2X, the gNB (Next Generation NodeB) or eNB assigns and manages the SL radio resources for V2X communications under mode 1 using the NR (or LTE) Uu interface. SL radio resources can be allocated from licensed carriers dedicated to SL communications or licensed carriers sharing resources between SL and UL communications. ITS stations must therefore be in network coverage to operate using mode 1.

- **Mode 2:** Like mode 4 in LTE V2X, ITSs can autonomously select their SL resources (one or several sub-channels) from a resource pool when using mode 2 in NR V2X. In this case, ITS stations can operate without network coverage. The resource pool can be (pre-)configured by the gNB or eNB when the ITS is in network coverage. Mode 2 can operate using a dynamic or semi-persistent scheduling scheme.

## 2.3 Evolution of IEEE 802.11 Related Technologies

### 2.3.1 IEEE 802.11p

IEEE 802.11p is an extension of 802.11a (Wi-Fi) and was standardized by the IEEE in 2009 [12]. The IEEE 802.11p is based on the Carrier Sense Multiple Access protocol with Collision Avoidance (CSMA/CA) which is a statistical protocol for V2V and V2I communications.

This standard was developed as a vehicular communication standard that provides safety applications and better traffic management with a set of requirements as follows [13]:

- Relative velocities up to 200 km/hr.



- Response times of around 100 milliseconds.
- A communication range of up to 1000 m.

IEEE 802.11p sets the foundational protocols for both the Physical (PHY) and Medium Access Control (MAC) layers in vehicular communications. Building upon the structure established by IEEE 802.11p, various supplementary standards have emerged, forming distinct frameworks in both the United States and Europe [14]. In the United States, the suite of IEEE 1609 standards extends the foundational work of 802.11p, focusing on additional protocol layers beneath the application layer. These layers are further defined and regulated by the Society of Automotive Engineers (SAE). In contrast, Europe has embraced IEEE 802.11p through its integration into the ETSI ITS-G5 standard, which, along with a series of complementary documents, addresses the comprehensive scope of protocol layers above the foundational PHY and MAC layers established by 802.11p.

At the Physical (PHY) layer, IEEE 802.11p employs Orthogonal Frequency Division Multiplexing (OFDM), a method that efficiently manages the spectrum by dividing it into multiple subcarriers. This implementation includes 48 usable subcarriers for data transmission, along with an additional four subcarriers designated for pilot signals, which assist in maintaining the integrity and robustness of the communication. The duration of an OFDM symbol is 8 microseconds ( $\mu\text{s}$ ), and the subcarrier spacing is 156.25 kHz. This arrangement results in a total raw bandwidth of 10 MHz, optimized for V2X communication environments. In addition to the OFDM configuration, IEEE 802.11p supports a range of eight Modulation and Coding Schemes (MCSs). These schemes span from simpler modulations like Binary Phase-Shift Keying (BPSK), offering robustness, to more complex forms such as 64 Quadrature Amplitude Modulation (64-QAM), which allows for higher data rates. Accompanying these modulation techniques is the use of convolutional coding, which may include punctured coding to balance data throughput with error correction. These various modulation and coding combinations are comprehensively detailed in Table 2 [15].

*Table 2: IEEE 802.11p MCSs and corresponding values*

MCS #	Modulation Coding Rate	Gross Rate (Mb/s)	Minimum SINR [dB]	Range [m] (only PL)	Duration [ $\mu\text{s}$ ]
1	BPSK,1/2	3.0	10.0	223	848
2	BPSK,3/4	4.5	11.0	210	584
3	QPSK,1/2	6.0	13.0	188	448
4	QPSK,3/4	9.0	15.0	167	312
5	16QAM,1/2	12.0	18.0	141	248
6	16QAM,3/4	18.0	22.0	112	176
7	64QAM,2/3	24.0	26.0	89	144
8	64QAM,3/4	27.0	27.0	84	136

Since the MAC layer of IEEE 802.11p uses the CSMA/CA as an access method, when a node needs to transmit data, it first checks if the medium is idle. If not, a random backoff mechanism is employed to reduce the likelihood of collisions. Unique to vehicular communications in IEEE 802.11p, acknowledgments for transmissions are typically omitted, and as a result, the exponential backoff strategy commonly used for retransmissions in other contexts is not applicable.

### 2.3.2 IEEE WAVE

Standardization efforts for the Wireless Access in Vehicular Environments (WAVE) architecture began in 2004, culminating in the establishment of specifications for both the physical and MAC layers in the IEEE 1609.4 and IEEE 802.11p standards [16].

WAVE operates within the dedicated 75 MHz spectrum in the 5.9 GHz band, ranging from 5850 to 5925 GHz. At the physical layer, WAVE employs Orthogonal Frequency Division Multiplexing (OFDM) with 10 MHz channels, differing from the 20 MHz channels used in the 802.11a standard. The MAC layer utilizes the Enhanced Distributed Channel Access (EDCA) protocol, an adaptation designed to cater to the unique requirements of vehicular communications [17].

In the WAVE system, the total allocated spectrum is strategically divided into seven distinct channels. These include the Control Channel (CCH), designated solely for safety-related communications, and Service Channels (SCH), which are used for both safety and non-safety communications. The allocation and distribution of these channels are depicted in Figure 6 [18]:

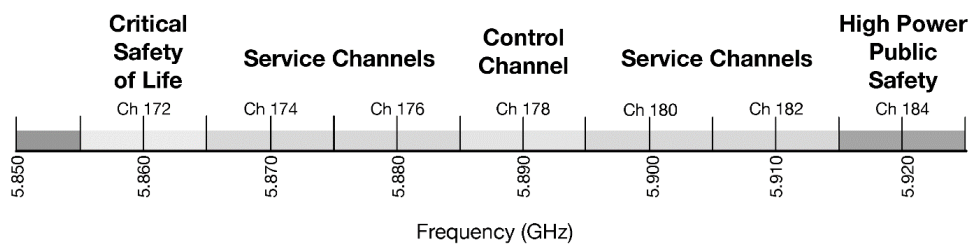


Figure 6. WAVE spectrum and channel types

Congestion management within the network is facilitated through an uncoordinated channel access strategy based on CSMA/CA. However, this method has demonstrated suboptimal performance in high-density scenarios. This is primarily due to the inherent nature of CSMA/CA, which relies on nodes independently sensing the channel's availability before initiating communication. In high-density scenarios, this method can result in increased collisions and subsequent network delays due to the frequent simultaneous attempts to access the channel [19].

## 2.4 ETSI ITS-G5 Standard for Direct V2X Communications

The ETSI ITS-G5 is the European technology used for direct (or sidelink) communications in vehicular environments, standardized since 2009, using the 5.9 GHz band. The harmonization of the spectrum 5850-5925 GHz, for intelligent transport security applications, was the object of the decision 2008/971/CE of the European Commission on August 8th, 2008 [20].

ITS-G5's primary objective is to facilitate the reliable delivery of short-range safety information. This technology excels in scenarios demanding low latency, and it is robust enough to support both high-density traffic environments and high-velocity vehicular movement.

The frequency allocation for ITS-G5 in Europe is specified into four different bands, as shown in Figure 7 [21]:

- ITS-G5A band: Used for ITS road traffic safety applications.
- ITS-G5B band: Used for ITS non-safety road traffic applications.
- ITS-G5D band: This band is set aside for future ITS applications.
- ITS-G5C band: Refers to Radio Local Area Network (RLAN). The use of this band is not supported when the Management Information Base (MIB) parameter dot11OCBActivated is set to True. Therefore, ITS-G5 stations can't communicate Outside the Context of a BSS (Basic Service Set) OCB mode (more details are given in [21]).

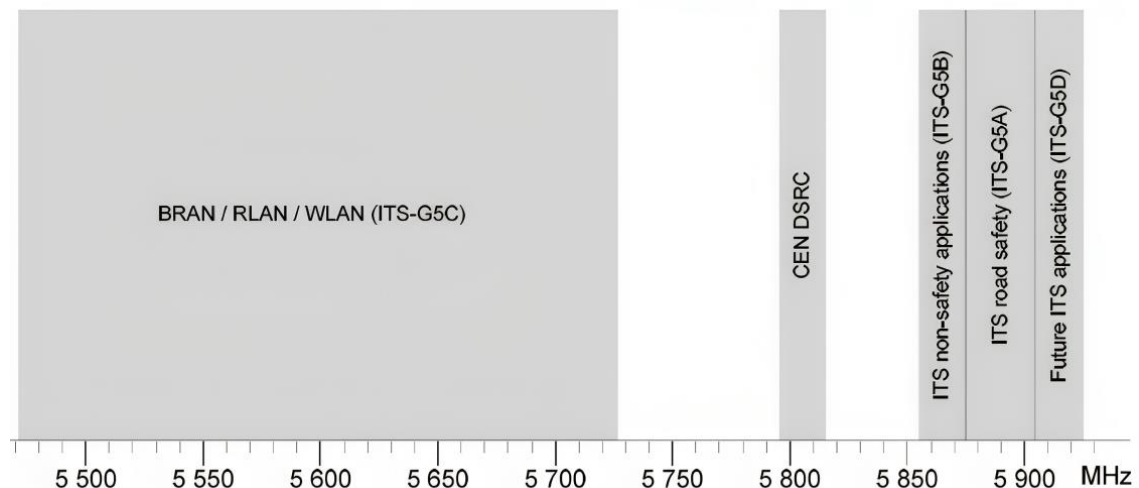


Figure 7. Channel allocation for the 5GHz range

Regarding channel types, there are two primary categories: CCH and SCH. The channel distribution within the European 5.9GHz spectrum for ITS-G5 services is comprehensively outlined in Table 3.

For ITS-G5A, B, and D bands, usage is restricted exclusively to ITS-G5-compliant stations. It's important to note that the 5.8GHz band is Europe's designated frequency for Dedicated Short Range Communication (DSRC), used for electronic toll collection (so-called CEN DSRC), in line with the EN 300 674 standard [22].

*Table 3. Channel allocation in Europe*

	<b>Channel type</b>	<b>Frequency range (MHz)</b>	<b>IEEE Channel number</b>
ITS-G5A	G5-CCH	5895 to 5905	180
	G5-SCH2	5885 to 5895	178
	G5-SCH1	5875 to 5885	176
ITS-G5B	G5-SCH3	5865 to 5875	174
	G5-SCH4	5855 to 5865	172
ITS-G5C	G5-SCH7	5470 to 5725	94 to 145
ITS-G5D	G5-SCH5	5905 to 5915	182
	G5-SCH6	5925 to 5925	184

### 2.4.1 Network Architectures for ITS Stations

The network architecture comprises external and internal networks. External networks interconnect ITS stations among each other or connect ITS stations to other network entities. The following external networks are identified [23]:

- **ITS Ad Hoc Network:** Enables direct communication between ITS stations without relying on a fixed infrastructure.
- **Access network:** This includes various types of networks such as ITS-specific access networks, public access networks, and private access networks.
- **Core network:** Typically represents broader networks like the Internet.
- **Internal network:** Additionally, an ITS station can have an internal network designed to interconnect the various components within the ITS station itself.

For practical applications, these networks are expected to support a range of use cases including road safety, traffic efficiency, infotainment, and business-related applications. However, it's important to note that a single network type might not be sufficient to meet the demands of all applications and use cases. Therefore, a hybrid approach that combines several ITS access and network technologies is often envisaged. Figure 8 [23] represents the highest level of abstraction of the ITS network architecture.

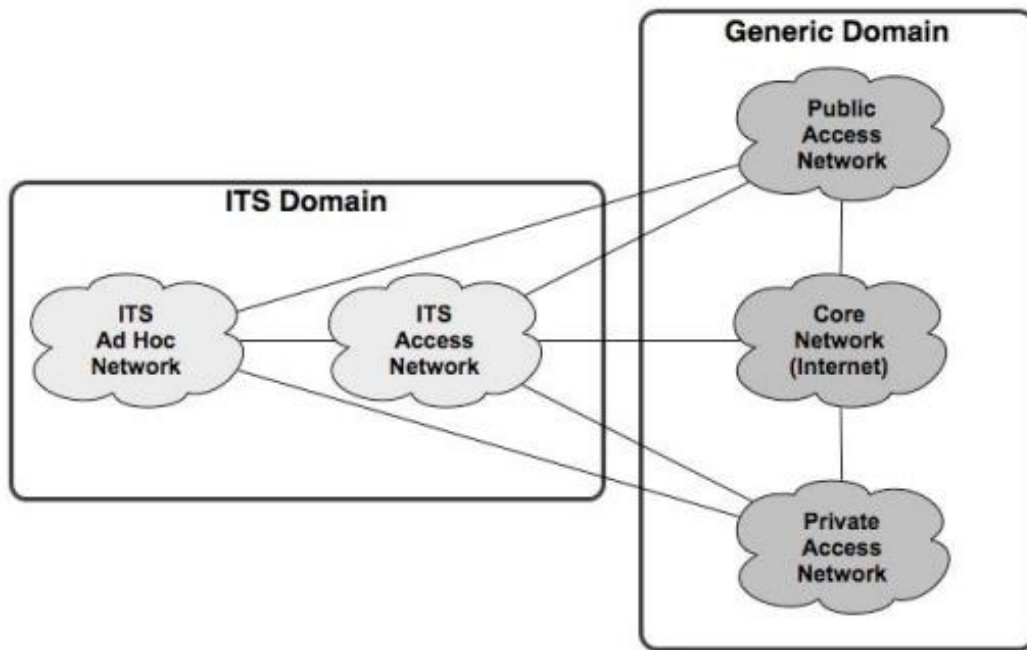


Figure 8. External networks involved in the ITS architecture and their interconnections

This layered approach to network architecture in ITS stations is crucial for handling the diverse and dynamic nature of vehicular communications. It allows for flexibility in adapting to various communication needs and scenarios, from high-speed vehicular communication to more static, infrastructure-based interactions. The integration of these diverse networks is key to the successful implementation and operation of advanced ITS applications.

The ITS network architecture encompasses various network elements essential for interconnecting ITS stations such as vehicle-based ITS stations (known as On-Board Units or OBUs), personal ITS stations, roadside ITS stations (known as Roadside Units or RSUs), and central ITS stations. This versatile architecture can adapt to a range of scenarios, tailored to meet specific economic considerations and regulatory requirements, and supports the progressive deployment of ITS services.

There are four primary deployment scenarios. The basic deployment scenarios can be expanded to provide hybrid deployment scenarios that integrate at least two deployment scenarios. These combinations also cover scenarios in which a network is simultaneously connected to multiple networks. Figure 9 represents an ad-hoc-centric abstraction of the ITS network architecture [23] composed of an ad-hoc network, access network, and core network. These networks are operated by a road operator, a telecom operator, or a national authority.



*Figure 9. High-level ITS network architecture: Ad hoc-centric*

In the context of ITS-G5, the ITS ad-hoc network uses ITS-G5 technology to enable direct communications between ITS stations. The ITS access network is a dedicated network that collects traffic from different ad hoc networks and provides access to specific ITS services and applications. It aims to interconnect RSUs and enable vehicle communications via the roadside infrastructure. The core network, characterized by its high-speed routers and switches, is tasked with directing and consolidating traffic towards a centralized ITS infrastructure, which houses servers for V2X applications. It is also responsible for providing connections to cloud services, car manufacturers' Original Equipment Manufacturer (OEM) systems, third-party networks, and public domains such as the Internet. This backbone is critical for the broad distribution of ITS services and for ensuring seamless integration across different network layers and providers.

## **2.4.2 ETSI Decentralized Congestion Control (DCC)**

In scenarios with high vehicle density, there is a significant increase in ITS-G5 traffic, primarily due to the transmission of Cooperative Awareness Messages (CAMs). These CAMs are the principal contributors to channel congestion issues [24].

In such scenarios, ETSI has implemented the Decentralized Congestion Control (DCC) mechanism. DCC adjusts traffic parameters to better manage channel load and improve data exchange quality [25]. DCC aims to ensure network stability, channel load limitation, and fair resource allocation. Figure 10 illustrates the ITS-G5 protocol stack along with the corresponding DCC functionalities [26].

Numerous DCC algorithms have been developed, and several are presently standardized [27]. The first class of algorithms is reactive DCC algorithms built on state machines. A vehicle continuously monitors channel occupancy and makes decisions about transmission parameters like message transmission rate (the maximum number of messages that can be transmitted per second), transmission power, etc. The second class of algorithms is adaptive algorithms inspired by the LIMERIC DCC adaptive algorithm [28]. An adaptive DCC algorithm is based on a set of parameters used to determine the optimal generation rate and control channel usage. The adaptive Dual  $\alpha$  algorithm outperforms reactive approaches and the ETSI adaptive algorithm [29].

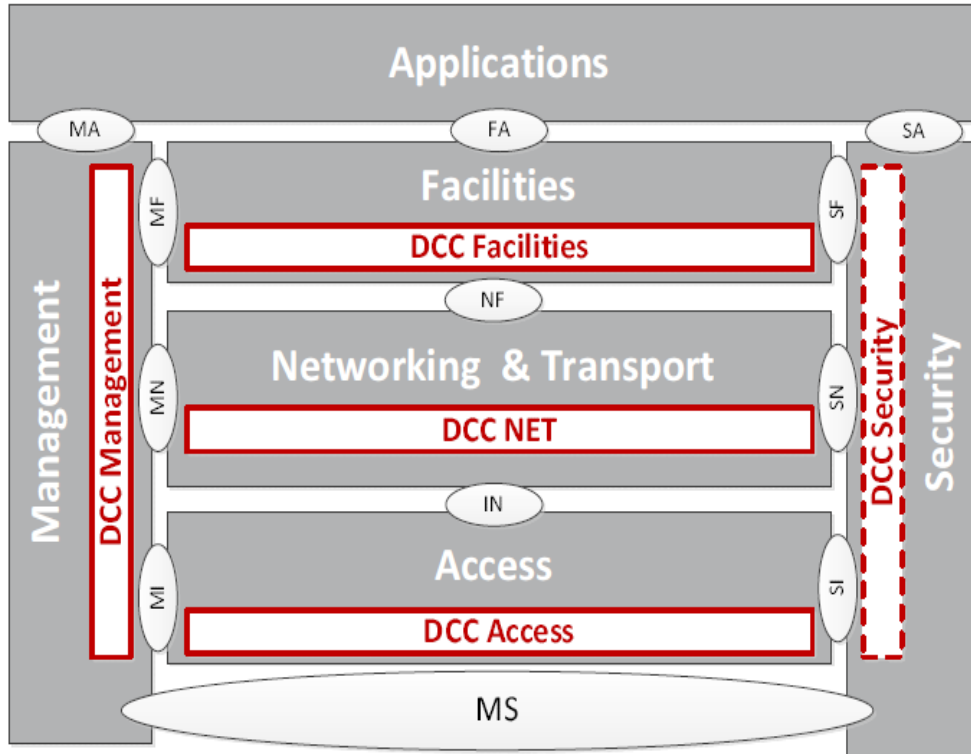


Figure 10. ITS-G5 protocol stack including DCC functionalities

In ITS-G5, resource management relies on DCC Profiles (DP), which are sets of transmission parameters designed to identify and regulate different traffic streams, like CAMs, DENMs, etc. These profiles are fundamental for differentiating traffic at the access, network, and transport layers [26]. Therefore, based on the DP assigned to a message and the current state of the transmission channel, one or more DCC algorithms can be applied to a traffic stream to help alleviate channel congestion.

DCC profiles in ITS-G5 are aligned with Enhanced Distributed Channel Access (EDCA) traffic classes (TCs), which play a crucial role in categorizing and prioritizing data traffic within the networking and transport layers. The MAC layer utilizes four distinct queues to prioritize this data traffic. Traffic classes with the highest priority are assigned shorter listening periods, known as the Arbitration Inter-Frame Space (AIFS), and more favourable Contention Window (CW) settings [30], which both allow high-priority access to the channel. Table 4 summarizes the default values for AIFS and CW for the different access categories [31].

Table 4: Default values for AIFS and CW at the access layer

AC	$CW_{\min}$	$CW_{\max}$	AIFS
Voice (VO)	3	7	58 $\mu$
Video (VI)	7	15	71 $\mu$
Best effort (BE)	15	1023	110 $\mu$
Background (BK)	15	1023	149 $\mu$

### 2.4.3 Transmit Rate Control (TRC) DCC

In the realm of vehicular networks, effective channel load management is pivotal for maintaining efficient communication. A variety of strategies have been developed for this purpose, with Transmit Rate Control (TRC) being one of the key techniques. TRC, along with other methods like Transmit Data Rate Control (TDC) and Transmit Power Control (TPC), plays a crucial role in regulating channel load. In simpler scenarios, a single strategy may suffice, but often, a combination of these techniques is employed for optimal results [33].

The TRC method, in particular, relies on packet timing thresholds to set the maximum message generation rate within a specified period. These thresholds are dependent on factors such as the chosen ITS-G5 sub-band in the 5.9 GHz spectrum and the selected transmit queue. Figure 11 illustrates a state-based DCC algorithm with multiple states ( $n > 1$ ), showcasing the flexibility and adaptability of TRC in different traffic conditions.

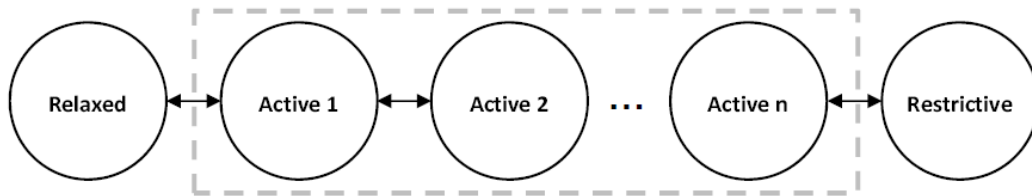


Figure 11. Example of state-based DCC algorithm

Table 5 provides a detailed overview of how the CBR values map to different states in the TRC algorithm, along with the corresponding allowed transmission rates. For instance, when the CBR is below 30%, the permitted packet transmission rate is 10 Hz (10 messages per second). As the CBR increases, indicating higher channel congestion, the TRC algorithm proportionally reduces the transmission rate to mitigate congestion.

Table 5: Packet rate settings for a 3-active states TRC

State	CBR	Packet transmission rate
Relaxed	<30%	10 Hz
Active 1	30% to 39%	5 Hz
Active 2	40% to 49%	2.5 Hz
Active 3	50% to 59%	2 Hz
Restricted	>60%	1 Hz

### 2.4.4 Dual $\alpha$ DCC

Adaptive approach mechanisms are linear control systems in which the process variable is the transmission rate, and the setpoint is a fraction of the channel capacity. An adaptive



congestion control strategy seeks to accomplish three goals: 1) Converging to a pre-set channel utilization level; 2) achieving local fairness among immediately surrounding vehicles; and 3) attaining global fairness among all vehicles contributing to congestion. The Linear Message Rate Integrated Control (LIMERIC) [32] algorithm serves as the foundation for the ETSI adaptive approach outlined in [33]. There have been several iterations of the LIMERIC algorithm, each using a different combination of parameter values. In this study, we focus on the Dual  $\alpha$  DCC method, which was first published in [29] as an improvement to the ETSI adaptive DCC algorithm. Compared to TRC, the Dual  $\alpha$  DCC algorithm achieves a faster convergence rate and fairness in transitory scenarios while converging to utilization that is close to the CBR target [28].

In the ETSI adaptive DCC, the ITS stations (ITS-Ss) calculate  $\delta$ : the maximum fraction of time an ITS-S is permitted to send messages in the channel, which is then used to decide how fast the ITS-S should send messages. An ITS-S measures the CBR, which is the fraction of time that the transmission channel is occupied, once every 100 milliseconds (i.e., 10 CBR values are measured per second). In the following step, the ITS-S calculates the appropriate value of  $\delta$  depending on the difference between the measured CBR and the Target CBR ( $CBR_t$ ). Following equation 1, the approach uses a smoothed version of the CBR that is calculated every 200 milliseconds following the equation 1 [29]:

$$CBR_s(n) = 0.5 \times CBR_s(n - 1) + 0.5 \times \frac{(CBR_m + CBR_{m-p})}{2} \quad (1)$$

where  $CBR_s(n)$  is the smoothed CBR for step  $n$ ,  $CBR_s(n - 1)$  is the previous  $CBR_s$ , and  $CBR_m$  and  $CBR_{m-p}$  are the last two measurements of the CBR (i.e., the newest measurements after the previous calculation of  $CBR_s$ ). Then, the ITS-S uses equation 2 to calculate  $\delta_{offset}$ :

$$\delta_{offset} = \begin{cases} \min\{\beta \times (CBR_t - CRR_s(n)), G_{max}^+\} & \text{if } CBR_t > CRR_s(n) \\ \max\{\beta \times (CBR_t - CRR_s(n)), G_{min}^-\} & \text{if } CBR_t \leq CRR_s(n) \end{cases} \quad (2)$$

where  $\beta$ ,  $G_{max}^+$ , and  $G_{min}^-$  are parameters of the algorithm.  $G_{max}^+$  and  $G_{min}^-$  are meant to limit the maximum variability of  $\delta_{offset}$  per step of the algorithm (i.e., to improve stability).  $\delta_{offset}$  represents the needed modification of the  $\delta$  to keep the CBR at the  $CBR_t$  value. The ITS station calculates the  $\delta$  at step  $n$ ,  $\delta(n)$ , every 200ms using equations (3):

$$\delta(n) = (1 - \alpha) \times \delta(n - 1) + \delta_{offset} \quad (3)$$

And if  $\delta(n) > \delta_{max}$ ,  $\delta(n) = \delta_{max}$ .

Else  $\delta(n) < \delta_{min} = \delta_{min}$ .

The value used for  $\alpha$  in equation (3) at step  $n$  is calculated using equation (4):

$$\alpha = \begin{cases} \alpha_{high} & \text{if } (\delta(n-1) - \delta_{\alpha_{low}}(n)) > th \\ \alpha_{low} & \text{otherwise} \end{cases} \quad (4)$$

where  $th$  is a threshold with a heuristic value to improve stability, more details are available in [29]. Finally, time intervals, called  $t_{go}$ , between successive transmissions are calculated. The values of  $th$ ,  $\alpha_{high}$ , and  $\alpha_{low}$  are stated in Table 6 [28]. The  $t_{go}$  is limited to a minimum value of 25ms and a maximum value of 1s, and is defined in equation 5 [28]:

$$t_{go} = t_{pg} + \min\left(\max\left(\frac{t_{onpp}}{\delta}, 25ms\right), 1s\right) \quad (5)$$

where  $t_{pg}$  is the time when the last transmission ended, and  $t_{onpp}$  is the value for the transmit duration of the last packet.

Table 6: Parameter values of the Dual  $\alpha$  algorithm

$th$	0.016
$\alpha_{high}$	0.1
$\alpha_{low}$	0.0006

## 2.5 Network Slicing

Network slicing is an advanced network architecture concept that facilitates the partitioning of a single, physical telecommunications network into several unique and independent virtual networks, commonly known as "slices". Each slice is uniquely engineered to deliver specific network services and functionalities, catering to a variety of distinct requirements [34]. This concept plays a crucial role in the advancement of next-generation wireless networks, especially within the realm of 5G architectures.

In the context of network slicing, there are two primary methodologies: vertical and horizontal slicing [35][36]. Vertical network slicing segments a network into various slices, with each slice being specifically tailored and optimized for a unique service or application. In contrast, horizontal network slicing enables the sharing of resources among nodes and network devices. These two approaches are not mutually exclusive and can be employed simultaneously, complementing each other effectively.

The implementation of network slicing, particularly when leveraging a shared infrastructure to support a diverse array of smart applications, involves significant complexities due to the involvement of numerous stakeholders [37]. These include entities in edge computing, cloud computing, telecommunications operations, and IoT networks. The orchestration of network slicing necessitates intricate coordination among these various players, posing a notable

challenge. Thus, a thorough examination of the network slicing architecture is essential [38]. The architecture of network slicing typically encompasses three distinct layers: the Physical Network Infrastructure Layer, the Network Slice Instance Layer, and the Service Instance Layer. Figure 12 outlines these three layers, and also illustrates a range of smart services enabled by network slicing.

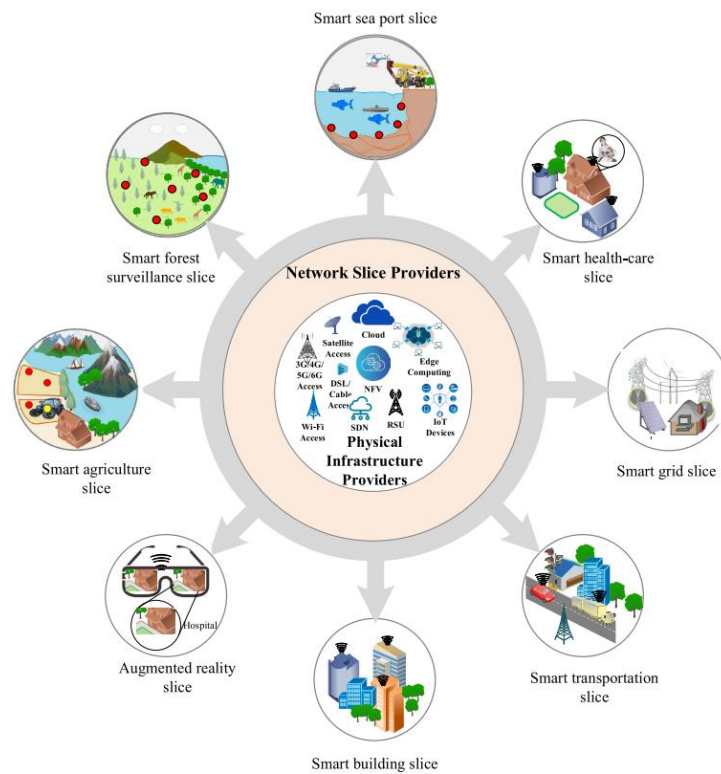


Figure 12. An overview of network slicing in enabling smart services

## 2.5.1 Key Aspects

Key principles in the design of network slicing include [39]-[43]:

- **Software-Driven:** The creation and management of network slices are driven by software controls, which orchestrate and automate the process using advanced algorithms, artificial intelligence, and policy-driven frameworks. This is enabled by underlying technologies such as Software-Defined Networking (SDN) and Network Functions Virtualization (NFV).
- **End-to-End Virtualization:** Network slicing capitalizes on end-to-end virtualization, meaning that every segment of the network from the device to the core can be virtualized and dedicated to a particular slice.
- **Customization and Flexibility:** Network slicing allows for the creation of tailored network environments to meet specific requirements of different services or applications. It offers the flexibility to dynamically adjust network resources and capabilities based on demand.

- **Resource Optimization:** Efficient utilization of physical network resources is a core aspect, enabling multiple virtual networks to coexist on the same infrastructure. It optimizes network capacity and performance and reduces operational costs.
- **Scalability:** Network slices can be scaled up or down according to the needs of the application, user, device, or available resources, supporting everything from individual IoT devices to city-wide network deployments.
- **Isolation:** Slices are logically isolated from one another, even though they run over the same physical infrastructure. This isolation is essential for security and performance reasons, as it ensures that the operations in one slice do not adversely affect another. In 802.11 environments, there are some strategies used for resource isolation, such as EDCA (Enhanced Distributed Channel Access) control, slice scheduling, and traffic shaping [44].
- **Security:** Each slice has its own distinct security protocols, ensuring enhanced data protection and preventing breaches in one slice from affecting others. This approach provides robust, customized security for each virtual network segment within a shared infrastructure.
- **End-to-End Management and Orchestration:** Covers the entire network, from the core to the edge, including RAN (Radio Access Network) and transport segments. Facilitates seamless management and orchestration across different network domains.

## 2.5.2 Slicing Scope in 5G Networks

5G network slicing includes slicing the RAN, core network, and even end-user devices [45].

### 2.5.2.1 RAN Slicing

RAN slicing involves customizing and sharing resources efficiently within the limited frequency spectrum. Key requirements include [40]:

- **Efficient Resource Management:** Essential for flexible MAC scheduling and meeting specific Key Performance Indicators (KPIs) for different types of slices, such as enhanced Mobile BroadBand (eMBB) and Ultra Reliable Low Latency Communications (URLLC).
- **Resource Sharing and Isolation:** Critical for creating logically isolated networks within the RAN, balancing the need for spectrum isolation against the limitations it imposes on multiplexing techniques.
- **Functional Requirements:** Each slice may require a unique functional split of the control/user plane and distinct Virtual Network Function (VNF) placement.

In the domain of RAN slicing, Software Defined RAN (SD-RAN) brings programmability to the forefront, allowing for enhanced control and management of resources. Platforms such

as SoftRAN [46] and FlexRAN [47] exemplify this by offering open APIs and a centralized control plane, thereby facilitating more efficient and dynamic management of RAN resources. Complementing this, the Cloud RAN (C-RAN) concept revolutionizes RAN virtualization by enabling the virtualization of RAN functions at cloud infrastructures [40]. This approach offers various functional split options, including PHY-layer, MAC-layer, RLC-layer, and PDCP-layer, which can be selected based on specific network requirements such as latency and data rate, further optimizing network performance and flexibility.

### **2.5.2.2 Core Network Slicing**

Core network slicing in 5G refers to the strategic division and optimization of the mobile core network, transitioning from a full IP core network to a more virtualized and software-oriented architecture. This evolution is significantly driven by technologies such as SDN and NFV, which introduce greater flexibility and elasticity into Evolved Packet Core (EPC) networks. The 3GPP has restructured the core network into a modular architecture, dividing main EPC elements into granular network functions and enabling the creation of dedicated core network instances for diverse services.

EPC Virtualization and Dedicated Core Networks (DCN) are integral components of this slicing strategy. Virtualization allows for the main core network entities, such as Mobility Management Entity (MME), Home Subscriber Server (HSS), Packet data network GateWay (PGW), Serving GateWay (SGW), and Policy and Charging Rules Function (PCRF), to be deployed on virtual platforms, offering more adaptable, elastic service provisioning and quality of service assurance. This enables mobile network operators to simultaneously deploy multiple EPC instances, tailored to specific user requirements. The DCN concept introduced by 3GPP in Release 13 further supports network slicing by allowing eNBs to select appropriate core network functions for specific user equipment in both control and user planes.

The Next Generation 5G Core network, another key aspect of core network slicing, is characterized by its enhanced elasticity, flexibility, and scalability. This new architecture disaggregates the EPC functions into more detailed network functions, supporting a mix of LTE Network Functions (NFs) and new ones such as Access and Mobility Management Function (AMF), Session Management Function (SMF), Policy Control Function (PCF), User Plane Function (UPF), Unified Data Management (UDM), and NF Repository Function (NRF). It involves two primary deployment phases: the initial phase based on point-to-point connections and the subsequent phase that adopts a service-oriented architecture. This service-oriented approach allows NFs to interact more dynamically, querying an NRF to discover and communicate with each other, significantly enhancing the flexibility and scalability of the core network and enabling it to be tailored specifically to different network slices.

### 2.5.2.3 User Device Slicing

User Device (UD) Slicing in 5G networks represents a significant shift from the approach taken in 4G networks. In 4G, User Equipment (UEs) such as smartphones, tablets, and laptops are not differentiated based on their service demands and functional requirements; the network treats all UEs uniformly. However, 5G networks are designed to recognize and handle UEs based on their distinct characteristics or usage-class types. This means that instead of connecting every UE to a generic, one-size-fits-all network, 5G networks connect UEs to customized network slices specifically created for their type. For example, a UE categorized under the CriC (critical communications) usage-class type would connect to a CriC-specific network slice, while a UE for V2X communications would link to a V2X slice, ensuring a high degree of QoS tailored to their needs.

Furthermore, recent advancements in 5G technology have introduced a novel concept of slicing within the UE itself, especially for portable and smart devices. This concept, known as UE slicing, envisions treating these devices as platforms with pre-installed middleware, similar to a hypervisor in laptops, which can manage and allocate resources across multiple mobile operating systems (OS). These OSs operate within logical container partitions set up by the middleware, effectively managing resources among them. This setup allows the UE to function as a platform where different OSs, each with its unique set of applications and features, run concurrently as separate slices. Such an arrangement offers users a broader range of applications and greater customization, reflecting the versatility and user-centric focus of 5G network technology.

### 2.5.3 Network Slice Instance Lifecycle Management

The 3GPP study document [48] introduces a crucial distinction in the lifecycle management of Network Slice Instances (NSIs) by decoupling them from the service instances that utilize them. This separation enhances scalability by allowing network slices to be provisioned independently of the service instances, promoting efficient sharing across multiple services. To manage an NSI effectively, the following management procedures are essential:

1. **Fault Management:** This involves identifying, diagnosing, and rectifying faults within the NSI to ensure continuous and reliable service.
2. **Performance Management:** Monitoring and optimizing the performance of the NSI to meet predefined Key Performance Indicators (KPIs) and service level agreements (SLAs).
3. **Configuration Management:** Managing the settings and configurations of the NSI to adapt to changing network demands and service requirements.
4. **Policy Management:** Implementing and overseeing policies that govern the operation and utilization of the NSI, ensuring compliance with network standards and business objectives.

In this framework, service management is conducted within the service provider’s domain, while the management of network slices falls under the purview of the network operator. This arrangement necessitates a business-to-business interface, such as an SLA, which outlines the terms of the relationship between the service provider and the network operator. Under this business relationship, the network operator can offer varying levels of control to the service provider. These levels range from basic monitoring to limited control, where the service provider can compose slices from a catalogue to extended control, enabling the service provider to instantiate its Virtual Network Functions (VNFs) and Management and Network Orchestration (MANO) stack. This tiered approach to control and management allows for flexibility and customization in the deployment and operation of network services.

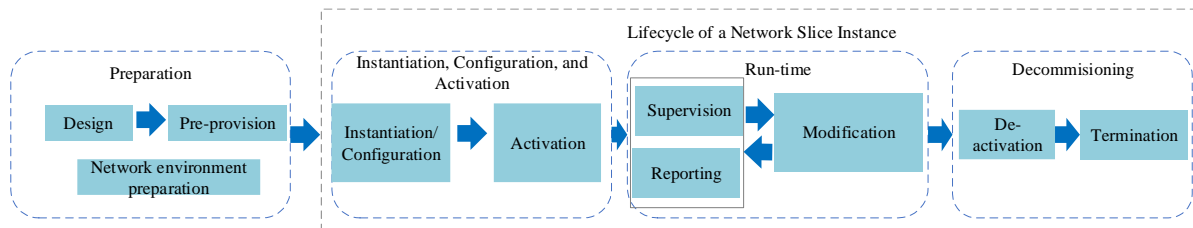


Figure 13. Life-cycle management of a network slice instance.

Network Slice Instance (NSI) lifecycle management in 5G networks involves several distinct phases (Figure 13 [48]) each with specific tasks and objectives:

- **Preparation Phase:** Before the NSI exists, this phase encompasses the creation and verification of network slice templates, onboarding these templates, and preparing the necessary network environment. These preparatory actions support the lifecycle of NSIs and any other required network preparations.
- **Instantiation, Configuration, and Activation Phase:** This phase involves creating and configuring all resources (shared or dedicated) for the NSI, bringing it to a state ready for operation. Activation includes actions like diverting traffic to the NSI and provisioning databases. It may also include the instantiation, configuration, and activation of other shared or non-shared network functions.
- **Run-Time Phase:** During this phase, the NSI is active and capable of handling traffic to support communication services of specific types. It includes supervision/reporting for KPI monitoring and tasks related to modification, such as upgrades, reconfiguration, scaling, capacity changes, and modifications of the NSI topology, including association and disassociation of network functions with the NSI.
- **Decommissioning Phase:** This final phase involves deactivating the NSI (taking it out of active duty) and reclaiming dedicated resources, such as terminating or reusing network

functions, and configuring shared or dependent resources. After decommissioning, the NSI ceases to exist.

Each phase of the NSI lifecycle entails high-level tasks and requires appropriate verification of the output of each task within the phase, ensuring that the NSI meets its intended operational goals effectively.

#### 2.5.4 Network Slicing for V2X Applications

In the realm of V2X, network slicing presents several advantages:

1. **Priority Handling:** Network slices can prioritize V2X messages that are critical for road safety over less time-sensitive data.
2. **Resource Allocation:** Slices can ensure that the necessary bandwidth is available for vehicle communications even when the network is under heavy load.
3. **Scalability:** As the number of connected vehicles grows, network slices can be dynamically adjusted to scale the network resources up or down as needed.
4. **Security and Isolation:** Slices dedicated to V2X can be secured with specific protocols and isolated from other network traffic to protect sensitive vehicular data.

Generally, network slicing can be a beneficial technology for V2X applications, since vehicular technology involves multiple use cases, traffic types, and communication paths. Communication between vehicles in proximity may involve a slice using the New Radio Side-Links NR-SL (PC5) that can be used for direct V2V, whereas other network slices can offer communication services for V2N links through the edge network.

End-to-end network slicing can enable Public Land Mobile Network (PLMN) operators or road operators to provide a differentiated level of V2X services to different classes of vehicular applications or driving use cases, through appropriately instantiating network slices per each application or use case, as well as isolating their traffic flows, and enabling multi-tenancy. Of course, each different slice may be assigned and managed by a specific tenant, or on behalf of it, given that a tenant may be a PLMN operator, a service provider, or a third company that may desire to rent a slice to offer tailored services to its clients. Consequently, the guidelines stated by [35] and [49] for network slicing can be put into the context of V2X types of applications and services.

In 3GPP network slicing, PLMN operators and V2X service operators should be able to:



- Support providing connectivity to home and roaming vehicles in the same network slice and allow a vehicle to be simultaneously assigned and access services from one or more network slices of one operator.
- The V2X network slice operator should be able to create, modify, and delete a network slice; define and update the set of V2X services and capabilities supported in a network slice, and configure the information that associates a vehicle to a network slice as well.
- The V2X network slice operator should be able to assign, move, and terminate a vehicle to or from a network slice based on service subscription, the vehicle's capabilities, access technology used by the vehicle, and the operator's policies and services provided by their designated network slice.
- The V2X network slice operator should be able to scale up or down (i.e., vertically or horizontally, on virtual and physical networks) their network slices' capacities; define a priority order between different network slices in case multiple network slices compete for resources on the same network.

## 2.6 Edge Computing

The advent of advanced vehicular services, such as autonomous driving, real-time traffic management, and in-vehicle infotainment systems, has significantly increased the demand for processing power and data storage in the context of Vehicle-to-Everything (V2X) communications. These services rely heavily on rapid data processing and decision-making capabilities to function efficiently. Autonomous driving, in particular, requires the real-time analysis of vast amounts of sensor data to make split-second decisions, while traffic management systems need to process and interpret large streams of traffic data to optimize flow and safety. Similarly, infotainment systems demand substantial storage and processing resources to deliver rich multimedia content seamlessly.

Traditionally, the solution to these demands has been to offload these tasks to the cloud, where powerful computing resources handle the processing and storage remotely. This approach, while effective in managing the computational load, introduces a significant challenge: latency. In V2X applications, where milliseconds can mean the difference between safe operation and potential hazards, the latency involved in transmitting data to and from a centralized cloud can be a critical bottleneck.

This challenge is adeptly addressed by Multi-access Edge Computing (MEC), formerly referred to as Mobile Edge Computing, which situates computing power and data storage at the network's edge, closer to where data is generated and consumed in vehicular environments. MEC's proximity significantly reduces latency, enabling autonomous vehicles to process real-

time data swiftly, allowing traffic management systems to react promptly, and ensuring seamless delivery of content by infotainment systems.

Moreover, a notable benefit of MEC in this context is its energy efficiency. Implementing MEC can decrease energy consumption by up to 40% [50], making it a much more sustainable option compared to accessing remote network clouds, which can consume up to five times more energy [51]. Thus, MEC not only meets the intense processing and storage demands of contemporary V2X services and adheres to their strict latency requirements, but also presents a more energy-efficient solution, aligning with the growing emphasis on sustainability in vehicular technology.

### **2.6.1 Introduction to MEC**

Multi-access Edge Computing (MEC) represents a paradigm shift in network architecture, where cloud computing capabilities and IT services are embedded within the RAN. This innovative concept, endorsed in clause 5.13 of [52], is designed to alleviate the core network from the burdens of data processing by redistributing these tasks to the network's edge, closer to where the demand originates [53]. Such a configuration promises a host of benefits, particularly for V2X applications and third-party services, by situating operational functions proximal to vehicles' access points, such as gNBs or RSUs.

The essence of MEC lies in its ability to minimize latency, a critical requirement for the burgeoning suite of V2X applications. With the 5G Core Network's ability to select a User Plane Function (UPF) near the vehicle, MEC facilitates traffic steering directly to local data networks where V2X application servers reside. This is achieved through the standardized 3GPP N6 interface, ensuring that data does not traverse unnecessary network hops, thus significantly reducing end-to-end latency and lightening the load on the transport network.

Within the MEC infrastructure, the 'edge' is conceptualized as the logical endpoint of a network. This could manifest as a base station, an access point, or a peripheral data centre, each serving as a pivotal node for data processing and service delivery. MEC's architecture is lauded for its flexibility and modularity, which enables support for an extensive array of services and applications. These range from content caching and Internet of Things (IoT) integration to location-based services, video analytics, environmental monitoring, and augmented reality solutions, among others.

The deployment of MEC is synergistically coupled with cutting-edge technologies like SDN and NFV [54]. These technologies impart the required dynamism and scalability, allowing MEC platforms to host a diverse spectrum of applications. This versatility opens the door to novel service offerings and monetization avenues for service providers, third parties, and network operators.

In the landscape of 5G, MEC's role is particularly pronounced in scenarios that demand swift data processing and ultra-low latency. V2X communications, smart cities, and industrial IoT stand to benefit immensely from MEC's capabilities. By handling vast data volumes at the network edge, MEC enables real-time analytics and decision-making processes, essential for the autonomous systems that drive these applications. This decentralized approach not only enhances the user experience by ensuring prompt service delivery but also contributes to the optimization of network resources, heralding a new era of efficient and responsive network architectures for the interconnected world of tomorrow.

### 2.6.2 MEC for V2X: Architectural Integration and Service Enhancement

The integration of MEC within V2X communications unlocks a myriad of opportunities, enhancing service delivery through distributed computing resources. This synergy becomes particularly significant in the architectural framework of vehicular networks, where MEC's alignment with the RAN enables a multifaceted service environment. The architecture comprises three distinct layers as depicted in Figure 14 [55]: The cloud layer with its centralized servers, the MEC layer with nodes such as Roadside Units (RSUs), and the user layer consisting of vehicular terminals. Each layer is instrumental in achieving the end-to-end service delivery objectives, contributing to a cohesive operational structure.

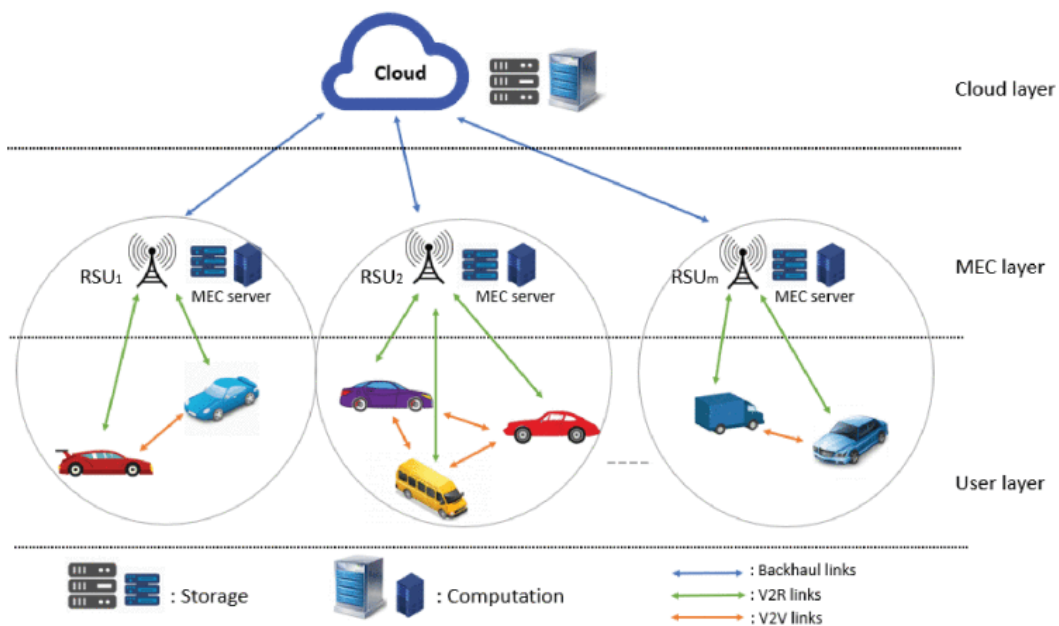


Figure 14. MEC-enabled vehicular network architecture

MEC serves as a pivotal force in several V2X application scenarios. For instance, path navigation benefits immensely from MEC's ability to process and store data collected from vehicles and neighbouring RSUs. This facilitates enhanced navigation accuracy and up-to-date route information. Traffic control is another domain where MEC's influence is transformative.

RSUs, equipped with MEC capabilities, collect, and analyse data, subsequently aiding in traffic management and reducing congestion. The advent of low latency services through MEC is particularly advantageous for autonomous driving systems, where the timely execution of tasks is paramount for safety and traffic flow efficiency.

MEC's computational prowess also lends itself to applications requiring intensive processing, such as Augmented Reality (AR) and facial recognition technologies. These services, traditionally reliant on centralized cloud resources, can now leverage edge computing for improved quality of service. Additionally, as vehicular networks evolve, data mining and aggregation become crucial in harnessing the wealth of information generated. MEC facilitates the deployment of Artificial Intelligence (AI) and Machine Learning (ML) algorithms, which can deeply analyse the data to enhance network performance and data efficiency.

Integrating MEC into V2X communications presents several challenges that are critical to address. Resource sharing fairness stands out, requiring a delicate balance to ensure that the computational and storage capabilities at the edge are distributed equitably among competing services and applications. Alongside this, optimizing edge node placement is a complex task, essential for maintaining efficient service delivery in the face of dynamic vehicular traffic patterns and fluctuating service demands.

Moreover, ensuring robust security and privacy for the data processed at the edge is paramount, given the sensitivity of vehicular data and the need for stringent adherence to diverse regulatory standards. Among these challenges, service migration is particularly significant. It involves the seamless transition of services as vehicles move through the network, a process that must be handled with minimal latency to uphold the continuous and reliable performance of safety-critical V2X applications. Given its complexity and importance, this aspect of service migration forms a key focus of one of our contributions.

## **2.7 Toward a New Technology for Vehicular Networks: IEEE 802.11bd**

IEEE 802.11bd, an advanced standard for Vehicular Ad-hoc Networks (VANETs), represents a significant upgrade from its predecessor, IEEE 802.11p. Developed under the IEEE Task Group since January 2019 [56], 802.11bd aims to enhance the Physical (PHY) and Medium Access Control (MAC) layers by incorporating advancements from existing IEEE 802.11 Wireless Local Area Network (WLAN) standards, such as IEEE 802.11ac and IEEE 802.11ax.

One of the primary objectives of IEEE 802.11bd is to provide an ad-hoc vehicular environment with substantial improvements in key areas such as throughput, latency, reliability,

and communication range [57]. According to its project authorization report, 802.11bd is expected to deliver performance that is double that of 802.11p [58], including twice the MAC throughput of 802.11p, support for relative velocities up to 500 km/hr, and a communication range that is twice as long as that of its predecessor.

Technically, IEEE 802.11bd diverges from 802.11p by utilizing a 20 MHz bandwidth channel for communication instead of the 10 MHz bandwidth used in 802.11p [59]. The modulation and coding scheme (MCS) of 802.11bd can reach up to 256-QAM [60], supported by multiple-input multiple-output (MIMO) antennas, thereby enabling very high throughput. To ensure reliable communication, 802.11bd incorporates the Low-Density Parity-Check (LDPC) coding mechanism and Midambles, which are similar to preambles but are used between OFDM data symbols to estimate channel variation. These Midambles help in reducing interference effects on message integrity.

Moreover, the MAC layer in IEEE 802.11bd is based on the Enhanced Distributed Channel Access (EDCA) method, which is essential for channel access. However, with a 20 MHz bandwidth and a 256-QAM MCS, 802.11bd allows for message retransmissions by sending each OFDM symbol over two different sub-carriers. This feature, combined with LDPC and Midambles, contributes significantly to improved reliability in message delivery compared to IEEE 802.11p.

In summary, IEEE 802.11bd represents a considerable leap forward in the technology available for V2X communication. It offers higher performance, better reliability, and an extended range [61], which are essential for meeting the increasingly demanding requirements of modern vehicular applications. These enhancements, particularly in terms of throughput and reliability, make IEEE 802.11bd a pivotal standard in the realm of V2X communications, paving the way for more advanced and reliable vehicular communication systems.

## 2.8 Summary of Key Features of IEEE 802.11, LTE-V2X and 5G-V2X

In this section,

Table 7 provides a comparative summary of V2X technologies, highlighting the evolution and distinguishing features of IEEE 802.11-based standards and Cellular V2X (C-V2X) technologies encompassing both LTE-V2X and 5G V2X.

*Table 7: Summary of V2X communication technologies features*

<b>Radio Design</b>	<b>802.11p</b>	<b>802.11bd</b>	<b>LTE-V2X Rel 14/15</b>	<b>5G NR Rel 16</b>
<i>Radio bands</i>	5.9 GHz	5.9 GHz, 60 GHz [62]	5.9 GHz (PC5 interface)	5.9 GHz~52.6 GHz including mmWave
<i>Subcarrier spacing</i>	156.25 KHz	312.5 KHz, 156.25 KHz, 78.15 KHz	15 KHz	Sub-6 GHz: 15, 30, 60 KHz mmWave: 60, 120 KHz
<i>Synchronization</i>	Asynchronous	Asynchronous	Synchronous	Synchronous
<i>Channel size</i>	10/20 MHz	20 MHz	10/20 MHz	10/20 MHz and wideband (40/60/80/100 MHz)
<i>Resource multiplexing</i>	CSMA/CA	CSMA/CA	TDMA/FDMA	TDMA/FDMA
<i>HARQ retransmission</i>	No	No	Yes	Yes
<i>Waveform</i>	OFDM	OFDM	SC-FDMA	SC-FDMA, OFDMA + other options
<i>Resource selection</i>	Listen-Before-Talk	Listen-Before-Talk on the first Channel (first 10 MHz)	SPS + Listen-Before-Talk	Many available options
<i>MIMO</i>	Not supported	Up to 8 RX/TX antennas	2 RX/TX antennas	Up to 8 RX/TX antennas
<i>Modulation</i>	Up to 64 QAM	Up to 256 QAM	Up to 64 QAM	Up to 256 QAM
<i>Data channel coding</i>	BCC	LDPC	Turbo	LDPC

# 3. Proactive C-ITS Decentralized Congestion Control Using LSTM

## 3.1 Introduction

The evolution of vehicular networks has been a cornerstone in the advancement of modern transportation systems. With the rise of Cooperative Intelligent Transport Systems (C-ITS), the integration of communication technologies into vehicular environments has become increasingly critical. These systems, which encompass a wide range of applications from traffic management to safety measures, rely heavily on the seamless and efficient exchange of information.

As vehicular networks become more complex and heavily trafficked, managing the flow of this communication becomes a paramount concern. The challenge is not just in facilitating the exchange of data but in doing so in a manner that is both efficient and reliable, even under the strain of high vehicle densities and rapidly changing network conditions. The effectiveness of these communication systems is vital, not only for the smooth operation of transport networks but also for ensuring the safety and well-being of road users. This is where the issue of channel congestion in vehicular networks, a critical factor in the performance of C-ITS, comes into play.

This chapter investigates the use of LSTM neural networks for proactive congestion control in C-ITSs. It highlights a shift towards predictive management and allocation of network resources for better stability and increased fairness, showcasing the superior performance of this method in reducing channel load and enhancing channel convergence in vehicular networks.

### 3.1.1 Problem Statement

Channel congestion C-ITS poses a critical challenge. The efficacy of V2X communication is primordial in these systems, underpinning safety, and traffic management protocols. The European standard ITS-G5, tailored for V2X communications in Ad hoc Vehicular Networks (VANETs), adopts DCC strategies to counteract channel congestion. However, the prevailing DCC methodologies, such as reactive Transmit Rate Control (TRC) and adaptive Dual  $\alpha$  algorithms, tend to address congestion post-occurrence. This delay in response adversely affects vehicular network performance, leading to inefficiencies in traffic flow and potential safety hazards.

The reactive nature of existing DCC mechanisms is particularly problematic in highly dynamic and high-density traffic scenarios, where the increased number of vehicles exacerbates

channel load and congestion. The communication channel becomes saturated in such environments, leading to data transmission delays and packet losses. This scenario undermines the reliability of safety-critical applications of C-ITS, such as collision avoidance systems and emergency vehicle notifications. The reliance on legacy DCC techniques also means that the network is constantly playing 'catch-up' with the changing traffic conditions, which is far from ideal in the fast-paced, dynamic environment of vehicular networks. As a result, there is an urgent need to shift from a reactive to a proactive approach in managing channel congestion.

### **3.1.2 Research Objectives**

In this research, we introduce proactive DCC algorithm, harnessing the predictive capabilities of the Long Short-Term Memory (LSTM) recurrent neural network. This approach is designed to forecast the Channel Busy Ratio (CBR) across multiple time steps, thereby enhancing the ability of the vehicular network to anticipate and manage congestion. By shifting from reactive to proactive management, our algorithm aims to respond more effectively to the dynamic conditions of vehicular networks.

The process involves generating precise forecasts of the CBR, which are further refined using the Exponentially Weighted Moving Average (EWMA). This step ensures a smoother and more accurate representation of network conditions. These enhanced CBR values are crucial for the DCC algorithms, as they are the key parameter in the decision-making process regarding the transmission configurations of vehicles. Through this method, we aim to significantly enhance the efficiency of vehicular networks.

The primary goal of our research is to significantly enhance channel load convergence and network stability, while simultaneously reducing overall channel load. This advanced approach is poised to substantially increase vehicular communication efficiency, optimizing resource allocation and mitigating channel exhaustion risks. In doing so, it promises to elevate fairness in resource distribution among vehicles. Our focus spans both local and global scales, aiming to refine vehicular communication within discrete clusters of vehicles and across the broader network. Ultimately, this research contributes meaningfully to C-ITS, striving to establish a more responsive, efficient, and equitable vehicular communication landscape.

## **3.2 Related Works**

Extensive research has been conducted on ITS-G5 and its associated algorithms, leading to significant insights in the field of vehicular networks. The study by Rostami et al. [63] provided a thorough simulation-based comparison of the DCC TRC algorithm against LIMERIC, demonstrating LIMERIC's enhanced efficiency in managing inter-packet gaps. Another



important evaluation in [64] focused on optimizing the DCC TRC reactive algorithm, exploring various approaches to refine its operational effectiveness.

The introduction of the Dual  $\alpha$  algorithm, as mentioned before in [29], marked a notable advancement in the ETSI DCC adaptive algorithm. Although it didn't surpass the original LIMERIC algorithm in overall performance, it showed notable improvements in aspects of convergence and fairness, especially under transient conditions. The research in [65] tackled the challenge of unfairness in DCC, proposing a new mechanism aimed at achieving a more balanced distribution of network resources.

Comparative studies, such as the work by Bansal et al. [66], have also been instrumental in understanding these algorithms' efficacy. This particular study compared LIMERIC with the DCC TRC reactive algorithm, finding that LIMERIC generally resulted in lower reception intervals and reduced tracking error. Finally, the work in [67] proposed a short-term density prediction approach based on an algorithmic scheme to provide more accurate and up-to-date information about the network state. As a result, this algorithm improves the adaptation of the transmission parameters to achieve better overall network performance.

The performance comparison between ITS-G5 and LTE-V2X communication technologies has also been a focal point of research. In [68], the authors conducted a comprehensive system-level simulation comparing ITS-G5 and LTE-V2X mode 4. The study focused on the Packet Delivery Ratio (PDR) in relation to the Signal to Interference plus Noise Ratio (SINR), where LTE-VTX showed superior PDR performance, albeit with ITS-G5 having the advantage of lower communication latency. Similarly, the research by Karoui et al. [69] evaluated both ITS-G5 and LTE-V2X mode 3, comparing their performance in terms of end-to-end delay and radio frequency conditions. The results concluded that while ITS-G5 had a shorter E2E delay, it was less reliable than LTE-V2X mode 3. Moreover, the study in [70] delved into the performance degradation due to the co-channel coexistence of LTE-V2X and ITS-G5, examining various mechanisms to alleviate this loss.

Despite these extensive studies, there remains a significant gap in the application of Artificial Intelligence (AI) models for proactive DCC. This unexplored area is where our current research is positioned, aiming to harness the power of AI to develop a more effective, proactive approach to DCC in vehicular networks. By integrating AI techniques, particularly advanced machine learning models, we seek to anticipate and manage network congestion more effectively than traditional methods. This approach not only promises to enhance network efficiency but also to revolutionize the way vehicular networks adapt to dynamic traffic conditions and evolving communication demands.

### 3.3 Proactive DCC Approaches Based on LSTM Recurrent Neural Networks

This section delves into our innovative proactive TRC DCC and proactive Dual  $\alpha$  algorithms, which integrate the capabilities of Long Short-Term Memory (LSTM) networks. These advanced algorithms are evolved from the legacy reactive TRC DCC and adaptive Dual  $\alpha$  algorithms, respectively. By incorporating LSTM, a sophisticated artificial recurrent neural network (RNN), we enhance these algorithms for more effective performance in diverse scenarios.

LSTM is an artificial RNN architecture used in deep learning for a variety of applications, including classification, processing, and prediction. An LSTM network is composed of several blocks (refer to Figure 15 [71] for visualization). The key components of an LSTM block include:

- Cell state: This component serves as the network's memory, allowing information to flow through the LSTM. It updates the new cell state ( $C_t$ ) based on the previous state ( $C_{t-1}$ ), ensuring continuity and memory retention over time.
- Forget gate: Responsible for filtering out unnecessary information, the forget gate selectively removes data from the cell state, streamlining the LSTM's processing capabilities.
- Input gate: This gate plays a crucial role in determining which new information is significant enough to be stored in the cell state, enhancing the network's learning accuracy.
- Output gate: Drawing from the current cell state and the previous hidden state, the output gate calculates the next hidden state. This hidden state is crucial for predictions and decision-making processes in the LSTM network.

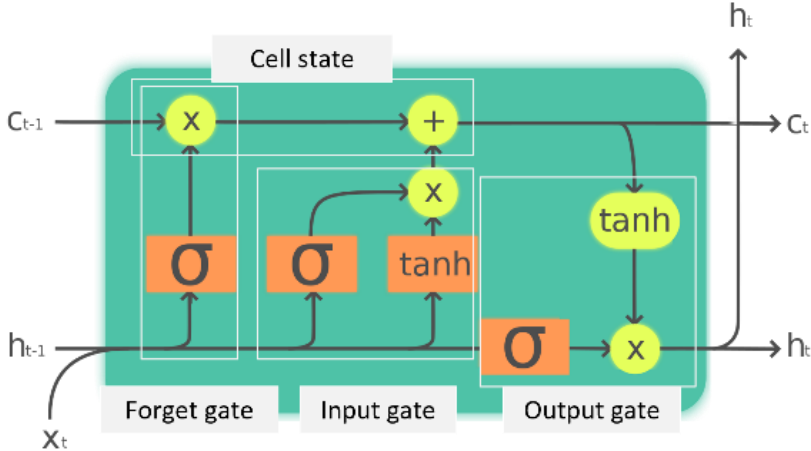


Figure 15. LSTM block components

Multi-step time series forecasting is the term used to describe a time series forecasting problem that requires the prediction of multiple time steps into the future. Specifically, these are issues in which the prediction horizon or interval is longer than a single step. When it comes to multi-step forecasting, two types of LSTM models can be used: The Vector Output Model and the Encoder-Decoder Model. In our case, we use the former, where the LSTM will directly output a vector that can be interpreted as a multi-step forecast.

In the context of our proactive DCC algorithms, we leverage the LSTM network for both the training and prediction phases, focusing on the CBR time series. The CBR is a critical determinant in controlling the message generation rate across all DCC algorithms. By utilizing LSTM's predictive power, we can forecast smoothed CBR values, significantly reducing the likelihood of out-of-control scenarios in network traffic. This proactive approach enables the algorithms to anticipate and adapt to varying network conditions more effectively, enhancing overall communication efficiency.

The LSTM agent operates by analysing the measured CBR values ( $CBR_m$ ) every second, using data from the last  $N$  seconds ( $N \times 10$  values) to predict the future CBR values ( $CBR_p$ ) for the upcoming  $M$  seconds ( $M \times 10$  values). In our methodology, only the last set of predicted values (the last ten CBR values) from each prediction cycle is used for the subsequent second decision-making process. To further refine these predictions, we apply the Exponentially Weighted Moving Average (EWMA) formula, calculating the smoothed predicted CBR ( $CBR_{sp}$ ) at each time step  $t$ , as defined in equation (6), where  $t_0$  marks the start of prediction generation and  $\theta$  is the smoothing factor:

$$CBR_{sp}(t) = \begin{cases} CBR_m(t), & \text{if } t < t_0 \\ (1 - \theta) \times CBR_{sp}(t - 1) + \theta \times CBR_p(t), & \text{else} \end{cases} \quad (6)$$

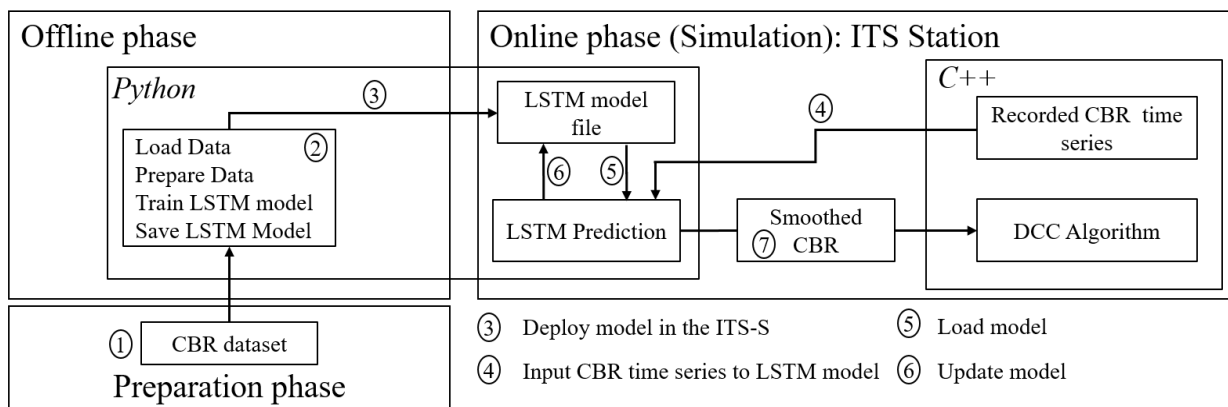


Figure 16. ML modelling for proactive DCC algorithms

The system architecture consists of the 3 main phases as depicted in Figure 16:

1) Preparation phase: The foundation of our system begins in this phase, where simulations are conducted using the conventional DCC reactive and adaptive algorithms, namely TRC and Dual  $\alpha$ . The primary objective here is to generate a CBR dataset (step 1), which forms the backbone of our LSTM training process. This step involves a meticulous collection of CBR values from every ITS station engaged in the simulation, ensuring a robust and diverse dataset that reflects a wide range of network conditions.

2) Offline phase: Transitioning to the offline phase, the focus shifts to transforming the accumulated dataset into usable samples. This process, depicted in step 2, involves reshaping the data into segments of specified lengths and adapting it for effective LSTM processing. A specialized function is utilized to partition the univariate time series into samples with several input (N) and output (M) time steps. Following this segmentation, we embark on defining the LSTM model's configuration – this includes setting parameters such as the number of layers, the number of cells per layer, and the choice of activation function. Once configured, the model undergoes thorough training, ending in the saving of the trained LSTM model for deployment in the subsequent phase.

3) Online phase: The culmination of our system's architecture is the online phase. Here, the LSTM model, freshly deployed at each ITS station after the offline phase (step 3), springs into action. It utilizes the CBR data from the preceding N seconds ( $N \times 10$  values) to generate predictions for the upcoming M seconds ( $CBR_p$ ). In this real-time application, an ITS station emits a vector of CBR values every second. This vector then serves as input to the LSTM agent (step 4), which is adept at generating accurate CBR predictions for the immediate future. Post prediction, the LSTM model undergoes an update process (step 6). The freshly obtained CBR predictions are then subjected to a smoothing process to calculate  $CBR_{sp}$ . This smoothed data is subsequently fed into the DCC algorithms (step 7). It's at this juncture that the DCC algorithms, now armed with predictive data, make informed decisions about the rate at which new messages should be generated.

A critical aspect of our approach is the processing time needed by the ITS station to generate a sequence of predictions. We propose incorporating predictions on a per-second basis, necessitating that the LSTM's prediction duration be less than one second. Achieving this rapid prediction rate could be effectively facilitated through the deployment of a dedicated onboard processing unit. This unit would be specifically tailored to handle the computational demands of the LSTM, ensuring the timely generation of accurate traffic predictions.

In our first approach, we applied predictive techniques to enhance the TRC mechanism. This adaptation aims to accelerate transitions between different channel states, both in increasing and decreasing density scenarios. Such quickened state transitions are vital in mitigating

channel overloads, an issue that intensifies with increasing vehicle density, and optimizing resource utilisation. By utilizing predictions to adjust message generation intervals more rapidly, the TRC system is better equipped to handle the growing demands on network capacity. The outcome of this integration is a proactive TRC algorithm, thoughtfully designed to include three active states, highlighted as follows:

Our second approach introduces the proactive Dual  $\alpha$  algorithm, an iteration of the existing adaptive Dual  $\alpha$  algorithm. The primary distinction in our proactive version is the application of smoothed predicted CBR values ( $CBR_{sp}$ ) in equation (1), rather than the traditionally measured CBR values. This adaptation facilitates swifter transitions to higher  $t_{go}$  values, thereby promptly reducing channel congestion. It achieves this by lessening the message generation rate per second, thus effectively handling dense traffic conditions. The details of this proactive Dual  $\alpha$  algorithm are presented in the following:

Algorithm1: Proactive Transmit Rate Control
Input: Predicted CBR value ( $CBR_p$ ) Output: $T_{off}$ (Time interval between two consecutive message generations)
<ol style="list-style-type: none"> <li>1. If (<math>t &gt; t_0</math>) <math>CBR_{sp}(t) \leq CBR_m(t)</math></li> <li>2. Else <math>CBR_{sp} = (1-\theta) \times CBR_{sp}(t-1) + \theta \times CBR_p(t)</math></li> <li>3. If (<math>CBR_{sp} &lt; 0.3</math>) then: <math>T_{off} \leq 100</math> // <math>T_{off}</math> in ms</li> <li>4. Elseif (<math>CBR_{sp} &gt; 0.3</math> and <math>CBR_{sp} &lt; 0.39</math>): <math>T_{off} \leq 200</math></li> <li>5. Elseif (<math>CBR_{sp} &gt; 0.4</math> and <math>CBR_{sp} &lt; 0.49</math>): <math>T_{off} \leq 400</math></li> <li>6. Elseif (<math>CBR_{sp} &gt; 0.5</math> and <math>CBR_{sp} &lt; 0.59</math>): <math>T_{off} \leq 500</math></li> <li>7. Else: <math>T_{off} \leq 1000</math>; // <math>CBR_{sp} &gt; 0.59</math></li> </ol>

Algorithm2: Proactive Dual $\alpha$ Algorithm
Input: Predicted CBR value ( $CBR_p$ ) Output: $t_{go}$ (Time of wait for the next transmission)
<ol style="list-style-type: none"> <li>1. If (<math>t &lt; t_0</math>) <math>CBR_{sp}(t) \leq CBR_m(t)</math></li> <li>2. Else <math>CBR_{sp}(t) = (1-\theta) \times CBR_{sp}(t-1) + \theta \times CBR_p(t)</math></li> <li>3. <math>CBR_s(n) = 0.5 \times CBR_s(n-1) + 0.5 \times (CBR_{sp}(t-1) + CBR_{sp}(t-1) / 2)</math></li> <li>4. If (<math>CBR_t &gt; CBR_s(n)</math>): <math>\delta_{offset} = \min(\beta \times (CBR_t - CBR_s(n)), G_{max}^-)</math></li> <li>5. Else <math>\delta_{offset} = \max(\beta \times (CBR_t - CBR_s(n)), G_{min}^+)</math></li> <li>6. If (<math>\delta(n-1) - \delta_{low}(n) &gt; th</math>): <math>\alpha(n) = \alpha_{High}</math></li> <li>7. Else <math>\alpha(n) = \alpha_{Low}</math></li> <li>8. <math>\delta(n) = (1-\alpha) * \delta(n-1) + \delta_{offset}</math></li> <li>9. If (<math>\delta(n) &gt; \delta_{max}</math>): <math>\delta(n) = \delta_{max}</math></li> <li>10. If (<math>\delta(n) &lt; \delta_{min}</math>): <math>\delta(n) = \delta_{min}</math></li> <li>11. <math>t_{go} = t_{pg} + \min(\max(t_{onpp} / \delta, 25ms), 1s)</math></li> </ol>

## 3.4 Simulations and Results

### 3.4.1 Simulation Scenario and Configurations

In this section, we conduct simulations to illustrate the effectiveness of our proposed solution for both the DCC reactive TRC and the Dual  $\alpha$  algorithms. The focal metrics of this study are the average CBR and the CBR convergence speed. A lower average CBR indicates a less saturated transmission channel, while the convergence speed reflects how quickly the channel stabilizes and the extent of local and global fairness achieved among all vehicles.

For the reactive DCC algorithm, our approach considers a sophisticated 5-active state-machine algorithm, which encompasses a total of 7 distinct states. This configuration is designed to provide granular control over the communication channel, adapting dynamically to varying network conditions. Detailed parameters for each state, along with their respective CBR threshold values, are given in Table 8.

*Table 8: Reactive TRC states and corresponding parameters.*

State	CBR interval	T <sub>off</sub> (ms)	Transmission rate (Hz)
Relaxed	CBR<0.19	60	16.7
Active1	0.19<CBR<0.27	100	10
Active 2	0.27<CBR<0.35	180	5.6
Active 3	0.35<CBR<0.43	260	3.8
Active 4	0.43<CBR<0.51	340	2.9
Active 5	0.51<CBR<0.59	420	2.4
Restrictive	0.59<CBR	460	2.2

In addressing the Dual  $\alpha$  algorithm, we have adopted the parameter set originally (detailed in [6]). This ensures consistency with established research and allows for a direct comparison of our enhancements. The specific values and configurations used in the Dual  $\alpha$  algorithm are outlined in Table 9.

*Table 9: Parameter value of the Dual  $\alpha$  algorithm*

Parameter	Value	Parameter	Value
$\alpha_{low}$	0.016	$\delta_{min}$	0.0006
$\alpha_{high}$	0.1	$\delta_{max}$	0.03
$\beta$	0.0012	$G^+_{max}$	0.0005
$th$	0.00001	$G^-_{min}$	-0.00025
$CBR_t$	0.68		

To comprehensively assess the performance of these algorithms under various conditions, we employed a range of traffic densities in our simulations. The chosen testing ground was the oval scenario, which comprises a 3.75-kilometer oval-shaped road with four lanes in each direction. This particular setup provides an ideal environment for examining the algorithms' behaviour in a controlled yet realistic setting. Our analysis focused on evaluating vehicle metrics in both transitory and steady states to gain insights into the algorithms' adaptability and performance. By monitoring these metrics before and after the environment reaches its maximum density, we can understand how effectively the algorithms respond to escalating network demands and maintain stability in high-density scenarios.

All simulations were conducted using the open-source V2X simulation framework, Artery [72]. This framework enables and facilitates the implementation of V2X services adhering to ETSI ITS-G5 protocols. Artery integrates with OMNet++ [73], a renowned discrete event simulation framework. OMNeT++ provides the capabilities to model and simulate complex network scenarios and interactions with high precision. Also, Artery encompasses additional frameworks, such as Vanetza [74], which implements the DCC mechanisms and algorithms. Within Vanetza, we implemented our proposed solution. Traffic generation and simulation are handled by SUMO (Simulation of Urban Mobility) [75]. Table 10 encapsulates the principal parameters of the simulation scenario.

*Table 10: Simulation parameters*

Vehicle flow parameters		
Density (vehicle/lane/Km)	Max Speed (m/s)	Simulation time (s)
50	13.2	350
Other simulation parameters		
Transmission Channel		CCH
ITS-G5 Service		CAM
Scenario		8-lane bidirectional oval road
Length		3750m

Our model was built using Python 3.9 and TensorFlow 2.5 [76]. The creation of the LSTM models, as well as running all simulations, were performed on an Ubuntu-based system with an Intel i7-9700 processor and 16 GB of memory.

Initial simulations employed the TRC reactive and Dual  $\alpha$  algorithms in their standard forms, with the subsequent collection of CBR values from each node in all the simulation environments, stored as dataset files.

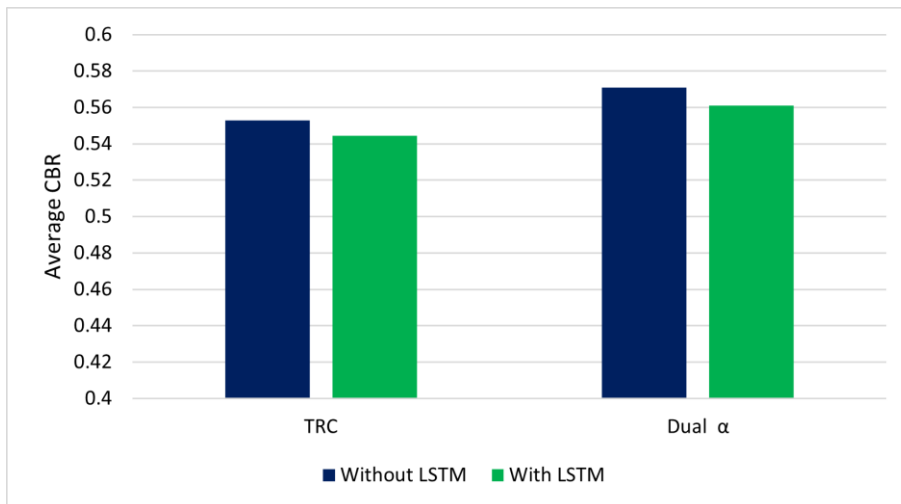
The next step is to initiate the offline phase by training and creating our LSTM model (a specific LSTM model dedicated to each algorithm). We explored a range of LSTM

configurations to determine the most effective ones for our simulation environment. Finally, to test these models in a dynamic setting, we equipped each vehicle in our simulation with one of these LSTM models. We conducted a series of simulations, altering both the LSTM configurations and the vehicle densities, to rigorously test the models under varied conditions. The goal was to identify the most efficient LSTM configuration for real-time applications in diverse traffic scenarios. We define the number of input and output steps employed by our LSTM agent using the `n_steps_in` and `n_steps_out` variables, representing our multi-step time-series forecasting model. The chosen configurations are detailed in Table 11.

*Table 11: LSTM parameter settings*

Parameter	Parameter Value
<code>n_steps_in</code>	20 (N=2)
<code>n_steps_out</code>	40 (M=4)
LSTM layers, number of cells	4,[50,50,50,50]
Activation Function	ReLU
Batch size	16
Optimizer	AdamOptimizer
Discount factor	0.9
Hidden layer 64 neurons	Hidden layer 64 neurons
Replay memory D	500

### 3.4.2 Simulation Results



*Figure 17. Average CBR for both DCC algorithms*

Figure 17 illustrates the average CBR values using both traditional and LSTM-based proactive approaches. The TRC reactive DCC algorithm, when enhanced with LSTM predictions, shows a reduction in average CBR from over 0.55 to approximately 0.54. Similarly, in the Dual  $\alpha$  algorithm, the average CBR decreased from 0.57 to 0.56. These reductions signify fewer channel congestion peaks, attributable to the proactive adjustment of CAM generation



rates based on predicted rather than current CBR values. Consequently, this integration of smoothed CBR predictions into both algorithms leads to more efficient resource utilization and reduced channel congestion

Figure 18 and Figure 19 depict the CBR variations for the TRC and Dual  $\alpha$  algorithms (both legacy and proactive versions). Figure 18 shows that, for the Dual  $\alpha$  algorithm, the use of LSTM predictions does not significantly alter average CBR values during transitory and steady states. However, the proactive Dual  $\alpha$  demonstrates better and faster convergence in the steady state compared to its adaptive counterpart.

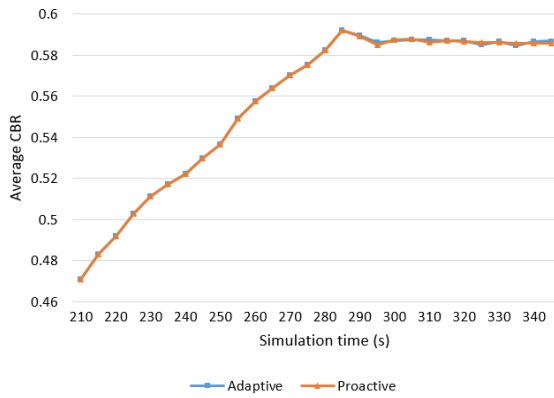


Figure 18. Average CBR variation for DCC Dual  $\alpha$

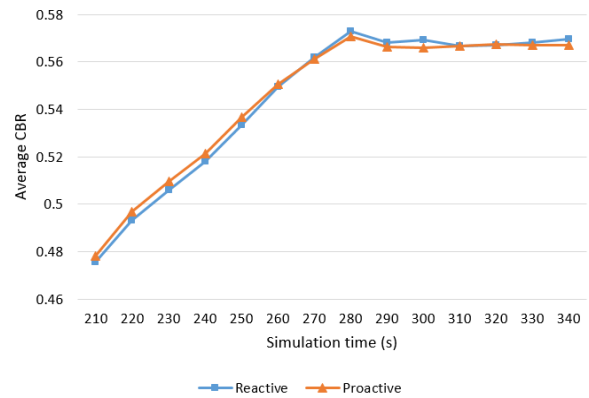


Figure 19. Average CBR variation for DCC TRC

In the case of the DCC TRC algorithm, the difference becomes clearer, as illustrated in Figure 19. In the transitional phase, the proactive algorithm demonstrates a higher average Channel Busy Ratio (CBR) compared to its traditional counterpart. This indicates that vehicles utilizing LSTM escalate their data transmission rates more rapidly and transmit a greater volume of data during periods of low channel congestion. This behaviour is deemed optimal, as it suggests that vehicles efficiently maximize network resource utilization without causing network overload. This efficiency becomes even more evident during the steady state phase. Initially, the proactive approach effectively mitigates the initial surge in average CBR observed with traditional methods. Furthermore, vehicles employing LSTM technology achieve a rapid convergence to a stable average CBR, which subsequently exhibits minimal fluctuations. These observations underscore the superiority of integrating LSTM predictions into both methodologies, leading to enhanced network resource management, alongside quicker and more reliable network stability. Consequently, our proactive approaches enable faster achievement of local fairness among nearby neighbouring vehicles, as well as faster achievement of global fairness among all vehicles contributing to congestion.

Despite the seemingly modest reductions in average CBR values, these improvements carry substantial implications for network efficiency and reliability. Specifically, the lower frequency

of channel congestion peaks, as indicated by the reduced average CBR, leads to less strain on the network. This, in turn, enhances the network's reliability and fosters a more equitable distribution of resources among vehicles. By ensuring a smoother flow of communication and reducing the likelihood of network saturation, the integration of LSTM-based predictions into traffic management algorithms contributes to a more resilient and balanced vehicular network environment.

It is important to emphasize that our approach has no negative impact on the reliability or transmission quality of the channel. This is demonstrated by the average Packet Delivery Ratio (PDR) values. Indeed, when using proactive DCC algorithms, the average PDR in all simulation environments is slightly increased, as seen in the results summarized in Table 12.

*Table 12: Average PDR*

TRC	TRC + LSTM	Dual $\alpha$	Dual $\alpha$ + LSTM
80.19635%	80.34709%	77.65844%	77.66450%

### 3.5 Conclusion

In this chapter, we explored the TRC reactive and Dual adaptive algorithms and introduced the proactive Decentralized Congestion Control (DCC) algorithms empowered by LSTM artificial intelligence. These innovations effectively help to mitigate channel exhaustion, smooth congestion peaks, and accelerate channel stability, leading to faster convergence, resource optimization, and enhanced fairness among vehicles. Additionally, a slight increase in the PDR attests to the system's reliability. The application of LSTM-based predictions represents an initial foray into leveraging AI for traffic management. Future research may extend to alternative AI algorithms that could further refine these results, promising even greater efficiency and reliability in vehicular networks. Also, potentially revolutionizing DCC mechanisms in vehicular networks. Our work marks a significant advancement toward efficient and equitable vehicular communication. The Proactive DCC algorithms, with LSTM predictions, demonstrate the transformative potential of AI-driven solutions in modern transportation systems, paving the way for further innovation.

## 4. ETSI ITS-G5 Network Slicing

### 4.1 Introduction and Problem Statement

The integration of Network Slicing within ITS-G5 for Vehicular Ad hoc Networks (VANETs) marks a significant advancement in the field of C-ITS, particularly in the context of

the emerging 5G network architecture. This development is increasingly critical as the digital technology and transportation sectors continue to evolve rapidly. Network slicing offers a transformative solution to address the complex challenges encountered in C-ITS, especially in areas with high-density traffic. This approach involves creating multiple, independent logical networks, or "slices", on a single shared physical network infrastructure. It is crucial for meeting the diverse and stringent Quality of Service (QoS) requirements that are essential across various Vehicle to Everything (V2X) applications and use cases, where latency and reliability are of utmost importance.

One of the central challenges in this domain is the effective management of data communication within ITS-G5 VANETs. This includes the handling of CAMs and DENMs, among other types of messages, which are vital for real-time information sharing among ITS Stations (ITS-S). These messages cover crucial data such as vehicle presence, position, speed, and other significant status indicators, along with alerts regarding events like accidents or road hazards. In high-density traffic scenarios, the excessive volume of data exchange can lead to significant issues with bandwidth and QoS, presenting a major challenge in ensuring the timely and efficient delivery of these critical communications.

The concept of network slicing in ITS-G5 VANETs represents a paradigm shift, introducing customizable, scalable, and isolated network environments tailored to specific communication needs. This advancement promises to enhance performance and security within each network slice. However, it also brings forth the challenge of ensuring that high-priority traffic, such as emergency vehicles, is adequately serviced with the necessary urgency and reliability.

Moreover, as the automotive market undergoes a technological transformation towards more advanced C-ITS, there is an increasing demand for ultra-low latency and ultra-high reliability under high mobility and density conditions. Achieving these standards in a dynamic and constantly evolving traffic environment poses significant challenges. The need for a robust and flexible network architecture capable of adapting to varying traffic conditions and priorities is paramount. Addressing these challenges is essential for the advancement of C-ITS, contributing to the development of transportation systems that are not only smarter and safer but also more efficient and responsive to the needs of a modern, connected society.

## **4.2 Literature Overview**

The majority of V2X network slicing literature such as [77][78][79], among others, primarily focuses on the aspect of resource allocation across the Radio Access Network (RAN), with less emphasis on the Core Network (CN) and computation resource allocation, primarily within LTE-V2X and 5G (NR-V2X) infrastructures. This field of study encompasses a range of approaches, from analytical and simulation-based methods to practical experimental setups.

Many of these works are summarized in [37]. [80] introduces a reference model (Figure 20) built upon a three-layered structure similar to that in [81]. This model comprises infrastructure, service, business, and management and orchestration layers. The architecture highlights the instantiation of a slice dedicated to autonomous driving, with an emphasis on using Vehicle-to-Vehicle (V2V) as the primary Radio Access Technology (RAT) connection mode. This slice requires ultra-low latency and highly reliable Vehicle-to-Network (V2N) connectivity for effective data exchange. The work in [82] integrates ITS-G5/C-V2X, MEC, 5G, and network slicing in a comprehensive reference architecture.

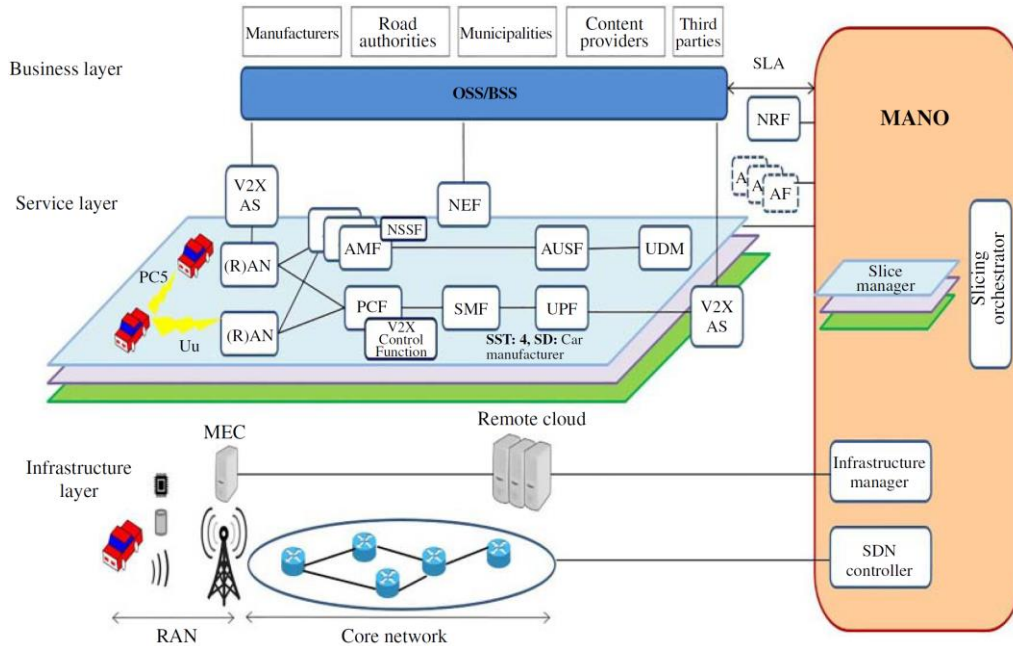


Figure 20. Network slicing architecture and slice instantiation for autonomous driving

Expanding on the domain of V2X network slicing, a notable contribution is from [43], where the authors provide a comprehensive framework for a V2X network slicing model. This model aligns with the 3GPP specifications, advocating for a dedicated slice specifically for V2X services. The proposed model includes four distinct slices designed to cater to various V2X services such as autonomous driving, tele-operated driving, vehicular infotainment, and vehicle remote diagnostics and management.

Further studies like [83] and [84] delve into specific aspects of vehicular networks. [83] focuses on cooperative driving under roaming conditions, while [84] develops an architecture for traffic differentiation and flow isolation in network slices. This architecture is validated in realistic scenarios considering various traffic types and RATs. [85] introduces an end-to-end network slicing framework that leverages Software-Defined Networking (SDN), fog, edge, and cloud computing technologies, specifically for autonomous driving services.

Resource allocation within the RAN segment for enhanced Mobile BroadBand (eMBB) and V2X slices is the focus of [86]. The study employs a strategy based on offline Q-learning to optimize resource distribution. In a similar vein, [87] tackles the issue of radio resource allocation for infotainment and autonomous driving slices. These slices are supported through Vehicle-to-Infrastructure (V2I) and V2V links, respectively. Regarding RAN slicing for vehicular networks, [42] introduces an architectural framework to support V2X network slices using network softwarization technologies like SDN and NFV, and [88] discusses the main enablers and challenges of RAN slicing.

As for Wi-Fi-based slicing research, the work in [89] presents a novel approach to infrastructure sharing and slicing in Wi-Fi Access Points through airtime resource allocation. Key to this architecture is a classifier module that discerns various traffic flows, assigning each flow to a specific queue. Additionally, the architecture utilizes multiple queues for each user, corresponding to the number of slices their traffic is associated with. This setup ensures that a Wi-Fi station has individual queues for each slice it services. These queues are managed by a Proportional Time Deficit Round Robin mechanism, initially introduced in [90], which effectively schedules services across queues to fulfil the QoS requirements of each slice. Finally, The work presented in [91] proposed a Virtual Network Slicing Broker (VNSB) that creates logical slices using NFV and SDN and Wireless Network Virtualization (WNV) technologies to create and maintain slices in the IEEE 802.11ah network.

However, specific guidelines for ITS-G5 RAN slicing remain underexplored. Also, there is a notable gap in the literature regarding end-to-end slicing of the ITS-G5 network. To date, to the best of our knowledge, no work has specifically addressed an end-to-end network slicing architecture for ITS-G5 that encompasses slicing across ITS-Stations, access networks, core networks, and ITS data centres.

## **4.3 A RAN Slicing Architecture for ITS-G5 C-ITS**

### **4.3.1 Research Objectives**

In this research, we aim to develop and detail a novel ITS-G5 RAN slicing architecture within the scope of C-ITS as standardized by ETSI. This architecture is designed to integrate new modules through the ITS protocol stack, enabling the creation of multiple RAN slices. Each slice is characterized by distinct priorities and performance parameters, ensuring optimal QoS, efficient traffic prioritization, shaping, and policing.

A key focus of this work is on the aspect of security, particularly in terms of securing vehicle authentication to the respective network slices. This approach is crucial in maintaining the integrity and trustworthiness of communication within the C-ITS environment. Additionally,

our architecture emphasizes the importance of isolation between network slices. This isolation is critical to guarantee both the performance and security of each slice, ensuring that the operational dynamics of one slice do not adversely affect the others.

### **4.3.2 RAN Slicing Advantages**

RAN slicing plays a significant role in fulfilling the network slicing motivation for user QoS and service customization, by providing efficient use of limited frequency spectrum resources. RAN slicing aims to contribute to the following [40][41]:

- **Maximized RAN resource utilization:** dynamically sharing RAN resources between services using schedulers with different key performance indicators (KPIs) and other mechanisms to allow optimized bandwidth use in a flexible and programmable manner. We extend C-ITS queuing with a Classifier and a Mapper to share RAN resources dynamically.
- **RAN slice-awareness:** QoS mechanisms allocate network resources to differentiate network traffic with different priorities, but these mechanisms are insufficient to tailor RAN resource attribution to slices. Thus, new specific RAN policies should be applied. We extend C-ITS with a Meter / Shaper that decides in-profile and out-of-profile traffic.
- **Traffic differentiation mechanisms:** calls for deploying traffic prioritization mechanisms over the RAN, such as radio schedulers and QoS, etc. This is fully taken into consideration in our architecture from the opportunities provided by ITS-G5 DCC and transmission queues.
- **Isolation and protection mechanisms:** when self-contained slices share the RAN resources, it is essential to use mechanisms that ensure proper isolation where congestion and security failure in one slice do not affect the others. Thanks to the introduced mechanisms, we provide secure access to a given slice while performing traffic shaping and policing alongside traffic prioritization and isolation.
- **Infrastructure management:** allowing the efficient creation and modification of applications, services, and software resources considering the dictated performance and business requirements. We enhance slice management and orchestration (MANO) with coordinated network resources and lifecycle management of network slicing including secure access to the slice.

### **4.3.3 Security and Privacy in C-ITS**

To enhance security and privacy in C-ITS, significant efforts have been made by IEEE and ETSI to establish a robust Public Key Infrastructure (PKI) architecture. PKI is an intricate system of authorities, policies, and procedures dedicated to the management, creation, distribution, utilization, storage, and revocation of digital certificates. It plays a pivotal role in

associating public keys with the respective identities of entities, achieved through a meticulous process of registration and issuance of certificates, as outlined in [92][93]. The architecture of PKI is hierarchical, resembling a tree structure of intercommunicating authorities. In the context of this study, our focus is primarily on the lower segment of this hierarchy, as depicted in Figure 21.

The Root Certificate Authority (RCA) serves as the uppermost authority in this hierarchy, responsible for certifying the identities of entities that issue digital certificates, namely the Enrolment Authority (EA) and the Authorization Authority (AA). The distinct roles of these two entities are critical in maintaining the vehicle's privacy and anonymity within the C-ITS framework. The EA is in charge of managing the vehicle's Enrolment Certificate (EC), which is a life-long certificate embedded in the vehicle. In contrast, the AA is tasked with providing Pseudonym Certificates (PCs) to the vehicle. These PCs are essentially alternate identities used by the vehicle to sign its messages, thereby significantly hindering the ability to track the vehicle. This strategic division of responsibilities between the EA and AA is essential for ensuring the privacy of vehicles, as the EA is only aware of the vehicle's EC, while the AA handles only its PCs [93]. This architecture effectively balances the need for security with the imperative of preserving vehicle anonymity and privacy in the C-ITS environment.

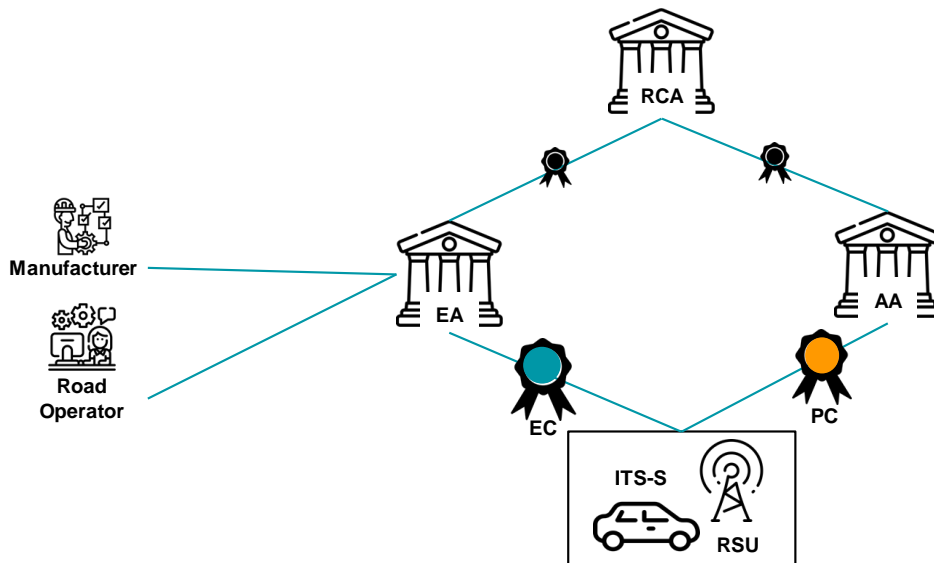


Figure 21. Architecture for PKI infrastructure.

#### 4.3.4 RAN Slicing Architecture

The proposed solution introduces a suite of modules designed to enhance the ITS station protocol stack, tailored to meet the QoS requirements of different network slices. These modules are specifically engineered to classify, mark, shape, and police network traffic, facilitating the differentiation and prioritization of slices based on their respective priorities. The key components of this solution are:

- **Waiting queues:** Aligned with the access layer's access categories (VO, VI, BE, BK), these queues are instrumental in ensuring QoS through traffic prioritization. Traffic in higher priority queues is given preferential channel access over lower priority queues. Each queue is equipped with its own Meter / Shaper mechanism.
- **Classifier:** Stamps the traffic with a DCC Profile value (DP), based on the slice it belongs to.
- **Mapper:** Effectively aligns stamped packets with their respective traffic class (TC), based on the DCC Profile (DP) values assigned to each packet.
- **Meter / Shaper:** It is central to the architecture, executing traffic shaping and policing. It regulates traffic burstiness by setting limits on the average rate and maximum burst size, as detailed in [94]. The Meter / Shaper operates with two parameters: Information Rate (IR) and Burst Size (BS). The IR specifies the traffic volume added to the Meter / Shaper per second, essentially representing the average message rate. The BS specifies the capacity of the Meter / Shaper, i.e., the maximum amount of traffic that the Meter / Shaper can hold, and thus the maximum number of packets that can be sent through the network in a very short time interval (burst). This module works as a counter incremented with points at a rate (IR) proportional to the desired mean message rate. However, the counter value can never exceed more than its maximum capacity (BS). When the counter reaches a value equal to or greater than one point, be a value “*n*”, the “*n*” following packets are labelled as "in-profile" packets, and the counter is decremented by “*n*” points [21]. Packets arriving when the counter has a value of zero, are labelled as "out-of-profile" packets. This mechanism aims to limit the packet transmission rate of the ITS stations based on the parameters that define their slices' QoS.
- **Enqueuing Element:** This component queues the in-profile packets and directs the out-of-profile packets either to the dropper or the marker, based on a Meter / Shaper that the slice manager and orchestrator (MANO) sets.
- **Marker:** When a predetermined number *X* of packets are classified as out-of-profile by the Meter / Shaper, this module redirects the *X* oldest packets at the queue's front to the Dropper. It then enqueues the *X* newest out-of-profile packets, containing newer information, at the tail of the queue for priority processing.
- **Dropper:** Responsible for discarding out-of-profile packets.
- **Monitor:** Records traffic statistics and monitors the state of the channel.

The RAN slice in our architecture is distinctively defined by its priority, Information Rate (IR), Burst Size (BS), and the specific services it supports. The slice manager and orchestrator (MANO) dynamically manages these slices, overseeing their lifecycle, resource orchestration,



and scaling. MANO's role includes creating RAN slices and configuring their QoS parameters, such as slice priority, IR and BS of the Meter / Shaper, and designated service types. MANO also sets the criteria for selecting a RAN slice, which can include factors like geographical location, mission criticality, authority stations, traffic flow, and station manufacturers.

Figure 22 illustrates the integration of proposed modules within the ITS-G5 protocol stack. The application layer produces various packet types (1), each potentially belonging to different RAN slices. The selection of a RAN slice is based on the conditions set by MANO. In the facilities layer, packets are assigned a DCC Profile (DP) by the Classifier (2), corresponding to their slice. These packets then move through the networking & transport layer to the access layer, where the Mapper (3) aligns each packet to a traffic class (TC) based on its DP. This mapping directs packets to the appropriate Meter / Shaper of their RAN slice.

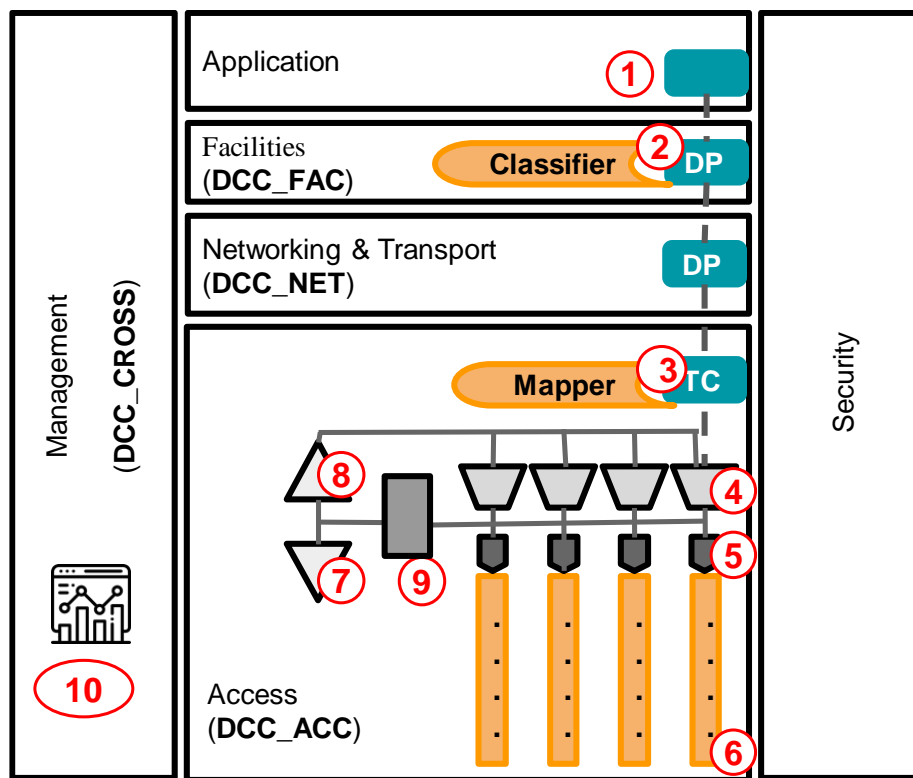


Figure 22. Integration of the solution modules through the ITS station protocol stack

The Meter / Shaper (4) categorizes traffic into in-profile or out-of-profile groups. In-profile traffic is queued by the Enqueuing element (5) in waiting queues (6), while out-of-profile traffic is directed to the dropper (7) or the marker (8), as per configurations (9) from MANO. The monitor (10) records all process statistics, including packet transmission rates, latency, and the number of dropped packets, aiding in decision-making. Higher priority slices typically offer better transmission rates, lower latency, and fewer packet drops compared to lower priority slices.

### 4.3.5 Slice Authentication

The proposed architecture enhances secure access to specific network slices, employing digital signatures for vehicle authentication within their respective slices. A pivotal addition to this architecture is the "slice access database" (SA-DB), managed by the MANO. SA-DB maintains a record of each vehicle's Enrolment Certificate (EC) and a dynamic list of slice identifiers that a station is permitted to access, with updates based on MANO's settings, network performance, and other policies. This integrated approach within the new PKI architecture is detailed in Figure 23.

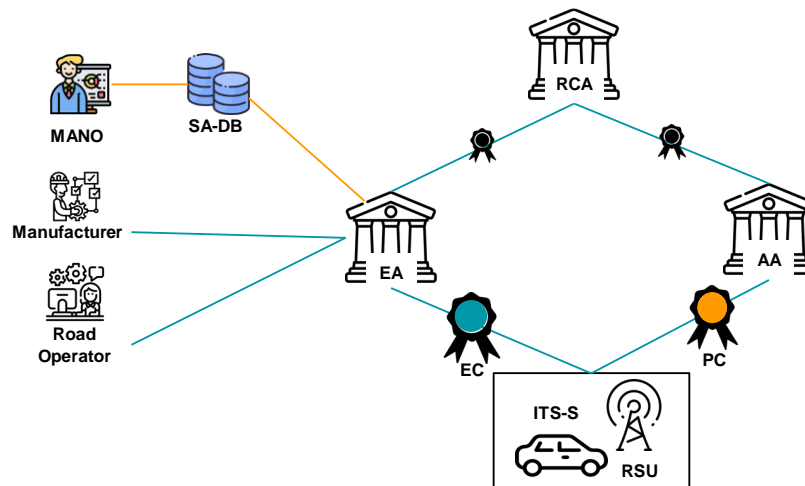


Figure 23. New PKI architecture

The process begins when an ITS station seeks access to a slice, initiating a request accompanied by its EC and digital signature to the Authorization Authority (AA) for a pool of Pseudonym Certificates (PC). Upon receiving the request, the AA first verifies the EC's validity. If valid, the request is passed to the Enrolment Authority (EA); if not, it is disregarded. The EA then authenticates the station's identity by checking its EC and digital signature. Following successful verification, the EA consults the SA-DB to confirm whether the requested slice is within the station's authorized slices. If the station is authorized to the slice, the EA notifies the AA, which then generates and dispatches a pool of PCs to the station. In the event of EC invalidation or if the station lacks authorization for the slice, the request is ignored. Figure 24 sums up this whole process.

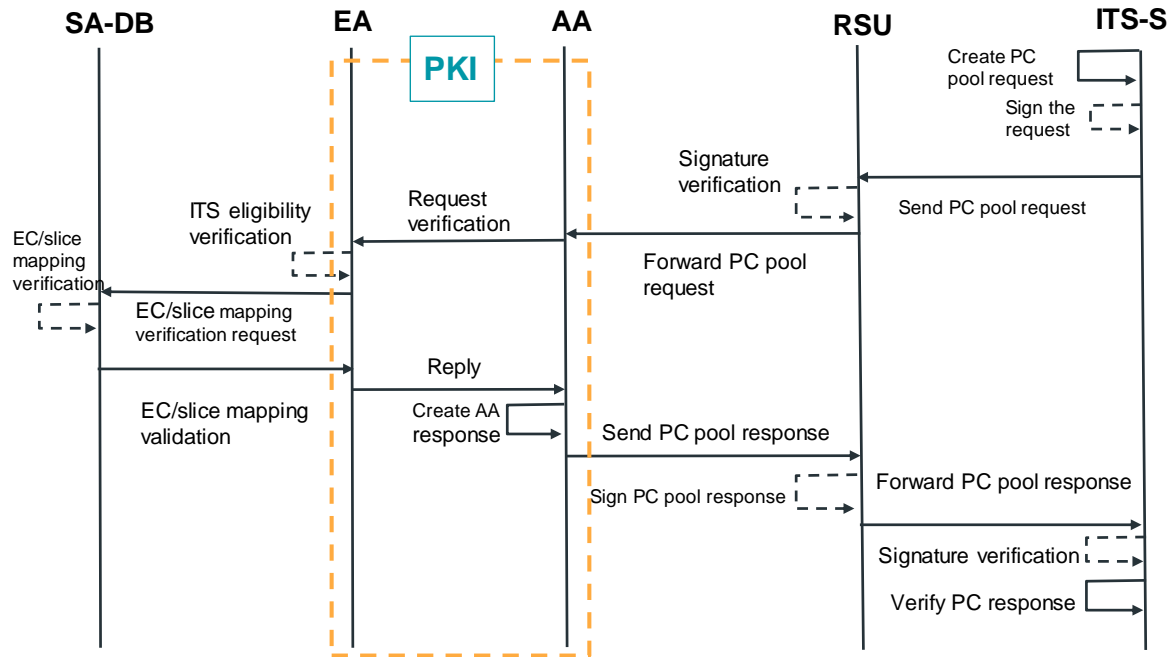


Figure 24. The process of PC pool attribution to an ITS-S

The station uses the PCs from the pool to sign its messages within the demanded slice. When the PC pool depletes or reaches a set threshold, the station requests a new pool from the AA, repeating the aforementioned process. Each PC, serving as a temporary and unique identity, is used only once to enhance the station's privacy and hinder tracking. The pool size and threshold are configured through MANO. Furthermore, Roadside Units (RSUs) play a crucial role in this framework, acting as intermediaries between the ITS-G5 network and the PKI network. They facilitate the transmission of station requests and AA responses, thereby streamlining the communication and authorization processes within the network.

### 4.3.6 Implementation and Simulation

In this subsection, we detail the implementation and simulations conducted to validate our ITS-G5 RAN slicing architecture. We deployed diverse scenarios, each aimed at demonstrating the distinct functionalities of our architecture. The results from these simulations are critically analysed to evaluate the performance of our contribution. It's also important to note that all DCC algorithms were deactivated in these simulations to ensure a clear assessment of the architecture's capabilities.

#### 4.3.6.1 Implementation of ITS-G5 RAN Slicing

Our proposed ITS-G5 RAN slicing architecture, and all simulations are implemented and run on the Artery simulator. The service used in this simulation is the CAM service for both vehicles and RSUs, and the priority of the CAM messages varies according to the slice to which the vehicle or the RSU belongs to. The different architectural elements are coded in C++ and then added as classes, methods, or segments of codes to the different Artery modules.

Regarding slice authentication, this component is implemented in a simplified format for the simulations. We assume that vehicles are pre-equipped with a pool of pseudonym certificates (PC). RSUs receive and authenticate messages from vehicles requesting access to particular slices. The detailed process of how vehicles acquire their PCs, which involves interactions with the core network and various authentication servers and databases, is out of the scope of this paper.

#### 4.3.6.2 Data Rate Results

In this scenario, we demonstrate the traffic shaping and policing capabilities of our architecture using a simulated 2 km bidirectional highway with five lanes per direction. Two traffic slices with distinct priorities were established: a high-priority (HP) slice for police vehicles, permitting up to 5 messages per second, and a lower-priority (LP) slice for cars, allowing up to 3 messages per second. The simulation settings are detailed in Table 13.

*Table 13: Data rate simulation parameters.*

Transmission channel	CCH
ITS-G5 service	CAM
Message generation rate	10 Hz
Simulation time	300 s
Bidirectional highway length	2 Km
Slices and traffic proportions	Police (HP): 20% Cars (LP): 80%
Slices' parameters	HP: IR = 5; BS = 10 LP: IR = 3; BS = 10

We analysed a subset of stations from both traffic flows, recording the message transmission rate. Figure 25 presents the traffic shaping effectiveness, illustrating how the Meter / Shaper controls the message rate. An initial burst of 10 messages (reflecting the burst size, BS = 10, for both slices) was observed, followed by regulation to the predefined rates: 5 messages per second for the HP slice (IR = 5) and 3 messages per second for the LP slice (IR = 3). These findings confirm the efficacy of our traffic shaping and policing strategy.

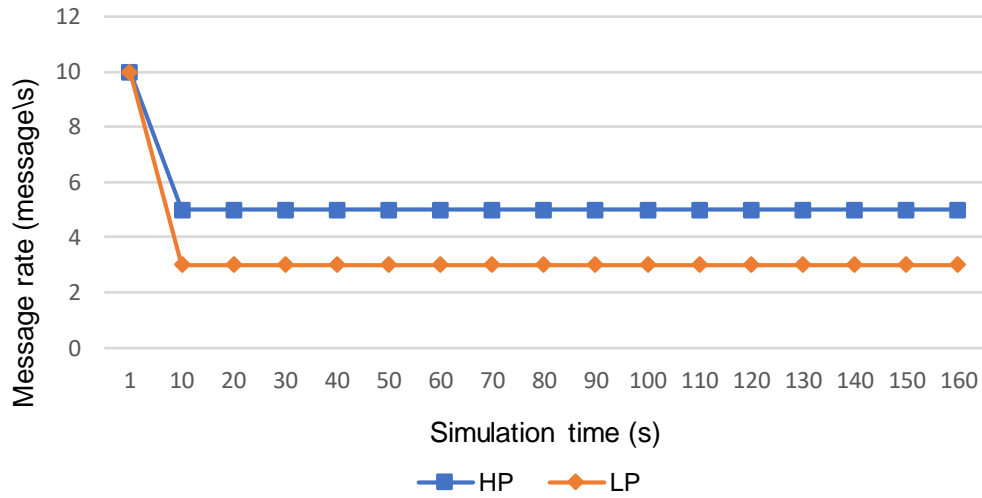


Figure 25. Message rate shaping of ITS-Ss belonging to HP and LP slices respectively

#### 4.3.6.3 Slice Isolation and Traffic Prioritization Results

In this section, we demonstrate the efficacy of our solution in managing traffic prioritization and maintaining isolation performance under varying traffic densities. This involves analysing the system's response to different ratios of high-priority (HP) and low-priority (LP) traffic. Furthermore, we highlight the latency enhancement achieved for the HP traffic slice compared to the LP slice. Table 14 provides a comprehensive summary of the simulation parameters utilized in our study.

The following charts (Figure 26 - Figure 29) show the mean latencies of the HP and LP slices, for each density value, while varying the HP traffic proportions (we present charts for densities of 20 and 50 vehicles/km/lane, as the results for other densities are remarkably similar and do not add significant additional insights).

Table 14: Traffic prioritization simulation parameters.

Transmission channel	CCH
ITS-G5 service	CAM
Message generation rate	10 Hz
Simulation time	300 s
Bidirectional highway length	2 Km
Number of lanes/direction	5 lanes/direction
Slices and traffic proportions	Police (HP) – {5%, 10%, 15%, 20%} Cars (LP) – {95%, 90%, 85%, 80%}
Densities (vehicle/km/lane)	{20, 30, 40, 50}

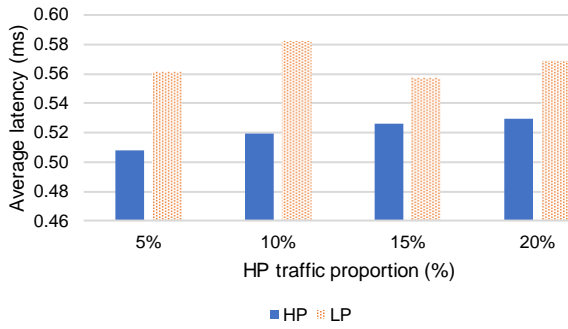


Figure 26. Average latencies for a density of 20 v/km/lane

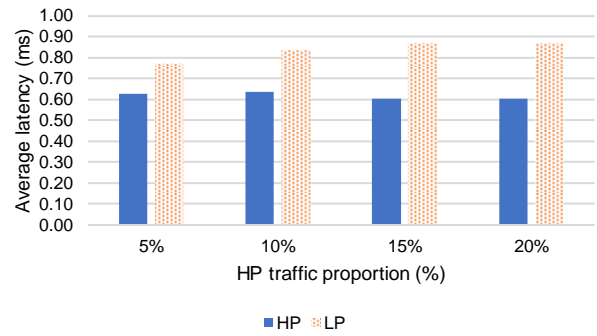


Figure 27. Average latencies for a density of 50 v/km/lane

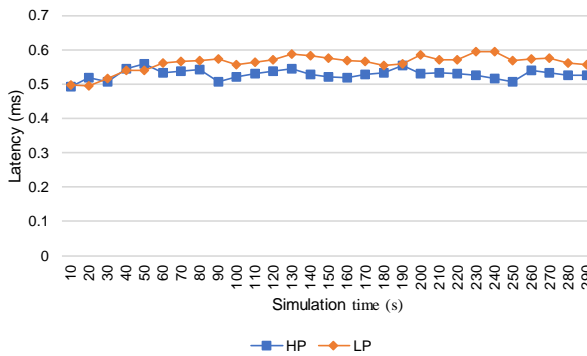


Figure 28. Latency variations for a density of 20 v/km/lane

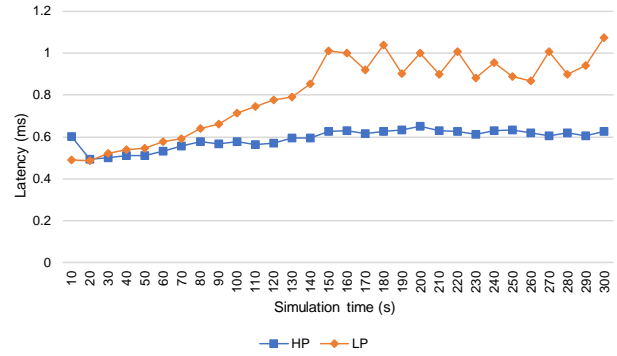


Figure 29. Latency variations for a density of 50 v/km/lane

The above bar charts provide a clear visualization of the isolation effectiveness between the high-priority (HP) and low-priority (LP) slices in varying traffic density scenarios. These figures consistently show that the average latency for the HP slice is lower than that for the LP slice. This pattern holds across different HP traffic proportions and for a range of densities, from less dense scenarios (20 vehicles/km/lane) to highly congested, ultra-dense scenarios (50 vehicles/km/lane).

This observed performance can be explained by the specific mapping of the HP slice's traffic to the highest priority DCC profile, DP0, and its transmission through the Voice (VO) access category. The VO access category is favoured with shorter waiting times for channel access. In contrast, the LP slice's traffic is mapped to the lowest priority DCC profile, DP3, and is transmitted through the Background (BK) access category. Consequently, the LP traffic incurs longer waiting periods in the queue to access the channel. The determinants of these waiting times are the contention window (CW) interval and the Arbitration Inter-Frame Space (AIFS) values, as detailed in Table 4.

The following Table 15 sums up the median and 95<sup>th</sup> percentile values for the different latencies with a percentage of HP traffic equal to 20%, and densities going from 20 v/km/lane to 50 v/km/lane.

Table 15: Median and 95th percentile values

Density	Slice	Median (ms)	95 <sup>th</sup> percentile (ms)
20 v/Km/lane	HP	0.4989	0.5496
	LP	0.5226	0.5878
30 v/Km/lane	HP	0.5254	0.5975
	LP	0.5705	0.6660
40 v/Km/lane	HP	0.5316	0.6102
	LP	0.6098	0.7340
50 v/Km/lane	HP	0.5536	0.4800
	LP	0.7767	1.0402

These results show that traffic of the HP slice has a lower median latency and lower 95th percentile latency compared to the LP slice. This proves again the performance of our architecture in terms of traffic prioritization and QoS differentiation.

#### 4.3.7 Conclusion

This study presents a new ITS-G5 RAN slicing architecture designed for V2X applications, with a major contribution being the integration of advanced traffic management strategies. These include traffic shaping and policing, as well as prioritization and differentiation, implemented through distinct priority slices with secure authentication for network resource access. The architecture's effectiveness was validated through simulations demonstrating efficient message rate regulation and traffic prioritization by creating slices of different priority levels.

Performance evaluations under various vehicle densities and traffic ratios showed that the high-priority (HP) traffic slice consistently achieved lower latency than the low-priority (LP) slice, even in congested conditions. These results emphasize the architecture's capability in effectively managing and prioritizing V2X communications, indicating its potential suitability for dynamic vehicular environments.

## **4.4 End-to-End Network Slicing for ITS-G5 Vehicular Communications**

### **4.4.1 Research Objectives**

The primary objective of this research is to propose and implement a novel end-to-end network slicing architecture tailored for ITS-G5 vehicular communications. This architecture is designed to orchestrate resources across various segments of ITS infrastructure, including radio access, aggregation, core networks, and the ITS data centre. The central focus is on enhancing the overall efficiency and QoS within vehicular communication systems by optimizing resource allocation.

A key goal of this study is to develop and validate a slicing approach based on traffic discrimination for ITS-G5. The proposed slicing mechanism specifically targets improved end-to-end (E2E) latency for high-priority traffic and user groups. By doing so, this contribution aims to showcase tangible improvements in E2E latency, particularly for the higher-priority slices. This aspect is crucial for advanced Vehicle-to-Everything (V2X) applications, such as autonomous driving and platooning scenarios, where minimizing latency is imperative for safety and efficiency.

### **4.4.2 Proposed Architecture Overview**

This section outlines our comprehensive end-to-end slicing architecture for the ITS-G5 network, detailing both the ITS-G5 slicing system architecture and its management. Building upon the network architecture discussed in Section 2.4.1, our design incorporates an aggregation network situated between the access and core networks. This network functions as a pivotal juncture, collecting traffic from various road infrastructures. Depending on the specific requirements of the network, multiple aggregation levels may be integrated, enhancing data consolidation from the access networks before transmission to the ITS-G5 core network.

The ITS data centre is equipped with substantial computing and storage capabilities. It serves as the hub for hosting diverse V2X application servers, Public Key Infrastructure (PKI), Authentication, Authorization, and Accounting (AAA) servers, among others. This architecture ensures that messages transmitted by an ITS station are efficiently routed to and from the remote V2X application servers. These servers process the data, playing a crucial role in delivering advanced ITS services. To optimize performance, certain service processing tasks are strategically executed at specific edge nodes, closer to the data sources.

Our slicing approach is comprehensive, spanning across the ITS ad hoc networks, access and aggregation networks, and the core network, extending all the way to the remote ITS data centre, as depicted in Figure 30. This mechanism is highly flexible, allowing the creation of



multiple slices to accommodate a range of scenarios and use cases, each with its specific priority settings. For example, a slice with higher priority could be configured using ITS-G5 service parameters (such as a high-priority DCC profile, optimal generation rate, etc.), coupled with dedicated resources (including bandwidth, CPU, storage, and memory). This configuration ensures adherence to stringent QoS requirements like latency, throughput, and reliability, while also maintaining robust security standards.

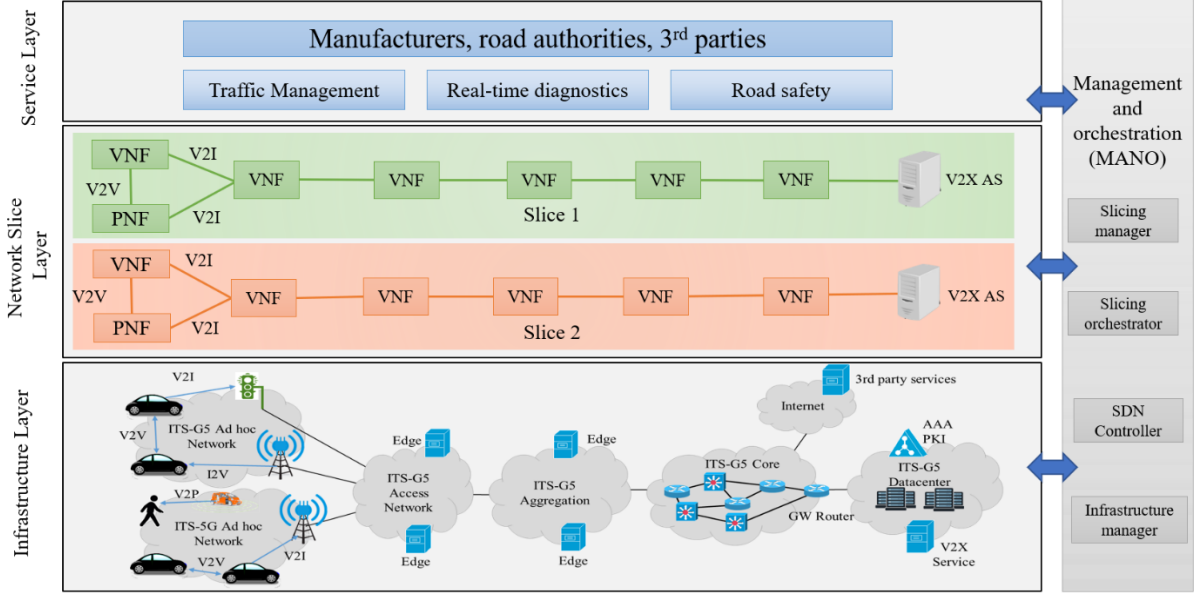


Figure 30. ITS-G5 global system architecture overview

The architecture of the end-to-end slice comprises two main components: (1) radio resources, represented as ITS-G5 Physical Network Functions (PNFs) for RAN slicing, and (2) resources for access/aggregation and core networks, manifested as end-to-end bandwidth and computational power allocations for links connecting roadside ITS-Ss to V2X application servers. These links traverse various nodes, such as edge nodes, routers and switches, and data centres, where computational resources are allocated for instantiating Virtual Network Functions (VNFs).

Formally, we describe a slice  $S$  as a graph  $\mathbb{G}^S(\mathbb{N}^S, \mathbb{E}^S)$  where  $\mathbb{N}^S$  is a set of VNFs and PNFs composing the slice and  $\mathbb{E}^S$  is a set of virtual links connecting these VNFs/PNFs. Besides, each VNF instance  $v \in \mathbb{N}^S$  of slice  $S$ , requires a predefined amount of resources in terms of computing, memory, and storage and is denoted by  $\Psi^v$ . Similarly, each virtual link  $(k, l) \in \mathbb{E}^S$  connecting two VNFs  $k, l \in \mathbb{N}^S$  has a bandwidth requirement  $\Omega^{(k,l)}$ . The placement and chaining of VNF on the underlying network (substrate network) are out of this paper's scope. A survey on this topic can be found in [95].

To enhance slice management within the ITS-G5 network, we include in our architecture a dedicated slice management and orchestration (MANO) component. This MANO

is designed to efficiently handle various critical functions including configuration management, optimization of physical resources, and managing the lifecycle of network slices. These capabilities are integral to ensuring that each slice or service consistently meets its QoS requirements. It includes [42]:

- **Slice Manager:** Responsible for slice description through well-defined templates, slice instantiation, and lifecycle management.
- **Slicing Orchestrator:** Manages the distribution of physical infrastructure resources among slices from the same operator and facilitates information exchange with other road operators. This is vital for enabling platform sharing on a global scale, which is particularly beneficial for V2X services as vehicles navigate through different operators' domains (Roaming).
- **SDN Controller:** This controller offers a centralized approach to managing network resources. Its functions include traffic engineering, network statistics monitoring, and controlling nodes via southbound interfaces like OpenFlow. It also steers traffic toward appropriate edge nodes and manages functions related to User and Control Plans, including resource reconfiguration and handover processes between edge nodes.
- **Infrastructure Manager:** Takes charge of the infrastructure orchestration by monitoring the storage and processing status of points of presence nodes such as edge nodes and data centres.

A primary goal of the MANO is to also set the optimal resource allocation parameters for ITS stations, ensuring the fulfilment of QoS requirements while maintaining slice isolation. These parameters, which are disseminated to all ITS-Ss (such as OBUs and RSUs), include service priorities (DCC Profile), transmission rates, and powers. The applicability of a specific ITS-G5 service configuration may vary, ranging from a single RSU's coverage area to larger geographic zones encompassing multiple RSUs. The choice of slicing configuration for each area is influenced by factors like environmental context (urban or highway), ITS station density, and vehicle types (ordinary or priority vehicles).

To align with our objectives, we've incorporated an ITS-S Element Management System (EMS) in the management layer, as detailed in Figure 31. This EMS is a cornerstone of our slicing mechanism, tasked with receiving and implementing slicing configuration parameters. It fine-tunes services by adjusting various parameters across different layers: setting priorities in the management layer, managing traffic control and shaping in the access layer, and configuring parameters for mechanisms like Lucky/Token bucket shaping to achieve specific rates. Additionally, it adjusts the message generation rate in the Facilities layer. The slicing configuration is dynamic, allowing for real-time adjustments based on transmission channel

conditions or updates in QoS parameters. For example, a vehicle could seamlessly transition from Slice 1 to Slice 2 as needed.

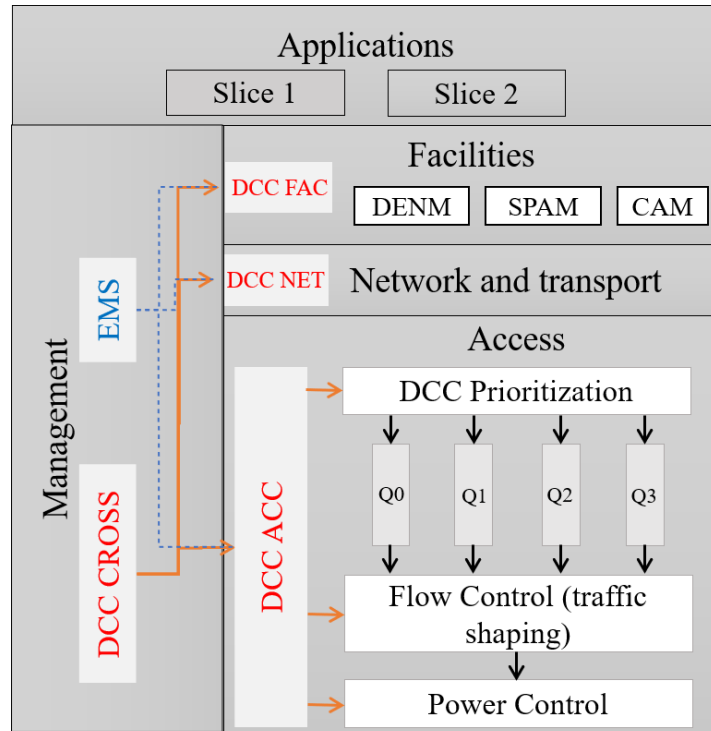


Figure 31. ITS station protocol stack + EMS

#### 4.4.3 Simulation Scenario, Parameters, and Results

This section outlines the simulation scenario utilized to evaluate the slicing approach for ITS-G5. Following this, we delve into an analysis of the results obtained, including assessments of service slicing and isolation testing outcomes.

##### 4.4.3.1 Implementation and Parameters of the E2E ITS-G5 Slicing

Our proposed E2E ITS-G5 slicing architecture and all simulations are implemented and run on the Artery simulator. Different vehicle densities (total number of vehicles in the simulation environment) are used as well. And depending on the density, the inter-vehicle spacing and speed can change. The vehicle flows and mobility is simulated using the traffic simulator SUMO. The simulation parameters are summarized in Table 16.

Table 16: Simulation parameters

Transmission Channel	CCH
ITS-G5 Service	CAM
Transmission rate	5 Hz
RSU Inter-location	750m
Simulation time	200 s

Scenario		8-lane bidirectional highway	
Lane width		3.5m	
Length		2000m	
Vehicle number	Low Density	Dense	High density
	300	450	600
Max speed	15m/s	13.3m/s	
Vehicle length		3m	
Min gap		5m	
Flow Type		Police (HPS) Regular vehicles (OPS)	

In our research, we implement a slicing strategy for ITS-G5, focusing on a high-vehicle-density scenario to demonstrate its effectiveness. The primary goal is to enhance the end-to-end latency for CAMs in high-priority vehicles. We establish two slices for different user groups: a High Priority Slice (HPS) for High Priority CAMs (HP CAMs) and an Ordinary Priority Slice (OPS) for Ordinary Priority CAMs (OP CAMs). Although our model utilizes two slices, it's scalable to support additional services and applications.

The core of this approach lies in prioritizing high-priority messages, ensuring fast transmission by reducing waiting times, and enhancing channel access opportunities. For HPS, we set the priority to DP0 (the highest), corresponding to the highest priority queue Q0 for VoIP traffic. Conversely, OPS uses the default priority (DP2), associated with queue Q2 for the Best effort (BE) traffic class.

The simulation environment is a 2 km bidirectional highway with eight lanes (four in each direction). Police car traffic is mapped to HPS, while regular vehicle traffic is assigned to OPS. HPS traffic varies from 5% to 30%, and OPS traffic from 95% to 70% of the total flow.

To mimic a congested real-world scenario, we introduce two additional services for all vehicles: a DENM service with priority DP1 (mapped to Q1) and a Background (BK) service with priority DP3 (mapped to Q3), both transmitting at 1Hz.

Our primary metric of analysis is the end-to-end latency, assuming identical resources for access, aggregation, and core networks. This latency, crucially linked to the ITS-G5 ad hoc network, is the time elapsed from a vehicle's CAM message transmission at the application layer to its reception by the Roadside Unit (RSU). It encompasses both the propagation delay over the transmission channel and the queuing delay in the system.

#### 4.4.3.2 ITS-G5 RAN Simulation Results

To effectively highlight the contributions of our work, we conducted a simulation scenario using the default CAM priorities as a baseline for comparison. Here, the priority for CAM

messages is set to the default value DP2. The resulting average latencies from various simulations are succinctly summarized in Figure 32 and Figure 33, providing a clear contrast between our proposed slicing approach and the standard priority settings. This comparative analysis serves to underscore the enhancements in latency achieved through our slicing strategy.

Figure 32 demonstrates that in a network with 450 vehicles, the latency for CAM is notably lower in the high-priority slice (HPS) compared to the ordinary priority slice (OPS). While OPS latency varies between  $700\mu\text{s}$  and  $1000\mu\text{s}$ , HPS maintains a more consistent latency, ranging from  $500\mu\text{s}$  to  $600\mu\text{s}$  in both 10% and 20% HPS traffic scenarios. This indicates the superior performance of HPS in terms of latency stability, particularly in comparison with the baseline scenario where OPS latency fluctuates between  $600\mu\text{s}$  and  $1000\mu\text{s}$ .

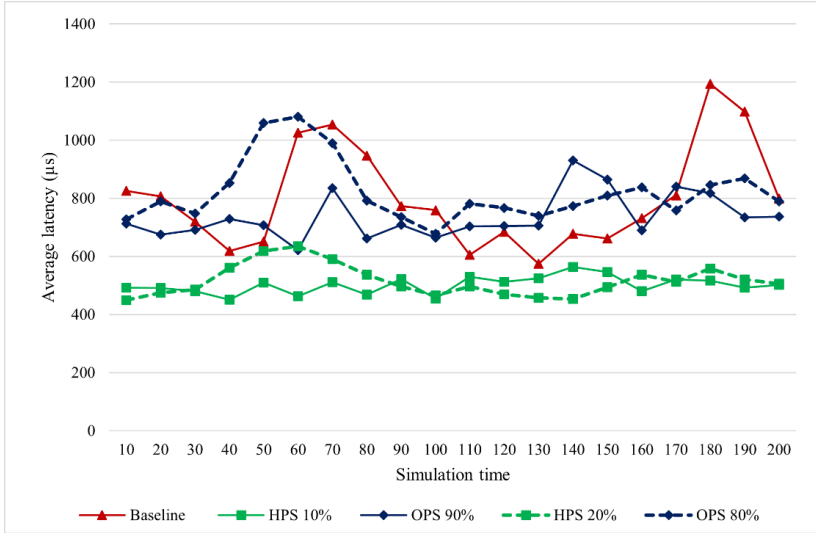


Figure 32. Average RAN latency (vehicle number: 450)

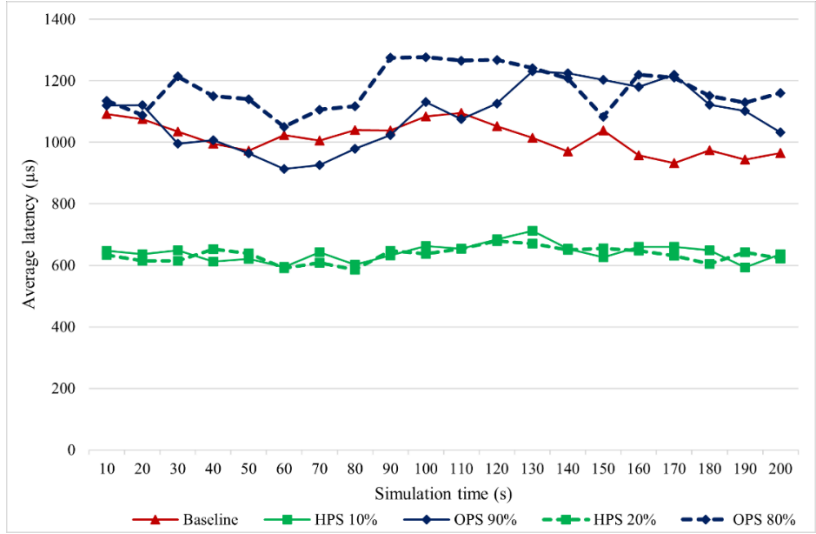


Figure 33. Average RAN latency (vehicle number: 600)

In Figure 33, increasing the vehicle density to 600 highlights the effectiveness of our slicing solution. For both 10% and 20% HPS scenarios, latency is consistently maintained between  $600\mu\text{s}$  and  $700\mu\text{s}$ . In contrast, OPS and baseline scenarios show higher latencies, averaging between  $1000\mu\text{s}$  and  $1200\mu\text{s}$ . This improvement in latency is critical for advanced V2X use cases like advanced driving, where latency exceeding a few milliseconds, specifically 1 ms for autonomous driving, can be detrimental.

In the following, we assessed the isolation property of our network slicing mechanism by overloading the ordinary priority slice (OPS) while monitoring its impact on the high-priority slice (HPS). We increased the CAM generation frequency for the OPS flows to 10Hz compared to HPS, which is still set to 5Hz. We used a worst-case scenario where the vehicle density is set to 600 and OPS flows represent 95% of the total traffic (5% HPS flows). The simulation results are presented in Figure 34 in which HPS\_O and OPS\_O represent the scenario where OPS is overloaded. The obtained results show that the average latency after overloading the OPS does not impact the performance of the HPS. This result strongly affirms the robust isolation capability of our slicing solution, demonstrating its effectiveness in safeguarding HPS performance against heavy traffic loads in OPS, a critical feature for ensuring consistent network performance under varying traffic conditions.

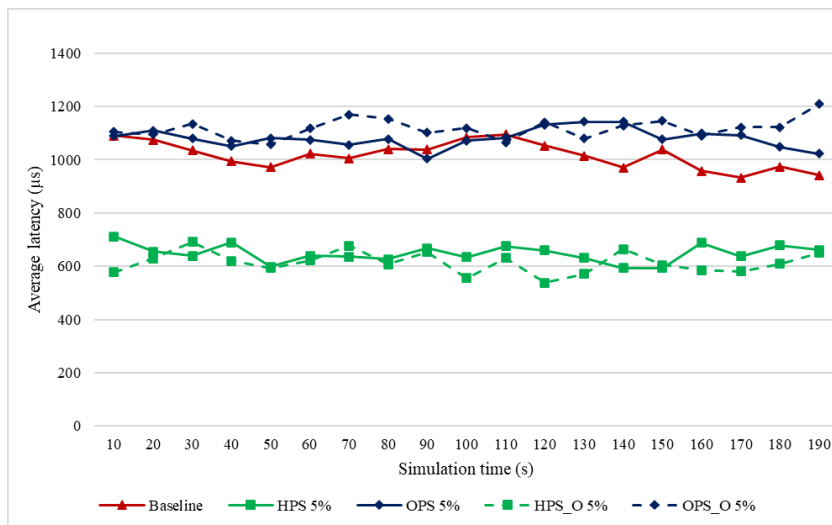


Figure 34. Average RAN latency (vehicle number: 600, HPS: 5%)

#### 4.4.3.3 End-to-End Slicing

The next step of our research is to set up a testbed with two network slices: the High Priority Slice (HPS) and the Ordinary Priority Slice (OPS), extending from ITS G5 RAN slicing (simulation-part) to the core network (real-world). We utilized Open-Source MANO (OSM) release 6, compliant with the ETSI-NFV framework [96], for managing and orchestrating the network slices, including lifecycle management and monitoring. Each slice consists of three VNFs and a V2X application server, all operational within Docker containers. Figure 35

presents an overview of this architecture, showcasing the integration of both simulated and real components in the testbed.

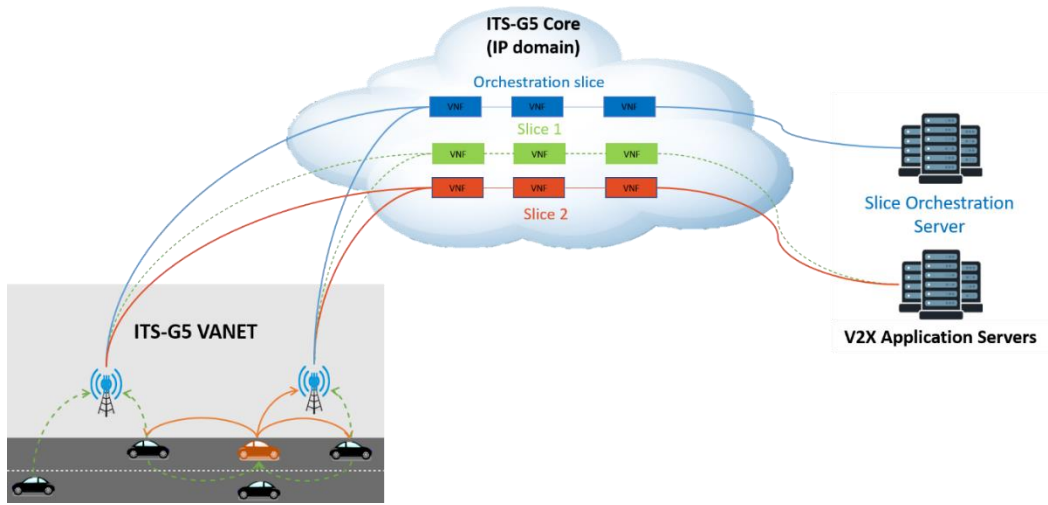


Figure 35. Overview of the E2E deployed architecture

In our experimental setup, we deployed a socket client and server at both endpoints to measure the end-to-end performance of network slicing. This socket was specifically utilized for transmitting CAMs originating from the ITS-G5 network (simulated component). The chosen scenario involved a vehicle number of 600, with the HPS traffic constituting 10% of the total traffic. We employed the TC tool (Linux traffic control [97]) for configuring link parameters, as detailed in Table 17, to effectively demonstrate the prioritization of slices while emulating Wide Area Network (WAN) characteristics.

Table 17: Link parameters configuration

Slice	Maximum link latency in the core	link reliability	% of flows per slice
HPS	50 ms	99%	10%
OPS	200 ms	95%	90%

The results of this simulation are illustrated in Figure 36. We observe an average latency of 40ms for HPS and 160ms for OPS. These findings provide a practical example of how road operators can establish different priority levels through two distinct network slices, thereby implementing a Service Level Agreement (SLA) for V2X services. While our focus was predominantly on latency as the key performance indicator, it's noteworthy that our architecture could incorporate other network slice KPIs to meet the QoS requirements for V2X services.

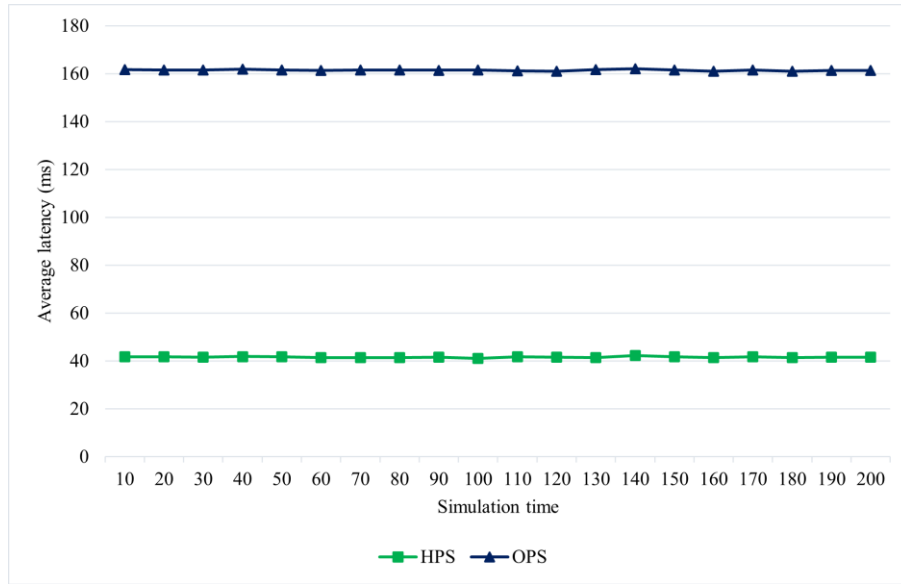


Figure 36. Average E2E latency (vehicle number: 600, HPS: 10%)

#### 4.4.4 Conclusion

In summary, this chapter introduced an architectural framework designed to facilitate end-to-end slicing for Vehicle-to-Everything (V2X) applications within the context of ITS-G5 vehicular communication. The proposed slicing mechanism was developed with the objective of enhancing end-to-end (E2E) latency for specific traffic categories and user groups.

Through our experimentation, we effectively showcased the efficacy of this slicing mechanism by employing two distinct slices: the High Priority Slice (HPS) and the Ordinary Priority Slice (OPS). The results indicated that HPS consistently achieved lower latency compared to OPS. This differentiation in Quality of Service (QoS) based on slice priorities holds significant promise, particularly for advanced V2X applications where minimizing latency is critical, such as in autonomous driving and platooning scenarios.

In conclusion, this architectural approach presents a valuable contribution to the field of V2X communications, offering a means to tailor QoS to the unique requirements of different traffic types and user groups through effective end-to-end slicing.



# 5. Dynamic Service Migration Empowered with Deep Q-Learning for C-ITS

## 5.1 Introduction and Problem Statement

The integration of AI and network slicing in C-ITS extends the optimization of resource allocation in V2X environments and the support for advanced V2X services. These advancements are poised to revolutionize the transportation sector, offering numerous societal benefits [98][99].

Additionally, these advancements facilitate a significant optimization in resource management [100], which is essential for ensuring precise control over QoS and Quality of Experience (QoE). As a result, ITSs are well-equipped to fulfill the diverse and rigorous requirements of various vehicular applications, ranging from critical safety communications to bandwidth-intensive entertainment services [101][102]. Altogether, these technologies form a robust and comprehensive framework, thoughtfully crafted to enhance the efficiency and adaptability of vehicular networks.

However, realizing the full potential of ITS and these novel services, such as video streaming, gaming, and augmented reality experiences, relies heavily upon the ability of vehicles to communicate and access services with minimal latency [103]. Addressing this requires innovative strategies that are acutely sensitive to the unique challenges of mobility and resource constraints within vehicular networks.

The adoption of the Multi-access Edge Computing (MEC) paradigm, which involves deploying servers at the network's edge, has been proposed as a solution to meet these rigorous QoS and QoE demands [104][105]. By efficiently positioning edge servers, MEC extends robust computational capabilities to vehicles, facilitating the execution of computationally demanding applications such as object detection, video stream analytics, and path navigation with reduced latency. This approach significantly enhances the performance and responsiveness of vehicular networks, enabling them to support the high demands of modern vehicular applications.

Nevertheless, MEC-enabled vehicular networks, while offering numerous advantages, confront new challenges in areas such as task offloading and computing, predominantly due to the high mobility of vehicles. This mobility leads to a dynamically shifting communication topology within vehicular networks, resulting in unreliable communication links and potential disruptions [106]. Vehicles frequently moving beyond the coverage area of an edge server exacerbate this issue, leading to increased latency and interruptions in service sessions.

Furthermore, the constant mobility of vehicles can compromise the effectiveness of static AI-based resource allocations and network slices, underlining the need for a dynamic and responsive approach within the MEC infrastructure.

In summary, the primary challenge in fully realizing the capabilities of ITS and MEC in the context of connected and autonomous vehicles lies in devising and implementing novel, mobility-aware strategies. These strategies must effectively address the dynamic nature of vehicular networks, ensuring continuous and reliable service delivery, and maintaining optimal network and computing resource utilization amidst the inherent high mobility of vehicles.

## 5.2 Research Objectives

Service migration stands as a vital solution to the challenges of high mobility in MEC-enabled vehicular networks. It entails the real-time relocation of computational resources and services from one edge server to another within the MEC framework, closely aligning with the movement patterns of vehicles. This dynamic process is crucial for sustaining consistent service quality and meeting the strict latency requirements that are essential for advanced vehicular applications [107]. Service migration thus enables the MEC framework to offer an uninterrupted and continuous experience, ensuring efficient delivery of services and persistent connectivity for vehicles, irrespective of their geographic mobility.

This research is primarily focused on addressing the intricacies of service migration in MEC-enabled vehicular networks, with a particular emphasis on balancing QoS and minimizing associated migration costs. This balance is especially pertinent given the diversity of geographical areas vehicles traverse. The study's objectives are geared toward refining the service migration process to achieve seamless service continuity, minimal latency, and cost-effectiveness in a dynamic vehicular context. The specific goals are outlined as follows:

- **Formulation of the Service Migration Problem as an MDP:** The study seeks to conceptualize the service migration issue in vehicular networks as a Markov Decision Process (MDP). This involves a comprehensive definition of the MDP's states, actions, and reward functions, which are pivotal for pinpointing the most opportune moments and locations for service migrations. A key aspect of this approach is its consideration of different service profiles, which allows for the customization of migration strategies to cater to each service's unique requirements.
- **Implementation of Deep Reinforcement Learning (DRL) Solutions:** A major objective is to leverage DRL methodologies, particularly Deep Q networks (DQN), for an effective resolution of the MDP. The proposed solution integrates a deep Q learning (DQL) approach with a double Q network and a replay buffer. This DRL-based strategy is intricately designed

to balance the trade-off between the costs of migration and latency for various service profiles while considering the vehicles' mobility, computational needs, and edge nodes' available capacities.

- **Evaluation of the DRL-Based Scheme's Performance:** It is important to thoroughly evaluate the efficacy of the proposed DRL-centric migration framework. This evaluation will compare its performance with other existing service migration approaches, with a focus on its capability to maintain optimal latency, minimize migration costs, and ensure uninterrupted service despite the challenges of high vehicular mobility.

Through these objectives, the study endeavours to contribute significantly to the optimization of service migration in MEC-enabled vehicular networks, enhancing their operational efficiency and responsiveness to the dynamic needs of contemporary transportation systems.

### **5.3 Related Works**

Various works have studied service migration in MEC as an MDP, such as in [108][109][110]. Recent research has delved into various strategies for service migration in MEC-enabled vehicular networks, each addressing unique challenges and proposing different solutions. In [111], the authors proposed a State-adaptation Reinforcement learning method for Service Migration (SRSM) in MEC, where they analyse the different states of edge network quantitatively and apply deep Q-learning into migration methods, which can adjust the learning rate adaptively to implement rapid convergence in the learning process. In [112], the service migration problem was addressed using a reinforcement learning-based model that can take a long-term objective into account and implement better service migration and communication decisions.

The study in [113] focuses on seamless service migration for mobile IoT devices within fog computing architectures, introducing the Follow Me Fog (FMF) framework. FMF employs a pre-migration mechanism based on signal strength monitoring to reduce latency, especially during handovers. While it effectively minimizes interruption, this model tends to migrate services to the nearest MEC server, overlooking the potential costs associated with such migrations.

In another significant study [114], researchers proposed the Follow-Me Cloud (FMC), which facilitates service mobility across federated data centres. This approach formulates migration decisions as an MDP, with the distance between the user and the edge server as a key factor. The model aims to find a balance between cost and user-perceived quality, but it

primarily focuses on the geographical distance for making migration decisions, which might not always align with optimal service delivery.

Peng et al. [115] investigated service migration in vehicular networks, emphasizing the balance between QoS and migration costs. They introduced a model that ties QoS and migration costs to vehicle velocities, utilizing deep Q-learning for decision-making. However, this strategy mainly focuses on binary migration decisions and defaults to migrating to the nearest edge server, which may not always be the most efficient choice considering server capacity and network congestion.

Abouaomar et al. [116] explored service migration intending to minimize total service latency and migration costs. They formulated this as a multi-agent MDP, solved through deep Q learning (DQL). Their model uniquely considers vehicular mobility but does not account for the load on edge servers, which can significantly influence migration decisions, especially in dynamic vehicular environments.

In contrast to these studies, our research introduces a novel framework that addresses several key aspects previously unexplored. Firstly, we factor in the capacity of each edge server as a crucial element in migration strategies, going beyond the conventional focus on latency or distance. This approach enables a more balanced distribution of computational resources across servers. Secondly, our framework offers personalized migration strategies for each service profile, acknowledging the diverse requirements and usage patterns of different services. This tailoring of strategies marks a significant departure from the generic approaches seen in earlier models.

Moreover, our work contributes a unique perspective by emphasizing the importance of differences in service profiles in migration strategies. While previous research has not extensively explored this aspect, our framework acknowledges the diversity in services and tailors migration strategies to these specific attributes. By integrating these considerations, our model aims to enhance service delivery efficiency and adaptability, marking a substantial advancement in the field of service migration in MEC-enabled vehicular networks.

## 5.4 System Model

In this section, we present the system model of the service migration problem. In our study, we consider a MEC-enabled vehicular architecture, characterized by a set of Roadside Units (RSUs), each linked to an edge server, denoted as  $n \in N := \{1, 2, \dots, N\}$ . This network configuration is visually represented in Figure 37 [117], providing a clear illustration of the system architecture.

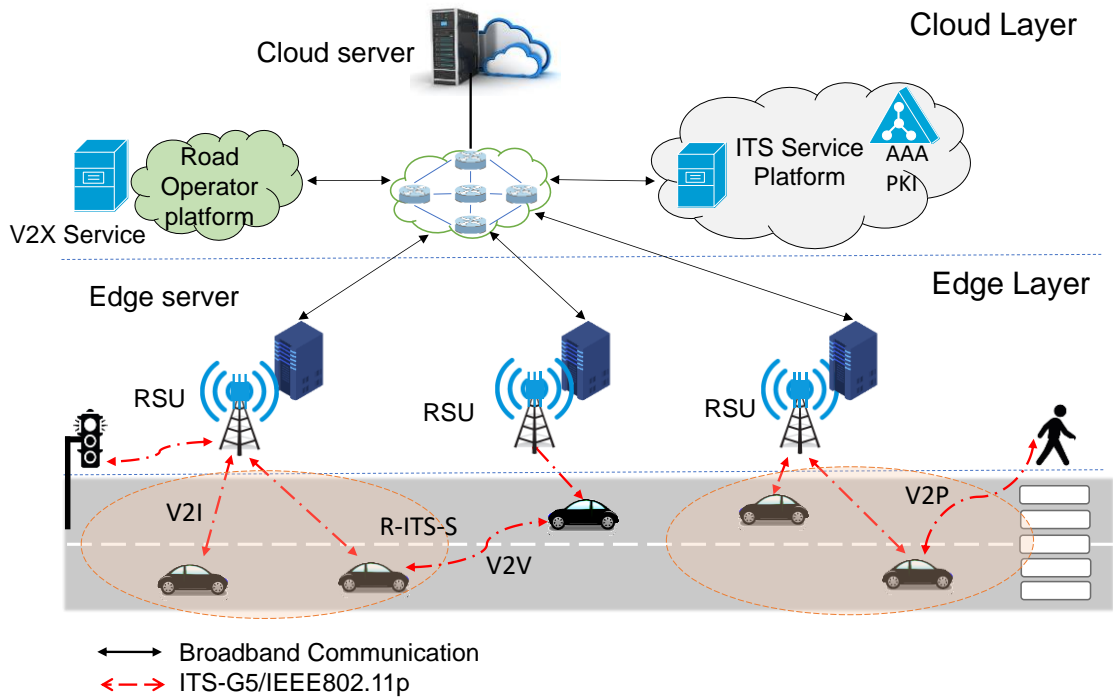


Figure 37. Illustration of the system architecture

Central to this architecture are  $K$  mobile users, typically vehicles, which are the dynamic components of our vehicular network. These users are continuously in motion, accessing a variety of services offered by the Edge servers. Each mobile user has unique service needs, which are encapsulated in what we refer to as a "service profile". A service profile is a detailed descriptor, that captures essential characteristics such as the service's class, required computing and memory capacity, and the size of the service image.

Our system model thus encompasses the interplay among the RSUs with integrated edge servers, mobile users with diverse service demands, and distinct service profiles that define each service's nature. The objective is to develop optimal service migration strategies that consider the vehicles' mobility, the variety of service profiles, and the capacity limitations of the edge servers. The goal is to create a system that efficiently balances quality of service and migration costs while ensuring uninterrupted service delivery in the dynamic environment of vehicular MEC-enabled networks.

Regarding service classification within our vehicular MEC-enabled architecture, we adopt a categorization system based on a comprehensive study [118]. This system classifies services into three distinct classes, each with unique attributes and priority levels:

- Class 0 (Service Profile SP0): This category includes the highest-priority services, demanding ultra-low latency due to their critical nature. Services in Class 0 are crucial for applications where even minor delays can have severe consequences.

- Class 1 (Service Profile SP1): Services in this class hold the second-highest priority. They require efficient handling but can tolerate slightly higher latency levels than Class 0. These services are common in applications needing timely delivery, albeit with a more flexible latency allowance.

- Class 2 (Service Profile SP2): This category is for lower-priority services with relaxed latency requirements. Class 2 services are typically used in applications where immediate delivery is not a critical concern.

This classification, encompassing Service Profiles SP0, SP1, and SP2, offers a structured method to manage the varied needs and priorities of services in our network. It enables us to tailor service migration strategies, ensuring that each service profile receives the necessary level of attention based on its specific requirements and priority.

For ease of reference and clarity, Table 18 in our documentation succinctly summarizes the key notations used in our system model.

*Table 18: System notations*

Parameter	Definition
$d_t$	Distance between the vehicle and the serving edge node.
$H_t$	Edge server hosting the vehicle's service at time slot $t$ .
$N_t$	Prediction of the next EN
$C_t[\text{K}]$	Available capacity of all network edge nodes (CPU, memory, disk)
$SP_t$	Service Profile (vehicle's service class, required capacity by the vehicle's service, service Image Size)
$R_{a_t}^{st}$	Reward function
$D^{\text{com}}$	Communication delay
$R_v$	Data rate between $v$ and its serving BS
$\lambda_n$	Size of the offloaded task of vehicle $v$
$D^{\text{back}}$	Backhaul delay
$R_b$	Backhaul bandwidth
$\mu, \gamma$	Positive coefficients
$\text{dis}(e_s, e_D)$	Distance between the serving edge servers $e_s$ and destination server $e_D$
$D_v^{\text{mig}}$	Migration delay
$z_v$	Service total size (VMs, data, ...)
$R_{q_v}$	Bandwidth of session $q$ for migration of service of vehicle $v$
$D_v^{\text{comp}}$	Computation delay

$u_v$	Number of CPU cycles needed for the offloaded task of vehicle $v$
$y_v^e$	Boolean denotes if the service of vehicle $v$ is hosted by EN $e$
$U_e$	CPU cycle capacity of EN $e$
$C_v$	Monetary delay

### 5.4.1 Network Model

**Communication delay:** Since wireless communications have a substantial impact on the quality of edge-assisted intelligent driving, we examine the communication latency given by:

$$D^{com} = \frac{\lambda_n}{R_v} \quad (1)$$

Where  $\lambda_n$  is the size of the offloaded task of vehicle  $v$ , and  $R_v$  is the data rate between vehicle  $v$  and its serving RSU.

**Backhaul Delay:** When the vehicle is served by a non-local edge server, transmission, propagation, processing, and queuing all contribute to the backhaul latency. The transmission delay is represented by the equation:  $\lambda_n(t)/R_b$ , Where  $\lambda_n$  is the size of the service request of vehicle  $v$  in time slot  $t$ ,  $R_b$  is the bandwidth of the outgoing link of the local edge server  $e$ ,  $\mu$  is a positive coefficient, and  $dis(e_S, e_D)$  distance between  $e_S$  (Source/Local server) and  $e_D$  (Destination/Remote server). The backhaul delay on the edge server is given by:

$$D^{back} = \begin{cases} 0, & \text{if } e_S = e_D \\ \frac{\lambda_n}{R_b} + 2 \mu \text{dis}(e_S, e_D), & \text{else} \end{cases} \quad (2)$$

### 5.4.2 Computational Model

Multiple services share computing resources at each edge server to assist their serving vehicles in performing the requested tasks.  $U_e$  represents the computing capacity of the edge server  $e$ , as measured in CPU cycles per second.

**Computation delay:** Let  $u_v(t)$  denote the number of CPU cycles required by the offloaded task of vehicle  $v$  in slot  $t$ . The computing delay on the edge server is given by:

$$D_v^{comp}(t) = \frac{\sum_{e \in E} u_v(t) y_v^e(t)}{\sum_{v' \in V} y_{v'}^e(t)} \quad (3)$$

Where  $y_v^e$  is a Boolean that denotes if the service of vehicle  $v$  is hosted by edge node  $e$ .

### 5.4.3 Migration Cost Model

The migration of a service across edge servers incurs added operating expenses. We consider that the migration cost depends on the size of each service's image. Let  $C_v$  be the migration cost of moving service  $v$  from the edge server  $e_S$  and  $e_D$ .

$$C_v = \begin{cases} 0, & \text{if } e_s = e_D \\ \gamma z_v, & \text{else} \end{cases} \quad (4)$$

Where  $\gamma$  is a positive coefficient and  $z_v$  is the service total size (VMs, data, ...).

## 5.5 Problem Formulation

In this section, we embark on a rigorous formulation of the service migration problem within the realm of MEC-enabled vehicular networks, casting it into the well-established framework of an MDP. This approach is instrumental in unravelling the sequential decision-making complexities under uncertainty, a characteristic intrinsic to dynamic service migration scenarios. MDPs are adept at integrating diverse factors like mobility patterns, resource availability fluctuations, and varied service profiles, thereby enabling the formulation of a nuanced and optimal migration strategy.

Simultaneously, we introduce the Double Deep Q Network (DDQN) method as a sophisticated, advanced solution specifically tailored to address the multifaceted challenges of this problem. DDQN, a reinforcement learning technique, excels in approximating optimal action-value functions, thus fitting seamlessly into the MDP framework to tackle the intricate decision-making process inherent in-service migration. The subsequent sections will delve deeper into the mechanics of DDQN, detailing its implementation and how it integrates into our comprehensive solution framework.

### 5.5.1 The MDP formulation

In the realm of reinforcement learning, the environment is meticulously formulated as a Markov Decision Process (MDP), which is vividly represented as  $(S, A, P, R, \gamma)$ .

**State Space ( $S$ ) and Action Space ( $A$ ):** In our context, the state space, denoted as  $S$ , encompasses all possible situations or conditions that our system can find itself in. These situations (states) encapsulate critical information like the current state of vehicles, the status of MEC servers, and various other relevant factors. On the other hand, the action space, represented as  $A$ , consists of all the feasible actions that our system can take. These actions dictate when and where service migrations occur.

**Transition Probability ( $P$ ):**  $P$ , symbolizing the transition probability, is the function that outlines the likelihood of transitioning from one state to another when a particular action is taken. This is mathematically represented as  $P(s' | s, a): S \times A \times S \rightarrow [0,1]$ , where  $s$  represents the current state,  $a$  is the chosen action from  $A$ , and  $s'$  denotes the next state. It quantifies the uncertainty and stochasticity inherent in the system's dynamics.



**Immediate Reward ( $R$ ):** The immediate reward, denoted as  $R(s, a, s'): S \times A \times S \rightarrow \mathbb{R}$ , embodies the feedback gained from transitioning between states due to a specific action. It quantifies the immediate benefit or cost associated with the state transition, considering factors like migration cost and latency.

**Discount Factor ( $\gamma$ ):** The discount factor,  $\gamma$ , lies within the range  $[0, 1]$ . It plays a pivotal role in balancing the importance of immediate rewards against long-term gains.  $\gamma$  determines the degree to which future rewards are valued in relation to immediate rewards. A smaller  $\gamma$  prioritizes immediate rewards, while a larger one emphasizes long-term cumulative rewards.

At each time slot, the service provider undertakes the task of collecting relevant states from both vehicles and MEC servers. This information is then processed through the DDQN algorithm, which, in turn, provides optimal action guidance for service migration decisions. The chosen action is executed, leading to the system transitioning to a new state and perpetuating the dynamic decision-making process.

This continuous interaction among the MDP's components:  $S$  (State Space),  $A$  (Action Space),  $P$  (Transition Probability),  $R$  (Immediate Reward), and  $\gamma$  (Discount Factor) forms the backbone of our intelligent service migration strategy, enabling the system to adeptly adjust and optimize its actions within a complex and evolving vehicular network environment.

### 5.5.1.1 State Space

We define the state space  $S = \{s_t | t = 1, 2, \dots, t_{Max}\}$  where a state at time slot  $t$ :  $s_t$ , is a 4-tuple given by:

$$s_t = \{d_t[\mathbf{K}], H_t, C_t[\mathbf{K}], SP_t\} \quad (5)$$

Where:

**Distance Vector  $d_t[\mathbf{K}]$ :** This component signifies the distance between the vehicle and the  $K$  edge servers at the specific time slot  $t$ . Here,  $K$  stands for all edge servers available in the system. By including this parameter in the state, our model inherently considers the spatial relationship between vehicles and edge servers at any given moment.

**Hosting Edge Server  $H_t$ :** At each time slot  $t$ ,  $H_t$  designates the Edge server that is hosting the vehicle's service. This element captures the dynamic aspect of service hosting and highlights the current server responsible for serving the vehicle's needs.

**Server Capacity  $C_t[\mathbf{K}]$ :** Reflecting the available capacity on all the  $K$  edge servers (comprising CPU, memory, and storage).  $C_t[\mathbf{K}]$  serves as a critical piece of information. This parameter acknowledges the computational and resource capacity status of each edge server at the given time slot, reflecting the system's resource utilization.

**Service Profile  $SP_t$ :** This multifaceted parameter encompasses various facets of the service profile:

- **Service Class:** It classifies services into three distinct categories, namely class 0 (SP0), class 1 (SP1), or class 2 (SP2). These classifications help differentiate services based on their priority and stringent requirements, such as latency.
- **Required Capacity:** This parameter characterizes the computing and memory capacity demanded by the specific service at that instant. It enables the model to account for the resource prerequisites of the service.
- **Service Image Size:** The size of the service image, a vital characteristic, is also embedded within this parameter. It provides insights into the data volume associated with the service.

By encompassing all these elements within our state space definition, our model is equipped to function as a multi-criteria migration strategy. It comprehensively considers factors such as vehicle-server distance, vehicle velocity (implicitly through distance changes over time), edge server loads, service capacity requirements, and most notably, the service's class. This holistic approach allows our model to dynamically adapt its migration strategy, ensuring that service migrations are not merely based on a single criteria but are orchestrated to balance a multitude of factors, ultimately optimizing the vehicular network's performance.

### 5.5.1.2 Action Space

Our action space represents the set of decisions available to our model, specifically focusing on whether to migrate a service to a particular edge server at a given time slot  $t$ . Thus, at each time slot  $t$ , we define an action as:

$$a_t = \{A_1, A_2, \dots, A_n\}$$

where  $A_i$  denotes the action of migrating the service to edge server  $e_i$  at time slot  $t$ . Each  $A_i$  is binary, i.e.  $A_i \in \{0,1\}$  and at every timestamp:  $\sum A_i = 1$ .

This action space entails:

**Decision to Migrate:** Each element  $A_i$  in the action space is a binary decision. It signifies whether the service should be migrated to edge server  $e_i$  at that particular time slot  $t$ . A value of 1 indicates that migration should occur, while 0 signifies no migration. This binary representation offers a clear and actionable choice for the model.

**Mapping to Edge Servers:** The set of actions  $A_i$  is directly linked to the edge servers in the system. Each action  $A_i$  corresponds to a specific edge server  $e_i$ . Therefore, the action space encapsulates decisions regarding where the service should be migrated.

**Migration Timing:** The action space's temporal aspect is embedded in the time slot  $t$ . It ensures that the migration decisions are time-sensitive, considering the evolving dynamics of the vehicular network.

**Control Over Migration:** By defining the action space in this manner, our model gains control over the migration process. It can make informed decisions regarding the optimal edge server destination for service migration at each time slot.

No migration will occur during time slot  $t$  if  $H_t=j$  is the hosting edge server and  $A_j=1$ .

This action space representation empowers our model to make decisions about both the timing and destination of service migrations. It allows for dynamic and context-aware migration decisions, ensuring that services are migrated when and where it makes the most sense based on the evolving conditions of the vehicular network.

### 5.5.1.3 Reward Function

In reinforcement learning, the reward function plays a pivotal role as it guides the agent's actions. Our goal is to achieve optimal latency for each service profile while minimizing migration costs. To achieve this, we define the reward function as a combination of latency and migration cost at time slot  $t$ , given by:

$$R_{a_t}^{S_t} = -((1 - \omega)D(t) + \omega Cost(t)) \quad (6)$$

Where  $D(t)$  is the sum of communication delay, backhaul delay, and computing delay, i.e.,  $D(t) = D^{com} + D^{back} + D_v^{comp}(t)$ . And  $Cost(t)$  is the migration cost  $C_v$  of service  $v$  at time slot  $t$ .  $\omega$  is the weight factor, and  $\omega \in [0,1]$ .

## 5.6 Proposed DDQN

DQN (Deep Q-Network) stands as a robust and efficient algorithm within the realm of Deep Reinforcement Learning (DRL). In DQN, the Q-values  $Q(s, a, \theta)$  for each state-action pair  $(s, a)$  are approximated using a Deep Neural Network (DNN), with  $\theta$  denoting the parameters of the Q-network. The training methodology for the DNN incorporates the experience replay memory mechanism, which involves periodically storing the experiences of the MEC agent in a replay buffer. These experiences, encompassing the current state, next state, chosen action, and the resulting reward, are pivotal for the learning process. The key advantage of this mechanism is its provision of uncorrelated data inputs, effectively reducing undesirable temporal correlations.

The DDQN, an advanced iteration of DQN, enhances the training efficiency and reliability by employing dual DNNs, as outlined in [119]. The primary network, known as the main Q-

network, is responsible for calculating the Q-values. Concurrently, the secondary network, termed the target Q-network, generates the target Q-values  $Q(s, a, \bar{\theta})$ . These target values are crucial for training the parameters  $\theta$  of the main Q-network. The specific steps of this training process are detailed in Algorithm 1.

---

**Algorithm 1:** Deep Q-learning algorithm for service migration

---

```

Initialize main DQN  $Q(s, a, \theta)$  with random weights  $\theta$  ;
Initialize Target DQN  $Q(s, a, \bar{\theta})$  with weights  $\bar{\theta} \leftarrow \theta$  ;
Initialize replay memory D to capacity N ;
for each episode do
    Observe the initial state  $s_0$  stat : Servers states, Vehicle-server distance ...;
    for each time slot  $t$  do
        The controller acquires information about vehicle, Service, servers;
        if random number  $< \epsilon$  then
            | Select action  $a_t = \operatorname{argmax}_a Q(s_t, a_t, \theta)$  ;
        else
            | Randomly select  $a_t$  ;
        end
        Execute action  $a_t$  , observe reward  $r_t$  and next state  $s_{t+1}$  ;
        Store the tuple  $\langle s_t, a_t, r_t, s_{t+1} \rangle$  in D ;
        Randomly Sample a minibatch of tuple  $\{\langle s_j, a_j, r_j, s_{j+1} \rangle\}_{j=1}^J$  from D ;
        for each tuple  $j=1..J$  do
            if episode terminate at step  $j+1$  then
                |  $y_j = r_j$  ;
            else
                |  $y_j = r_j + \lambda \operatorname{argmax}_a Q(s_{j+1}, a_{j+1}, \bar{\theta})$  ;
            end
        end
        Perform a gradient decent step on  $\text{Loss}(\theta)$  with respect to network parameters  $\theta$  ,
        where loss function is  $\text{Loss}(\theta) = (1/j) \sum [y_j - Q(s_j, a_j, \theta)]^2$  ;
        Update  $\bar{\theta} = \theta$  after K steps ;
        Terminate when the vehicle is out of simulation region;
    end
end

```

---

Training in each episode spans multiple time slots (or steps). Within this period, every MEC agent closely observes the state of its environment at each step and selects an action  $a_t$  from its action space. The selection of actions is governed by the  $\epsilon$ -greedy policy, wherein an action is chosen randomly with a probability of  $\epsilon$ . Following this, the agent receives its reward and transitions to the subsequent state. The experiences thus gathered are stored in the replay buffer, forming the basis of a training dataset. This dataset then facilitates the training process, which aims to minimize the loss function given by:

$$Loss(\theta) = \left(\frac{1}{j}\right) \sum [y_j - Q(s_j, a_j, \theta)]^2 \quad (7)$$

Where  $Q(s_j, a_j, \theta)$  is the Q-value of action  $a_j$  given in the state  $s_j$  which is calculated using the main Q-network with parameter  $\theta$ . And  $y_j$  is the target Q-value which is calculated using the target Q-network with parameter  $\theta$ .

To update the parameters  $\theta$  of the main Q-network, the agent employs a gradient descent step, a methodical approach to refine the model's accuracy. This refinement is crucial for aligning the network's predictions with the expected outcomes. Subsequently, the parameters  $\theta$  of the target Q-network are updated at regular intervals, specifically every  $K$  steps, through a process of synchronizing them with the parameters  $\theta$  of the main Q-network. This periodic update ensures that the target network gradually adapts to the evolving main network, maintaining a stable learning trajectory.

In the second phase, which is the inference phase of the DQL (Deep Q-Learning), the MEC agent utilizes the optimally trained DDQN. Here, the focus shifts from training to application, leveraging the trained network's optimal parameters to find the most effective migration strategy. During each step of this phase, the MEC agent assesses the current state of its environment and selects actions that maximize the Q-value as determined by the trained DDQN. This step is vital for efficiently resolving the service migration issue, focusing on pinpointing the best times and places for migration to optimally balance latency minimization and cost reduction.

## 5.7 Simulation and Evaluation

### 5.7.1 Simulation Environment

In our study, we utilize a simulated environment to evaluate the effectiveness of our proposed service migration strategy. This environment is composed of various key components and tools designed to facilitate accurate and lifelike simulations.

Central to our simulation setup is OMNeT++. In conjunction, we utilize the Artery framework [71], complemented by SUMO [75], to simulate communications and vehicular movements within the vehicular network. The integration of Artery is crucial for accurately mirroring the mobility patterns of vehicles and establishing realistic communication links between vehicles and Roadside Units (RSUs). This combination significantly contributes to the authenticity and precision of our simulations.

For the implementation of the DDQN agent, we turn to Python and the Keras library. Python offers a flexible and versatile environment for developing intelligent agents, while Keras simplifies the implementation of deep reinforcement learning algorithms.

Within this simulated setup, we create a scenario of a MEC-enabled vehicular network, as illustrated in Figure 38. This scenario involves deploying five edge servers along a highway. The servers are evenly distributed, ensuring a consistent and equal distance between each. In addition, Roadside Units (RSUs) are methodically positioned every 800 meters along the highway and are each connected to the nearest edge server, thus facilitating effective communication between vehicles and the edge servers.

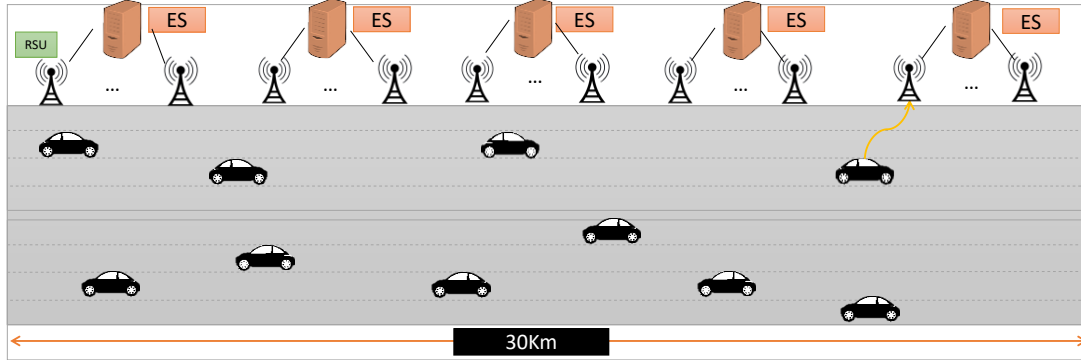


Figure 38. Simulation scenario

The highway in the simulation stretches over 30 Km, offering a realistic space for vehicle movement and service requests. To closely mirror real-world traffic conditions, the vehicles in the simulation travel at random but consistent speeds, chosen from a range between 60 and 110 Km/h. This speed range is representative of the diverse speeds observed in actual traffic. The DDQN and vehicular network parameters are presented in Table 19.

Table 19: Simulation parameters

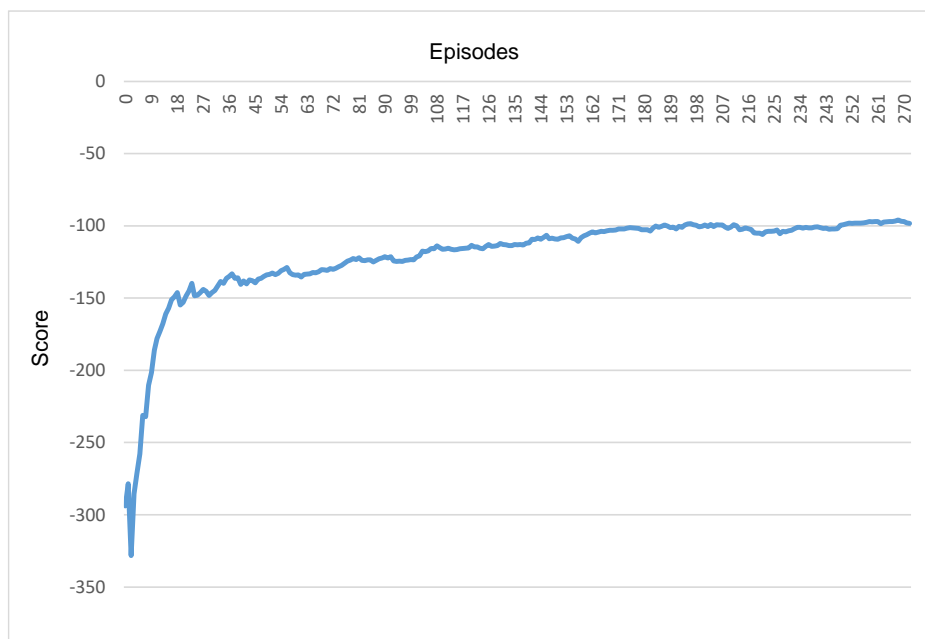
Parameter	Value
Number of edge servers	5
RSU Transmission power	47,9 mW
Vehicles transmission power	200 mW
Access Technology	IEEE 802.11p
Learning rate	$5e^{-4}$
Learning rate	0,99
Replay memory size	1000000
Mini-batch size	64
Target update interval	100
Optimizer	Adam
RNN hidden layers	Two hidden layers of 256 neurons each.
Activation function	ReLU

## 5.7.2 Simulation results and analysis

### 5.7.2.1 Average Reward Per Episode

Figure 39 illustrates the average reward per episode obtained in our simulations, a crucial metric for assessing the effectiveness of our DDQN algorithm in mastering optimal service migration decisions. The figure reveals a pattern of steady enhancement in rewards as the number of training episodes increases. This trend is a key indicator of the DDQN algorithm's successful performance. It demonstrates that the MEC agent, under the guidance of the DDQN, is effectively learning and adapting to the dynamic conditions of the vehicular network environment. With the accumulation of training episodes, the agent's capacity to make well-informed service migration decisions is notably enhanced.

An important observation is the point of convergence for the DDQN algorithm, occurring at around 190 episodes. This convergence signifies that the MEC agent has effectively learned to navigate the decision-making landscape, achieving a state of optimal learning. Such convergence reflects the agent's capability to efficiently explore and exploit the environment, thereby enabling it to make increasingly informed and effective service migration decisions.



*Figure 39. Average reward of MEC agent*

In summary, the results presented in Figure 39 strongly affirm the competence of our proposed DDQN algorithm in training MEC agents for intelligent service migration decision-making. The convergence observed is indicative of the agent's ability to explore various actions and eventually adopt strategies that lead to superior rewards. This implies enhanced service efficiency in terms of latency, reduced migration costs, and improved service continuity.

### 5.7.2.2 Comparison with Alternative Migration Strategies

To conduct a comprehensive evaluation of our proposed service migration strategy, we undertook a comparative analysis against a diverse array of alternative approaches. These approaches represent a wide range of strategies within the context of service migration in MEC-enabled vehicular networks. Each strategy provides a distinct approach to service migration, offering varied insights and methodologies. The strategies selected for this comparative analysis encompass:

- **Random Migration (RM):** In this approach, service migration occurs randomly at each time slot, devoid of any predefined criteria.
- **Never Migrate (NM):** This strategy entails hosting the service on one of the edge servers throughout the simulation, with no migration taking place.
- **Always Migration (AM):** This schema, proposed in [114], dictates that the service should migrate to the nearest edge server. However, we augment this approach by considering the load of the edge server. If the nearest server operates at 100% capacity, migration does not occur.
- **Peng et al. approach [115]:** This approach tackles the complexities of dynamic service migration in MEC-enabled vehicular networks through the lens of deep reinforcement learning. It is primarily aimed at striking a balance between maintaining a high Quality of Service (QoS) and minimizing migration costs amidst the challenges posed by high vehicle mobility. Nonetheless, this strategy primarily hinges on binary migration decisions, defaulting to automatic migration to the nearest edge server without taking into account specific migration destinations. Such an approach, while straightforward, may overlook potential optimizations that could be derived from considering additional variables such as server capacity and network congestion. This oversight could lead to missed opportunities in further enhancing the efficiency of service delivery within the network.

Our simulations encompassed a range of service profiles, each denoting a class of service, specifically SP0, SP1, and SP2. SP0 designates high-priority services with stringent latency requirements, while SP1 and SP2 correspond to services with progressively lower sensitivity to latency.

### 5.7.2.3 Migration Cost Analysis

Figure 40 presents the outcome of our migration cost analysis for service sized at 5 units, excluding the NM method since it incurs no migration cost.

The data vividly illustrates the efficacy of our Double DDQN method, particularly in managing migration costs for various service profiles. In Service Profile 0 (SP0), which is characterized by stringent latency requirements, our model initiates frequent service migrations. This strategy is designed to position the service optimally in proximity to the vehicle, thereby enhancing latency performance. Although Peng et al.'s approach demonstrates marginally lower



costs in SP0, it is in Service Profiles 1 and 2 (SP1 and SP2) where our model truly excels. With these profiles, characterized by lower latency sensitivity, our model significantly outperforms other methods in cost reduction. This is accomplished without any compromise to the QoS, showcasing the adaptability and efficiency of our migration strategy.

Overall, the results highlight the flexibility and robustness of our DDQN-based migration strategy. It adeptly balances migration costs with the imperative of maintaining high QoS across varied service profiles. This strategy ensures cost-efficiency while catering to the specific needs of each service category. In essence, it presents a dynamic and resource-optimized solution that maintains high QoS standards while minimizing migration-related expenses.

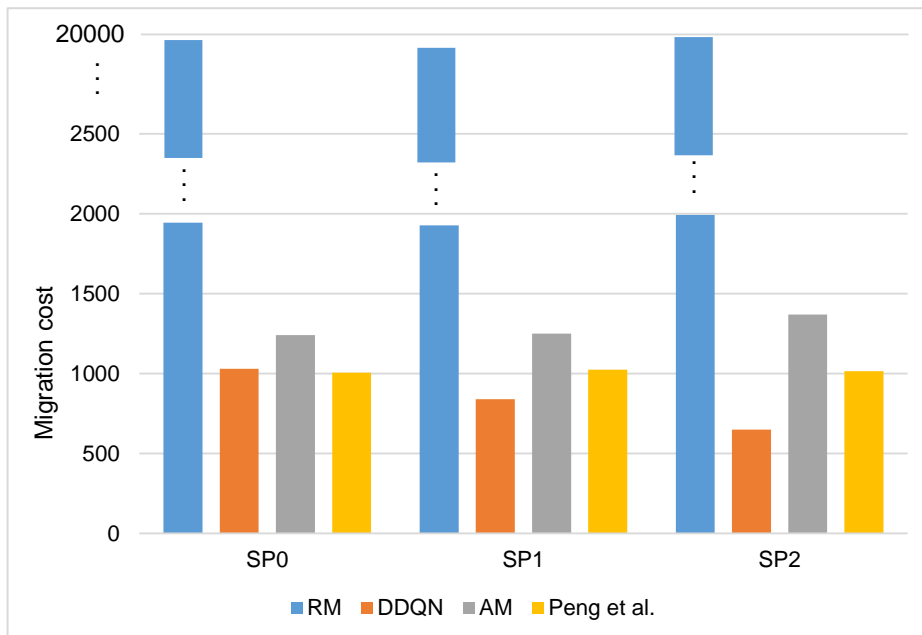
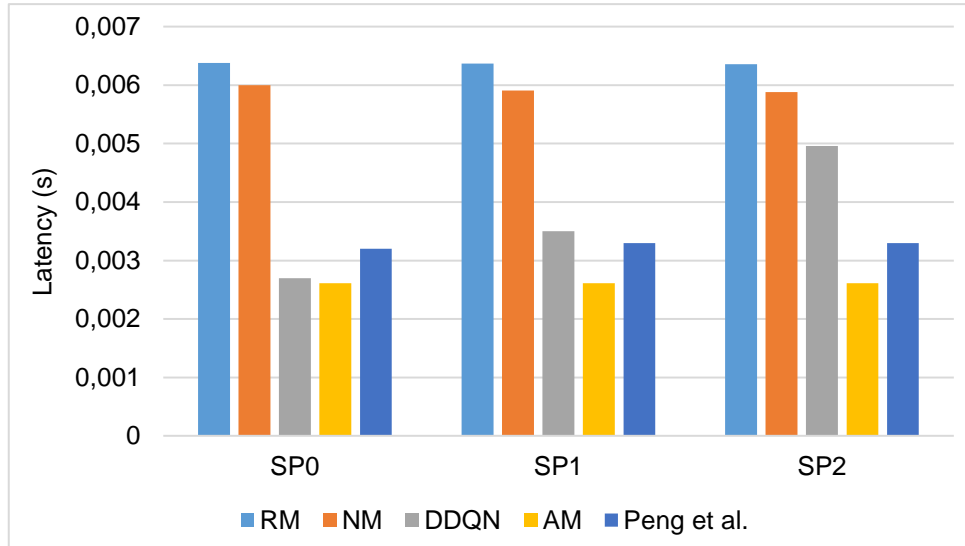


Figure 40. Migration cost results

#### 5.7.2.4 Latency analysis

Figure 41 reveals the latency performance of our model across various service profiles, providing insightful data into its operational efficacy. This figure shows an increase in latency for Service Profiles 1 and 2 (SP1 and SP2) compared to Service Profile 0 (SP0). This pattern is attributed to the differing latency tolerances inherent to each service profile. SP0, which demands exceptionally low latency, understandably records the lowest latency figures. However, it is crucial to note that achieving this reduced latency comes with higher migration costs, as detailed in Figure 40.



*Figure 41. Latency results*

The versatility of our model is clearly evident in these results. It adeptly adjusts its operational approach to align with the unique latency requirements of each service profile. By allowing for marginally higher latency in SP1 and SP2, the model successfully curtails migration costs. This strategic flexibility underscores the model's capacity to optimally balance migration costs against latency demands for each service category.

Compared to other methods like RM, NM, and Peng et al., our DDQN model demonstrates superior performance, especially in handling SP0. The Always Migrate (AM) strategy, while achieving the lowest latency due to its constant, proximity-based service migrations, results in substantially higher migration costs. This scenario underscores the inherent trade-off between latency and migration costs, a critical aspect in making informed service migration decisions in MEC-enabled vehicular networks.

In summary, our approach successfully demonstrates a high degree of effectiveness in achieving an adaptive and efficient balance between migration cost and latency in MEC-enabled vehicular networks. The results distinctly showcase the flexibility of our DDQN-based migration strategy, which adeptly meets the varying demands of different service profiles. It intelligently modulates its migration decisions to accommodate increased latency in profiles with lower latency sensitivity (SP1 and SP2), thereby achieving notable reductions in migration costs.

The adaptability of our approach is a crucial factor in optimizing service migration in dynamic vehicular environments. By carefully balancing migration costs with latency, our method ensures not only cost-effective service migration but also maintains satisfactory QoS levels. It outshines competing strategies by delivering superior latency performance in critical

low-latency scenarios (SP0) and concurrently offers significant cost advantages over proximity-based migration strategies like Always Migrate (AM).

Ultimately, our approach stands as a valuable tool for MEC-enabled vehicular networks, where effective service migration optimization is paramount. Capable of responding to the distinct requirements of various service profiles, it offers an efficient, cost-effective solution to the complexities of service migration in the ever-evolving landscape of vehicular networks.

## 5.8 Conclusion

In this chapter, we presented a comprehensive solution addressing the complexities of vehicular service migration in MEC-enabled networks. Our initial step involved defining this multifaceted problem within the framework of a Markov Decision Process (MDP), a methodology that allowed us to intricately consider factors such as vehicle mobility, server capacity, and service profiles. Integrating service profiles into our model was a pivotal move, enabling us to devise migration strategies that are finely attuned to the distinct needs of each service.

Our proposed solution harnessed the power of Deep Reinforcement Learning (DRL), utilizing a double Q network alongside a replay buffer to significantly enhance the learning process. Through comprehensive simulations, we have rigorously validated the efficiency of our DQL-based algorithm. This algorithm has proven its proficiency in navigating the delicate balance between latency and migration cost, adeptly adapting to the specific requirements of different service profiles. This ability to tailor its approach based on the distinct demands of each service profile stands as a testament to the versatility and effectiveness of our proposed solution in the dynamic landscape of MEC-enabled vehicular networks.

## 6. General Conclusion

This thesis has unfolded within the framework of Cooperative Intelligent Transport Systems (C-ITS) and Vehicle-to-Everything (V2X) communications, fields that are experiencing rapid evolution due to technological advancements. This research addresses key challenges, including channel congestion in vehicular networks, the essential requirement for ultra-low latency and high reliability in V2X communications, and the integration of ground-breaking technologies like AI, MEC, and network slicing. Particularly within the European context, where the ITS-G5 standard for Vehicular Ad hoc Networks (VANETs) plays a central role, these challenges are significant barriers to the advancement of intelligent transportation systems.

### 6.1 Summary and Key Contributions

The first contribution of this thesis addresses the pervasive challenge of channel congestion in C-ITS and V2X networks, particularly under the European standard ITS-G5. The objective was to shift from traditional reactive and adaptive congestion control methods to a proactive stance. This innovation aimed to predict and manage congestion before it could impact network performance adversely. The validation of this approach involved extensive simulations in high-density traffic scenarios, which are typically prone to congestion. These simulations demonstrated that the proactive DCC algorithms could effectively anticipate and alleviate congestion, leading to smoother traffic flow and enhanced network efficiency. The findings revealed a significant improvement in traffic management and safety protocols, establishing the proactive DCC method as a viable and effective solution for real-world vehicular networks.

The thesis' second significant contribution focused on implementing RAN slicing within ITS-G5 networks. The main goal was to optimize resource allocation and traffic prioritization in highly congested vehicular networks. This was achieved by designing a novel RAN slicing architecture that could create multiple network slices, each tailored to specific traffic types and priorities. The architecture's effectiveness was validated through rigorous simulations that mimicked real-world traffic conditions. These simulations underscored the architecture's capability to efficiently manage diverse communication demands, ensuring optimal Quality of Service (QoS) across different slices. The results showed marked improvements in network performance, particularly in handling high-priority traffic, thereby underscoring the potential of RAN slicing in enhancing vehicular communication systems.

Building on the concept of RAN slicing, the third contribution expanded its application to an end-to-end context within ITS-G5 networks. The aim was to develop an E2E network slicing framework that could ensure high-quality communication channels for various vehicular

services, especially in congested environments. This novel framework was put to the test through detailed simulations designed to replicate the complexities of modern vehicular networks. The simulation results were pivotal, demonstrating the framework's ability to maintain ultra-low latency and high reliability for critical vehicular applications. This was particularly evident in scenarios that demanded rapid and dependable communication, validating the E2E network slicing framework as an effective solution for the future of vehicular networks.

The fourth and final major contribution of the thesis explored the utilization of Adaptive Deep Reinforcement Learning for service migration in MEC-enabled vehicular networks. The objective was to develop a strategy that could dynamically adapt to the high mobility of vehicles and shifting communication topologies, optimizing service continuity while minimizing latency and migration costs. This strategy was thoroughly validated through a series of complex simulations that modelled the dynamic nature of vehicular networks. The simulations revealed that the DRL approach, particularly the Deep Q Networks (DQN) and Double Deep Q-Networks (DDQN), was highly effective in managing service migration. The findings showed a notable improvement in maintaining service quality and reducing disruptions, even in highly mobile environments, establishing this approach as a groundbreaking advancement in MEC-enabled vehicular network management.

These contributions, validated through comprehensive simulations and evaluations, collectively enhance the safety, efficiency, and reliability of C-ITS and V2X communications, addressing critical challenges and setting new benchmarks in the field.

## **6.2 Future Directions and Potential Developments**

The thesis has established a foundational framework in C-ITS, highlighting the need for innovative network architectures. Future research in the field of Cooperative Intelligent Transportation Systems (C-ITS) and V2X communications should focus on developing an advanced End-to-End (E2E) slicing architecture that integrates ITS-G5 and C-V2X standards. This architecture is expected to enable robust interoperability, adaptability, and functionality, particularly in complex traffic scenarios. The aim is to create a network framework that efficiently supports the diverse and dynamic nature of modern vehicular communications, ensuring seamless and efficient connectivity across various vehicular systems.

Another promising area of research is the integration of AI and machine learning in network management and congestion control within vehicular systems. The adoption of AI-based DCC systems, combined with the expansion of Multi-access Edge Computing (MEC) and the integration of upcoming 5G/6G technologies, offers a groundbreaking opportunity to enhance network performance. This approach is poised to optimize network efficiency and resource

allocation, thereby bolstering the reliability and functionality of vehicular networks, especially for high-demand applications.

The extension of ITS architecture to encompass bicycles, scooters, and railways represents a significant advancement towards a more inclusive and comprehensive transportation system. This extension will necessitate adapting service migration strategies and edge server placements to accommodate the unique communication requirements of these varied modes of transport. Such an expansion has the potential to revolutionize urban transportation by creating a cohesive network that integrates diverse forms of mobility.

Further, the evolution of current service migration models to more complex real-world scenarios is essential. Transitioning to a two-dimensional (2D) scenario and developing a multi-agent Markov Decision Process (MDP) framework will enhance the scalability and applicability of these models. This advancement is expected to lead to more efficient network resource management in dynamic vehicular environments through collaborative decision-making among edge servers.

Research in the area of NR-V2X is crucial for enhancing road safety. Investigating the interoperability of NR-V2X with existing vehicular communication technologies, such as ITS-G5 and LTE-V2X, and exploring hybrid communication frameworks that combine NR-V2X and LTE-V2X technologies, alongside ITS-G5, are vital for a comprehensive analysis and improvement of road safety mechanisms. This research direction is expected to leverage the advanced capabilities of 5G/6G technologies to enhance road safety significantly.

Addressing the challenges in hybrid vehicular communication systems is another critical area of focus. Developing robust protocols and algorithms for seamless interoperability between different communication technologies, and efficient radio resource management in scenarios where multiple technologies coexist in the same frequency band, are imperative. Such research aims to maximize bandwidth utilization and improve the overall efficiency and effectiveness of vehicular communication networks.

Finally, comprehensive testing of hybrid vehicular communication systems in a variety of environmental and traffic conditions is fundamental. Extensive real-world testing, including field trials and simulations in urban and rural settings, different weather conditions, and varying traffic densities, is essential. The insights from these tests will provide an invaluable understanding of the practical advantages and limitations of hybrid communication systems, guiding future improvements and implementations in vehicular communication networks.

## References

- [1] Available at: [https://leadingedgedc.com/blog\\_post/regional-smart-cities-edge/](https://leadingedgedc.com/blog_post/regional-smart-cities-edge/) (accessed in October 2023)
- [2] Available at: <https://www.digi.com/blog/post/introduction-to-smart-transportation-benefits> (accessed in October 2023).
- [3] ETSI, «ETSI EN 302 665 V1.1.1 Intelligent Transport Systems (ITS); Communications Architecture» 2010.
- [4] ETSI, «ETSI 102 636 V1.2.1 Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Subpart 2: Media-dependent functionalities for ITS-G5» 2013.
- [5] Hope Bovenzi (2019). How connected vehicles leverage data: 3 common questions [online]. Available at: [https://e2e.ti.com/blogs\\_/b/behind\\_the\\_wheel/posts/how-connected-vehicles-leverage-data](https://e2e.ti.com/blogs_/b/behind_the_wheel/posts/how-connected-vehicles-leverage-data)
- [6] Hasan, Monowar, et al. "Securing vehicle-to-everything (V2X) communication platforms." *IEEE Transactions on Intelligent Vehicles* 5.4 (2020): 693-713
- [7] Shrestha, Rakesh, et al. "Evolution of V2X communication and integration of blockchain for security enhancements." *Electronics* 9.9 (2020): 1338.
- [8] TransAID Consortium. (2019). TransAID D5.1 Definition of V2X Message Sets. Available at: [https://www.transaid.eu/wp-content/uploads/2017/Deliverables/WP5/TransAID\\_D5.1\\_V2X-message-sets.pdf](https://www.transaid.eu/wp-content/uploads/2017/Deliverables/WP5/TransAID_D5.1_V2X-message-sets.pdf)
- [9] ETSI, «5G; Service requirements for enhanced V2X scenarios (3GPP TS 22.186 version 17.0.0 Release 17)» 04/2022.
- [10] M. H. C. Garcia et al., "A Tutorial on 5G NR V2X Communications," in *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1972-2026, thirdquarter 2021, DOI: 10.1109/COMST.2021.3057017.
- [11] Mendiboure, Léo. *Distribution géographique de données dans l'Internet des Véhicules: une approche logicielle et sécurisée utilisant les réseaux cellulaires*. Diss. Université de Bordeaux, 2020.
- [12] ERTICO - ITS Europe, Version 1.1: "Report about technologies for future C-ITS service scenarios". 2015-03. Harounabadi, Mehdi, et al. "V2X in 3GPP standardization: NR sidelink in release-16 and beyond." *IEEE Communications Standards Magazine* 5.1 (2021): 12-21.
- [13] Mosbah, Mohamed. "Analysis and Comparison of IEEE 802.11 p and IEEE 802.11 bd."
- [14] Festag, A. Standards for vehicular communication—from IEEE 802.11p to 5G. *Elektrotech. Inftech.* 132, 409–416 (2015). <https://doi.org/10.1007/s00502-015-0343-0>
- [15] Bazzi, Alessandro, et al. "Survey and perspectives of vehicular Wi-Fi versus sidelink cellular-V2X in the 5G era." *Future Internet* 11.6 (2019): 122.
- [16] N. S. a. R. G. D. Eckhoff, «A performance study of cooperative awareness in ETSI ITS G5 and IEEE WAVE» chez 10th Annual Conference on Wireless On-demand Network Systems and Services (WONS), Banff, AB, Canada, 2013.

- [17] Haliti, «Update and Evaluate Vehicular Simulation Framework for LTE and 802.11 p in OMNeT++» pp. 7-8, 2018.
- [18] V. & K. K. Khairnar, « Performance of Vehicle-to-Vehicle Communication using IEEE 802.11p in Vehicular Ad-hoc Network Environment» International Journal of Network Security & Its Applications, vol. 5, 2013.
- [19] M. d. l. C. Viviane REDING, « Journal officiel de l'Union européenne » August 5th, 2008. [Online]. Available at: <https://eur-lex.europa.eu/legal-content/FR/TXT/HTML/?uri=CELEX:32008D0671&from=FR>.
- [20] ETSI, «ETSI EN 302 663 V1.3.1 -Intelligent Transport Systems (ITS); ITS-G5 Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band» 2020.
- [21] ETSI, «Draft ETSI EN 302 663 V1.2.0: "Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5GHz frequency band"» 2012.
- [22] ETSI, «ETSI EN 300 674 (all parts): Electromagnetic compatibility and Radio spectrum Matters (ERM); Road Transport and Traffic Telematics (RTTT); Dedicated Short Range Communication (DSRC)transmission equipment (500 Kbit/s / 250 Kbit/s) operating in the 5,8 GHz».
- [23] ETSI. « ETSI EN 302 636-3 V1.2.1: Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 3: Network Architecture». 2014-12
- [24] Bellache, Thiwiza, Oyunchimeg Shagdar, and Samir Tohme. "DCC-enabled contention based forwarding scheme for VANETs." 2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). IEEE, 2017.
- [25] ETSI TS 102 687 V1.1.1: "Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part", 2011-07
- [26] ETSI TS 102 724 V1.1.1 (2012-10). "Intelligent Transport Systems (ITS); Harmonized Channel Specifications for Intelligent Transport Systems operating in the 5 GHz frequency band", 2012-10.
- [27] Khan, Irfan, and Jérôme Härri. "Integration challenges of facilities-layer DCC for heterogeneous V2X services." 2018 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2018.
- [28] Amador, Oscar, et al. "Experimental evaluation of the ETSI DCC adaptive approach and related algorithms." IEEE Access 8 (2020): 49798-49811.
- [29] Soto, Ignacio, et al. "Strengths and weaknesses of the ETSI adaptive DCC algorithm: A proposal for improvement." IEEE Communications Letters 23.5 (2019): 802-805.
- [30] ETSI TS 102 636-4-2 V1.1.1; "Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 2: Media-dependent functionalities for ITS-G5", 2013-10.
- [31] Wu, Haitao, et al. "IEEE 802.11 e enhanced distributed channel access (EDCA) throughput analysis." 2006 IEEE International Conference on Communications. Vol. 1. IEEE, 2006.



- [32] G. Bansal, J. B. Kenney, and C. E. Rohrs, "LIMERIC: A linear adaptive message rate algorithm for DSRC congestion control," *IEEE Trans. Veh. Technol.*, vol. 62, no. 9, p. 41824197, Nov. 2013.
- [33] ETSI, «ETSI TS 102 687 V1.2.1: "Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part"» ETSI, 2018.
- [34] Ordonez-Lucena, Jose, et al. "Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges." *IEEE Communications Magazine* 55.5 (2017):80-87.
- [35] 3GPP TS 22.261; Technical Specification Group Services and System Aspects; Service requirements for the 5G system; Stage 1 (Release 19), 2023-09
- [36] Wijethilaka, Shalitha, and Madhusanka Liyanage. "Survey on network slicing for Internet of Things realization in 5G networks." *IEEE Communications Surveys & Tutorials* 23.2 (2021): 957-994.
- [37] Campolo, Claudia, Antonella Molinaro, and Vincenzo Sciancalepore. "5G network slicing for V2X communications: Technologies and enablers." *Radio Access Network Slicing and Virtualization for 5G Vertical Industries* (2021): 239-257.
- [38] Salman, Dheyab & Saleh, Hassan & Mishkhal, Israa. (2021). *Wireless Network Slicing: A Survey*.
- [39] Li, Qian, et al. "An end-to-end network slicing framework for 5G wireless communication systems." *arXiv preprint arXiv:1608.00572* (2016).
- [40] Afolabi, I., Taleb, T., Samdanis, K., Ksentini, A. and Flinck, H., 2018. Network slicing and softwarization: A survey on principles, enabling technologies, and solutions. *IEEE Communications Surveys & Tutorials*, 20(3), pp.2429-2453.
- [41] Da Silva, I., Mildh, G., Kaloxylos, A., Spapis, P., Buracchini, E., Trogolo, A., Zimmermann, G. and Bayer, N., 2016, June. Impact of network slicing on 5G Radio Access Networks. In *2016 European conference on networks and communications (EuCNC)* (pp. 153-157). IEEE.
- [42] Campolo, C., dos Reis Fontes, R., Molinaro, A., Esteve Rothenberg, C. and Iera, A., 2018. Slicing on the road: Enabling the automotive vertical through 5G network softwarization. *Sensors*, 18(12), p.4435.
- [43] Campolo, Claudia, et al. "5G network slicing for vehicle-to-everything services." *IEEE Wireless Communications* 24.6 (2017): 38-45.
- [44] Kotulski, Zbigniew, et al. "On end-to-end approach for slice isolation in 5G networks. Fundamental challenges." *2017 Federated conference on computer science and information systems (FedCSIS)*. IEEE, 2017.
- [45] Šeremet, Irena, and Samir Čaušević. "Benefits of using 5G network slicing to implement vehicle-to-everything (V2X) technology." *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*. IEEE, 2019.
- [46] Gudipati, Aditya, et al. "SoftRAN: Software defined radio access network." *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*. 2013.
- [47] Foukas, Xenofon, et al. "FlexRAN: A flexible and programmable platform for software-defined radio access networks." *Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies*. 2016.

- [48] 3GPP TR 28.801, Study on Management and Orchestration of Network Slicing for Next Generation Network, Rel.15, January 2018.
- [49] Abdel Hakeem, Shimaa A., Anar A. Hady, and HyungWon Kim. "5G-V2X: Standardization, architecture, use cases, network-slicing, and edge-computing." *Wireless Networks* 26 (2020): 6015-6041.
- [50] Porambage, Pawani, et al. "Survey on multi-access edge computing for internet of things realization." *IEEE Communications Surveys & Tutorials* 20.4 (2018): 2961-2991.
- [51] LI, Bing, Yunyong ZHANG, and Lei XU. "An MEC and NFV integrated network architecture." *ZTE Communications* 15.2 (2019): 19-25.
- [52] 3GPP TS 23.501, System Architecture for the 5G System, Rel. 15, Sep. 2017.
- [53] Mach, Pavel, and Zdenek Becvar. "Mobile edge computing: A survey on architecture and computation offloading." *IEEE communications surveys & tutorials* 19.3 (2017): 1628-1656.
- [54] Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., ... & Jue, J. P. (2019). All one needs to know about fog computing and related edge computing paradigms: A complete survey. *Journal of Systems Architecture*, 98, 289-330
- [55] L. Hou, M. A. Gregory and S. Li, "A Survey of Multi-Access Edge Computing and Vehicular Networking," in *IEEE Access*, vol. 10, pp. 123436-123451, 2022, DOI: 10.1109/ACCESS.2022.3224032.
- [56] J.-M. P. GAURANG NAIK, "IEEE 802.11bd & 5G NR V2X: Evolution of Radio Access Technologies for V2X Communications," The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24060, USA, vol. vol 7, pp. 70169-70184, June 10, 2019.
- [57] Y. Y. K. R. Y. K. a. W. A. Hanseul Hong, "Communications, An Effective Wide-Bandwidth channel access in Next-generation WLAN-Based V2X," *Applied Science*, Korea, 2019.
- [58] Sun, B., and H. Zhang. "802.11 NGV proposed PAR." *Proc. IEEE NGV Meeting*. 2018.
- [59] I. Jang, D. Lim, J. Choi, J. Kim, E. Park and S. Kim, "20MHz Channel Access in 11bd. IEEE 802.11-19/0366r6," Available online: <https://mentor.ieee.org/802.11/dcn/19/11-19-0366-06-00bd-20mhz-channel-access-in-11-bd.>, 24 September 2019.
- [60] N. F. a. G. F. Waqar Anwar, "Physical Layer Evaluation of V2X Communications Technologies" *Vodafone Chair Mobile Communications Systems*, 2019.
- [61] [G. a. C. B. a. P. J.-M. Naik, «IEEE 802.11 bd & 5G NR V2X: Evolution of radio access technologies for V2X communications» *IEEE Access*, vol. 7, pp. 70169-70184, 2019.
- [62] Car 2 Car Communication Consortium. (2023-02-02 ). IEEE 802.11bd: The V2X Evolution [White paper]. Retrieved from:
- [63] A. Rostami, B. Cheng, G. Bansal, K. Sjoberg, M. Gruteser, and J. B. Kenney, "Stability challenges and enhancements for vehicular channel congestion control approaches," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 29352948, Oct. 2016.
- [64] Shagdar, Oyunchimeg. *Evaluation of Distributed Congestion Control-Reactive DCC*. Diss. Inria, 2014.
- [65] Kuk, Seungho, and Hyogon Kim. "Preventing unfairness in the ETSI distributed congestion control." *IEEE communications letters* 18.7 (2014): 1222-1225.

- [66] Bansal, Gaurav, et al. "Comparing LIMERIC and DCC approaches for VANET channel congestion control." 2014 IEEE 6th International Symposium on Wireless Vehicular Communications (WiVeC 2014). IEEE, 2014.
- [67] Zemouri, Sofiane, Soufiene Djahel, and John Murphy. "A short-term vehicular density prediction scheme for enhanced beaconing control." 2015 IEEE Global Communications Conference (GLOBECOM). IEEE, 2015.
- [68] Roux, Pierre, et al. "System level analysis for ITS-G5 and LTE-V2X performance comparison." 2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS). IEEE, 2019.
- [69] Karoui, Mouna, Antonio Freitas, and Gerard Chalhoub. "Performance comparison between LTE-V2X and ITS-G5 under realistic urban scenarios." 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring). IEEE, 2020.
- [70] Roux, Pierre, and Valerian Mannoni. "Performance evaluation for co-channel coexistence between ITS-G5 and LTE-V2X." 2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall). IEEE, 2020.
- [71] Available in [Wikipedia](#) (accessed in March 2021).
- [72] Riebl, Raphael, et al. "Artery: Extending veins for VANET applications." 2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS). IEEE, 2015.
- [73] Available at: <https://omnetpp.org/>
- [74] Available at: <https://www.vanetza.org>
- [75] Available at: <https://sumo.dlr.de/docs/index.html>
- [76] Abadi, Martín, et al. "Tensorflow: A system for large-scale machine learning." 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16). 2016.
- [77] Khan, Hamza, et al. "On the application of network slicing for 5G-V2X." European Wireless 2018; 24th European Wireless Conference. VDE, 2018.
- [78] Albonda, Haider D. Resin, and Jordi Pérez-Romero. "An efficient RAN slicing strategy for a heterogeneous network with eMBB and V2X services." IEEE access 7 (2019): 44771-44782.
- [79] Albonda, Haider Daami Resin, and Jordi Pérez Romero. "Analysis of RAN slicing for cellular V2X and mobile broadband services based on reinforcement learning." EAI Endorsed Transactions on Wireless Spectrum 4.13 (2020): 1-11.
- [80] Campolo, C., Molinaro, A., Iera, A. et al. (2018). Towards 5G network slicing for the V2X ecosystem. 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft), IEEE, pp. 400–405.
- [81] Soenen, T., Banerjee, R., Tavernier, W. et al. (2017). Demystifying network slicing: from theory to practice. IFIP/IEEE Symposium on IM.
- [82] Buchholz, Michael, et al. "Enabling automated driving by ICT infrastructure: A reference architecture." arXiv preprint arXiv:2003.05229 (2020).
- [83] Campolo, C., Fontes, R., Molinaro, A. et al. (2018). Slicing on the road: enabling the automotive vertical through 5G network softwarization. Sensors 18 (12): 4435.
- [84] Sanchez-Iborra, R., Santa, J., Gallego-Madrid, J. et al. (2019). Empowering the internet of vehicles with multi-RAT 5G network slicing. Sensors 19 (14): 3107.

- [85] Chekired, D.A., Togou, M.A., Khoukhi, L., and Ksentini, A. (2019). 5G-slicing-enabled scalable SDN core network: toward an ultra-low latency of autonomous driving service. *IEEE Journal on Selected Areas in Communications* 37 (8): 1769–1782. 19
- [86] Albonda, H.D.R. and Pérez-Romero, J. (2019). An efficient RAN slicing strategy for a heterogeneous network with eMBB and V2X services. *IEEE Access* 7: 44771–44782.
- [87] Khan, H., Luoto, P., Samarakoon, S. et al. (2019). Network slicing for vehicular communication. *Transactions on Emerging Telecommunications Technologies*: 1–14
- [88] Elayoubi, S., Jemaa, S., Altman, Z., and Galindo-Serrano, A. 2019. 5G RAN slicing for verticals: Enablers and challenges. *IEEE Communications Magazine*, 57(1), p.28–34.
- [89] Richart, M., Baliosian, J., Serrati, J., Gorricho, J.L., Agüero, R. and Agoulmine, N., 2017, November. Resource allocation for network slicing in WiFi access points. In 2017 13th International conference on network and service management (CNSM) (pp. 1-4). IEEE.
- [90] Høiland-Jørgensen, T., Kazior, M., Täht, D., Hurtig, P. and Brunstrom, A., 2017. Ending the anomaly: Achieving low latency and airtime fairness in Wi-Fi. In 2017 USENIX Annual Technical Conference (USENIX ATC '17) (pp. 139-151).
- [91] Libório, Pedro Paulo, et al. "Network Slicing in IEEE 802.11 ah." 2019 IEEE 18th International Symposium on Network Computing and Applications (NCA). IEEE, 2019.
- [92] Monteuis, J.P., Hammi, B., Salles, E., Labiod, H., Blancher, R., Abalea, E. and Lonc, B., 2017, July. Securing PKI requests for c-its systems. In 2017 26th International Conference on Computer Communication and Networks (ICCCN) (pp. 1-8). IEEE.
- [93] Haidar, F., 2020. Validation platform for vehicle secure and highly trusted communications in the context of the cooperative ITS systems. (Doctoral dissertation, Institut Polytechnique de Paris).
- [94] Gary Lee 2014. Chapter 3 - Switch Fabric Technology. In *Cloud Networking* (pp.37-64).
- [95] Bhamare, Deval, et al. "A survey on service function chaining." *Journal of Network and Computer Applications* 75 (2016): 138-155.
- [96] Retrieved from: <https://osm.etsi.org/>
- [97] Linux manual page – tc(8): <https://man7.org/linux/man-pages/man8/tc.8>.
- [98] Lu, H., Liu, Q., Tian, D., Li, Y., Kim, H., & Serikawa, S. (2019). The cognitive internet of vehicles for autonomous driving. *IEEE Network*, 33(3), 65-73.
- [99] Coutinho, R. W., & Boukerche, A. (2019). Guidelines for the design of vehicular cloud infrastructures for connected autonomous vehicles. *IEEE Wireless Communications*, 26(4), 6-11.
- [100] Lin, Y., Wang, P., & Ma, M. (2017, May). Intelligent transportation system (ITS): Concept, challenge and opportunity. In 2017 IEEE 3rd international conference on big data security on cloud (bigdata security), IEEE international conference on high performance and smart computing (HPSC), and IEEE international conference on intelligent data and security (ids) (pp. 167-172). IEEE.
- [101] Wang, S., Charissis, V., Campbell, J., Chan, W., Moore, D., & Harrison, D. (2016, November). An investigation into the use of virtual reality technology for passenger infotainment in a vehicular environment. In 2016 International Conference on Advanced Materials for Science and Engineering (ICAMSE) (pp. 404-407). IEEE.

- [102] Khan, H., Samarakoon, S., & Bennis, M. (2020). Enhancing video streaming in vehicular networks via resource slicing. *IEEE Transactions on Vehicular Technology*, 69(4), 3513-3522.
- [103] Zhang, J., & Letaief, K. B. (2019). Mobile edge intelligence and computing for the internet of vehicles. *Proceedings of the IEEE*, 108(2), 246-261.
- [104] Zhang, K., Zhu, Y., Leng, S., He, Y., Maharjan, S., & Zhang, Y. (2019). Deep learning empowered task offloading for mobile edge computing in urban informatics. *IEEE Internet of Things Journal*, 6(5), 7635-7647.
- [105] Khamari, S., Ahmed, T., & Mosbah, M. (2022, December). Efficient Edge Server Placement under Latency and Load Balancing Constraints for Vehicular Networks. In *GLOBECOM 2022-2022 IEEE Global Communications Conference* (pp. 4437-4442). IEEE.
- [106] Lopes, D., & Sargento, S. (2014, June). Network mobility for vehicular networks. In *2014 IEEE Symposium on Computers and Communications (ISCC)* (pp. 1-7). IEEE.
- [107] Aljeri, N., & Boukerche, A. (2020). Fog-enabled vehicular networks: A new challenge for mobility management. *Internet Technology Letters*, 3(6), e141.
- [108] Zhang, X., Zhang, J., Liu, Z., Cui, Q., Tao, X., & Wang, S. (2020). MDP-based task offloading for vehicular edge computing under certain and uncertain transition probabilities. *IEEE Transactions on Vehicular Technology*, 69(3), 3296-3309.
- [109] Costa, J. B., de Souza, A. M., Rosário, D., Cerqueira, E., and Villas, L. A. (2019). Efficient data dissemination protocol based on complex networks' metrics for urban vehicular networks. *Journal of Internet Services and Applications*, 10(1):15.
- [110] Wang, S., Urgaonkar, R., Zafer, M., He, T., Chan, K., & Leung, K. K. (2019). Dynamic service migration in mobile edge computing based on Markov decision process. *IEEE/ACM Transactions on Networking*, 27(3), 1272-1288.
- [111] Rui, L., Zhang, M., Gao, Z., Qiu, X., Wang, Z., & Xiong, A. (2021). Service migration in multi-access edge computing: A joint state adaptation and reinforcement learning mechanism. *Journal of Network and Computer Applications*, 183, 103058.
- [112] Gao, Z., Jiao, Q., Xiao, K., Wang, Q., Mo, Z., & Yang, Y. (2019, April). Deep reinforcement learning based service migration strategy for edge computing. In *2019 IEEE international conference on service-oriented system engineering (SOSE)* (pp. 116-1165). IEEE.
- [113] Bao, W., Yuan, D., Yang, Z., Wang, S., Li, W., Zhou, B. B., & Zomaya, A. Y. (2017). Follow me fog: Toward seamless handover timing schemes in a fog computing environment. *IEEE Communications Magazine*, 55(11), 72-78.
- [114] Ksentini, A., Taleb, T., & Chen, M. (2014, June). A Markov decision process-based service migration procedure for follow me cloud. In *2014 IEEE International Conference on Communications (ICC)* (pp. 1350-1354). IEEE.
- [115] Peng, Y., Liu, L., Zhou, Y., Shi, J., & Li, J. (2019, December). Deep reinforcement learning-based dynamic service migration in vehicular networks. In *2019 IEEE Global communications conference (GLOBECOM)* (pp. 1-6). IEEE.
- [116] Abouaomar, A., Mlika, Z., Filali, A., Cherkaoui, S., & Kobbane, A. (2021, October). A deep reinforcement learning approach for service migration in MEC-enabled vehicular

- networks. In 2021 IEEE 46th conference on local computer networks (LCN) (pp. 273-280). IEEE.
- [117] Khamari, S., Ahmed, T., & Mosbah, M. (2022, December). Efficient Edge Server Placement under Latency and Load Balancing Constraints for Vehicular Networks. In GLOBECOM 2022-2022 IEEE Global Communications Conference (pp. 4437-4442). IEEE.
- [118] Maaloul, S., Aniss, H., Kassab, M., & Berbineau, M. (2021). Classification of C-ITS services in vehicular environments. *IEEE Access*, 9, 117868-117879
- [119] Van Hasselt, H., Guez, A., & Silver, D. (2016, March). Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 30, No. 1)