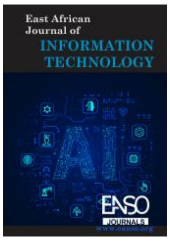


East African Journal of Information Technology
eajit.eanso.org
Volume 8, Issue 1, 2025
Print ISSN: 2707-5346 | Online ISSN: 2707-5354
Title DOI: https://doi.org/10.37284/2707-5354

EAST AFRICAN NATURE & SCIENCE ORGANIZATION

Original Article

Evaluation of Multi-Agent Deep Reinforcement Learning Model with Fault-tolerance Attention Mechanism for Traffic Light Control System

James Adunya Omina[1], Prof. Peter Wagacha Waiganjo, PhD[1], Dr. Lawrence Muchemi, PhD[1] & Dr. Nicodemus Aketch Ishmael, PhD[2]*

[1] University of Nairobi, P. O. Box 1166-00515, Nairobi, Kenya.
[2] Zetech University, P. O. Box 6372-00200, Nairobi, Kenya.
* Author's ORCID ID; https://orcid.org/0009-0004-4468-9310; Email:ishmaelna@gmail.com

Article DOI: https://doi.org/10.37284/eajit.8.1.3028

Date Published:

21 May 2025

Keywords:

Multi-agent Reinforcement Learning, Fault Tolerance, Attention, Prioritized Experience Replay.

ABSTRACT

Managing urban traffic at intersections is a complex challenge. Traditional traffic signal systems struggle to adapt to real-time congestion and variable vehicle flow, particularly at roads with high traffic volume. These systems also lack coordination between neighbouring intersections, leading to inefficient vehicle movement, delays for emergency vehicles, and unsafe pedestrian crossings. This paper proposes a solution using Multi-Agent Reinforcement Learning (MARL) to model a traffic network as a multi-agent system. Specifically, it employs Fault-Tolerant Attention Multi-Agent Deep Deterministic Policy Gradient (FT Attn. MADDPG), where decisions are based on average queue lengths. The Fault-tolerance Attention mechanism allows agents to minimize the impact of malfunctioning agents, improving overall performance. The approach also supports various intersection types through a parametric action space. Simulation results show that FT Attn. MADDPG significantly reduces travel time by 16.21% under high, 26.97% under medium, and 6.89% under low traffic demand compared to standard MADDPG.

APA CITATION

Omina, J. A., Waiganjo, P. W., Muchemi, L. & Ishmael, N. A. (2025). Evaluation of Multi-Agent Deep Reinforcement Learning Model with Fault-tolerance Attention Mechanism for Traffic Light Control System. *East African Journal of Information Technology*, *8*(1), 121-145. https://doi.org/10.37284/eajit.8.1.3028

CHICAGO CITATION

Omina, James Adunya, Peter Wagacha Waiganjo, Lawrence Muchemi and Nicodemus Aketch Ishmael. 2025. "Evaluation of Multi-Agent Deep Reinforcement Learning Model with Fault-tolerance Attention Mechanism for Traffic Light Control System". *East African Journal of Information Technology* 8 (1), 121-145. https://doi.org/10.37284/eajit.8.1.3028.

HARVARD CITATION

Omina, J. A., Waiganjo, P. W., Muchemi, L. & Ishmael, N. A. (2025) "Evaluation of Multi-Agent Deep Reinforcement Learning Model with Fault-tolerance Attention Mechanism for Traffic Light Control System", *East African Journal of Information Technology*, 8(1), pp. 121-145. doi: 10.37284/eajit.8.1.3028

IEEE CITATION

J. A., Omina, P. W., Waiganjo, L., Muchemi & N. A., Ishmael "Evaluation of Multi-Agent Deep Reinforcement Learning Model with Fault-tolerance Attention Mechanism for Traffic Light Control System", *EAJIT*, vol. 8, no. 1, pp. 121-145, May. 2025.

MLA CITATION

Omina, James Adunya, Peter Wagacha Waiganjo, Lawrence Muchemi & Nicodemus Aketch Ishmael "Evaluation of Multi-Agent Deep Reinforcement Learning Model with Fault-tolerance Attention Mechanism for Traffic Light Control System". *East African Journal of Information Technology*, Vol. 8, no. 1, May. 2025, pp. 121-145, doi:10.37284/eajit.8.1.3028

INTRODUCTION

In many urban areas, where traffic congestion does not show a peak pattern, conventional traffic signal timing tactics do not yield an effective control (Abdoos et al., 2011). A centralised agent cannot be trained for a wide range of traffic signal control, despite the fact that deep neural networks have improved the RL's scalability (Hu & Li, 2024). High failure rates and latency, as well as the loss of traffic network data, are the practical outcomes of this centralised state processing (Hu & Li, 2024). Because each local agent receives a portion of the global control, DRL with many agents provides an efficient way to control traffic signals. The authors Van Der Pol and Oliehoek (2016) adapted a single-agent solution to the settings of many actors in order to accomplish coordination among several crossings (Van Der Pol & Oliehoek, 2016). DRL techniques with numerous agents have been used more and more recently for traffic signal management, and each large study has produced positive research findings. In order to operate signal controls for a tiny traffic grid, (Wiering, 2000) included Q-learning into a multi-agent model. One potential technique for multi-agent DRL is the use of a Deep Q-Network in a coordination algorithm (Van Der Pol & Oliehoek, 2016). A Double Deep Q-Network was used by Gu et al. (2020) to effectively minimise traffic at a four-phase signalised intersection. However, its use for large-scale signal control was hindered by the curse of dimensionality (Hu & Li, 2024).

According to Hu and Li (2024), there has been a growing interest in multi-intersection modelling difficulties. Cooperation between the agents is essential in the traffic light management problem since each agent's activities have a direct effect on the others. A lot of traffic data is gathered in real time, and agents should be able to connect and exchange it in the right way. Effective collaboration mechanisms are necessary to increase performance at intersections since inter-agent interference can result in traffic chaos (Hu & Li, 2024).

According to (Chu et al., 2020), MA2C is a fully cooperative system where each junction teaches a separate agent to share observations with nearby agents (Chu et al., 2020). To make better decisions regarding traffic flow, local agents have access to information about local traffic rather than only their intersections (Hu & Li, 2024). The topic of large-scale traffic signal control using multi-agent deep reinforcement learning has been progressed by a significant amount of research, although the majority of these works have focused on the complex situation of mastering arterial traffic signal management (Hu & Li, 2024). The problem of arterial traffic control, which has a wide state-action space, makes it difficult to navigate the solution space and effectively extract relevant information. The MASAC model, as forth by Mao et al. (2023), strengthens traffic information extraction by including an attention mechanism into actor and critic networks. By enabling the model to flexibly focus on particular input data segments at each stage of its operation, the attention mechanism-a crucial deep learning technique-improves the model's performance when processing sequence data. By excluding the irrelevant, this method guarantees that the model concentrates on the most important data (Hu & Li, 2024).

The effectiveness and scalability of DRL algorithms are demonstrated by the several neural network-based DRL algorithms that are used to train agents and integrate multiple agents for successful traffic signal control (Hu & Li, 2024). This project builds numerous agents for traffic signal coordination using the Multi-Agent Deep Deterministic policy gradient (MADDPG) algorithm. For trials, we take into account various traffic situations, such as low, normal, crowded, and unbalanced.

This is how the remainder of the paper is structured. Section II discusses reinforcement learning, with a focus on MADDPG. Section III presents several related studies that have applied reinforcement learning to traffic signal control. Section IV presents the proposed MADDPG technique. In Section V, the network configuration is displayed, and in Section VI, the experimental results. Finally, Section VII concludes the work.

RELATED WORK

Wei et al. (2023) claim that the conventional reinforcement learning method performs admirably when tackling challenges with a limited sample space (Wei et al., 2023). It is ineffective, nevertheless, when it comes to solving the problem of enlarging the state and action spaces. Deep reinforcement learning has progressively made its way into academic and industrial domains since Silver used AlphaGo (Silver et al., 2016) to defeat the world chess champion. Silver has achieved remarkable accomplishments in a number of disciplines. Despite the many challenges that multi-agent learning faces, deep reinforcement learning offers a great way forward (Wei et al., 2023). Kolat et al. (2023) addressed the traffic signal control problem using a multi-agent deep Q-learning algorithm (Kolat et al., 2023). They introduced a novel reward mechanism tailored for multi-agent environments, aiming to enhance both sustainability and traditional traffic efficiency metrics. Their approach led to notable improvements, including an 11% reduction in fuel consumption and a 13% decrease in average travel time. The study highlights the potential of reinforcement learning to optimize traffic light coordination and mitigate urban traffic congestion, contributing to more sustainable and efficient transportation systems. Hu and Li (2024) employed a Double Deep Q-Network (DDQN) approach within a Deep Reinforcement Learning (DRL) framework to train local agents independently, enabling them to adapt to regional traffic patterns. After training, a global agent was introduced to coordinate the policies of these local agents for synchronized traffic signal control. Using the Simulation of Urban Mobility (SUMO), they demonstrated that their multi-agent model effectively enhanced intersection efficiency and significantly reduced average vehicle waiting times and queue lengths, outperforming traditional methods like PASSER-V and pre-timed signal controls (Li, Yu, et al., 2021).

According to Li, Xua, et al. (2021), traditional traffic signal control methods can lead to serious problems like traffic congestion and wasted energy (Li, Xua, et al., 2021). To address these issues, the authors highlight reinforcement learning (RL) as a modern, data-driven approach that adapts traffic signals in real time, making it well-suited for managing the complexities of urban traffic networks. Applying deep reinforcement learning (RL) to transportation networks with multiple signalised intersections still presents certain difficulties, despite the fact that the development of deep neural networks (DNN) (Li, Yu, et al., 2021) further improves its learning capability. These difficulties include non-stationarity environments, exploration exploitation dilemmas, multi-agent training schemes, continuous action spaces, etc. The authors (Li, Yu, et al., 2021) claim that MADDPG features a decentralised execution and centralised learning paradigm where actors act based on their own local observations and critics use extra information to expedite the training process. The Simulation of Urban MObility (SUMO) platform was used to simulate the model and assess its performance. The effectiveness of the suggested algorithm in managing traffic lights was demonstrated by the model comparison findings. To enhance traffic signal coordination, Li, Yu, et al. presented KSDDPG (Knowledge Sharing Deep Deterministic Policy Gradient), a multi-agent reinforcement learning technique, in their 2021 work (Li, Yu, et al., 2021). The system enables each agent to comprehend the larger traffic environment by facilitating knowledge sharing among agents via a communication protocol. In terms of efficiency and traffic fluctuation adaptability, KSDDPG performed better than current RL-based and conventional traffic control techniques when tested on both synthetic and real-world datasets. Furthermore, without incurring additional computational costs, the knowledge-sharing approach enhanced model convergence.

Wei et al. (2023) addressed the challenge of scalability in multi-agent systems under environmental uncertainty by proposing a cooperative model based on a graph attention network (Wei et al., 2023). Their approach combined graph convolution to model agent relationships and recurrent neural networks to manage continuous action spaces. By encoding interaction weights among agents and action dependencies, the model enhances coordination and decision-making. Evaluated through simulations in a 3D wargame with UAVs and radar stations, the model demonstrated superior scalability, robustness, and learning efficiency compared to existing methods. For urban traffic control, Azad-Manjiri et al. (2025) presented DDPGAT, a novel framework that combined Graph Attention Networks (GATs) and Multi-Agent Deep Deterministic Policy Gradients (MADDPG). In DDPGAT, traffic signal controllers use GATs to dynamically determine the value of each route in their role as independent agents. One important component is a moral reward system that incentivises choices that improve nearby intersections and advance moral traffic control. Agents are better able to recognise local and international traffic patterns when they receive shared attention during training. According to experiments, DDPGAT greatly enhances traffic flow and lessens congestion, proving its efficacy and signalling a significant advancement in intelligent traffic systems (Azad-Manjiri et al., 2025). Gu et al. (2021a) explored the security challenges in multi-agent reinforcement learning (MARL) systems, particularly when some agents behave in arbitrarily faulty or malicious ways due to harsh environments (Gu et al., 2021a). Traditional methods assumed prior knowledge of environmental noise intensity, limiting their adaptability. To address this, the authors proposed FT-Attn., an attention-based fault-tolerant model that dynamically selects relevant and accurate information for each agent without relying on prior noise knowledge. Using a multi-head attention mechanism, FT-Attn. enables agents to learn both communication and action policies effectively. Empirical results showed that FT-Attn. outperforms previous approaches in highly noisy cooperative and competitive settings, achieving performance close to optimal.

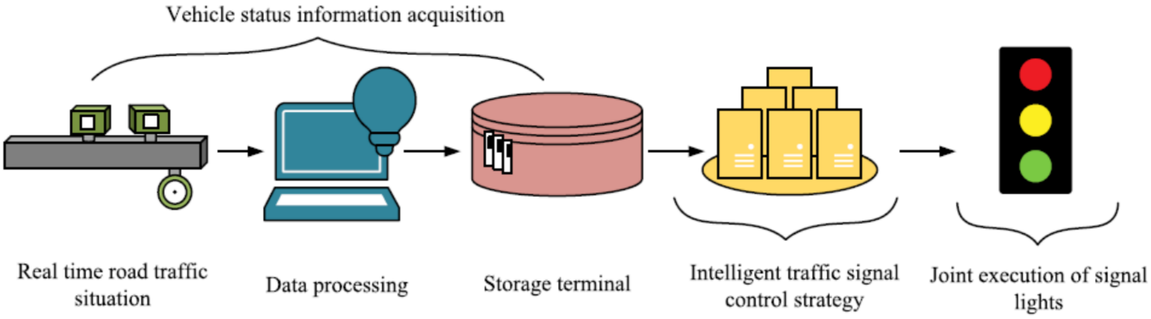
FT-Attn. MADDPG Model Framework

This paper developed an intelligent signal control system using Fault-tolerant Attention multi-agent DDPG (FT-Attn. MADDPG) and evaluated it on a simulated corridor with real-world traffic volumes, motions, and network topology, including intersection spacing. Each agent in the developed FT-Attn. MADDPG is constructed with a centralized critic that estimates the agent's value function based on global observations and an actor network that chooses autonomous actions conditioned on local observations. Every agent is designed to enable the selection and application of up to eight signal phases, which are frequently used in field controllers. The performance of the developed FT-Attn. MADDPG would be evaluated against the fixed time-coordinated signal timings that are currently in use in the field and are modelled using SUMO software in the loop simulation (SILs) for the test corridor and field recorded traffic loads. Sensitivity experiments are conducted using volumes that are (a) modified upward by 5% and (b) adjusted lower by 10% from the field measured volumes in order to assess the robustness of the developed method (Kwesiga et al., n.d.).

System Architecture

Figure 1 presents the proposed system architecture design of a traffic control system. The vehicle status information acquisition may send a report of the traffic incidents and send a notification of the traffic situation to the nearby control unit. The various details provided in the proposed model would be useful in providing a database of traffic and the Geographic Information System (GIS). It would also limit the time needed to report an accident and more accurately determine its location.

Figure 1: Structure of Traffic Light Control System


Vehicle status information acquisition
Real time road traffic situation
Data processing
Storage terminal
Intelligent traffic signal control strategy
Joint execution of signal lights

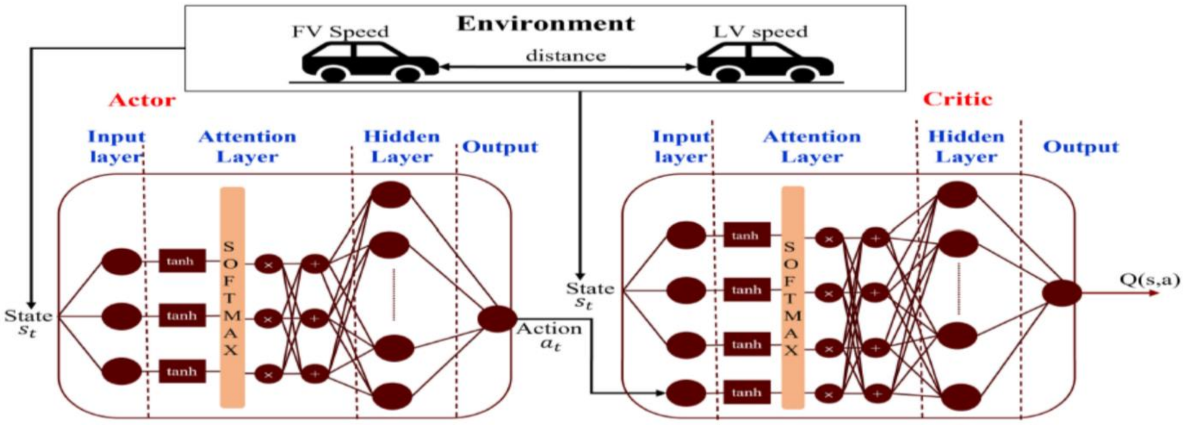
Source: *(Jin, 2024)*

The three main components of the FT-Attn. MADDPG model are depicted in Figure 1 as its basic framework (Fan et al., 2025). Data collection from traffic signals is the first component. Local observation data and location information at junctions are gathered using a variety of sensors at certain times (explained below) and converted into vectors. Using a multilayer perceptron (MLP) and gated recurrent unit (GRU) for dynamic updates, the second component concentrates on knowledge acquisition and updating. The third part is SAC, the main decision-making algorithm, which uses observational data and learnt information to provide optimal traffic signal control algorithms. The model ultimately generates traffic signal action decisions, facilitating intelligent regulation and enhancing the effectiveness of traffic flow.

B. FT Attn. MADDPG Model

Figure 2 depicts the Intelligent Traffic Light Controller. Vehicles are found using infrared sensors. This serves as the traffic light control (TLC) unit's input. Red, Green, and Orange output signals are produced by the Intelligent traffic light control (ITLC) unit. This traffic controller's fundamental functions are carried out by an embedded device. The system is supposed to change the cycle time depending upon the densities of cars behind green and red lights and the current cycle time. In a conventional traffic light controller, the lights change at constant cycle time, which is clearly not optimal. It would be more feasible to pass more cars at the green interval if there are fewer cars waiting behind the red lights. Obviously, a mathematical model for this decision is enormously difficult to find. However, with deep reinforcement learning (DRL), it is relatively much easier. In DRL traffic Signal control; the length of the green time is varied in accordance to the local traffic situation. The DRL controller is used to determine the duration of the green phase. The flow diagram is as shown in Figure 3:

Figure 2: The Structure of the Proposed DDPG Model



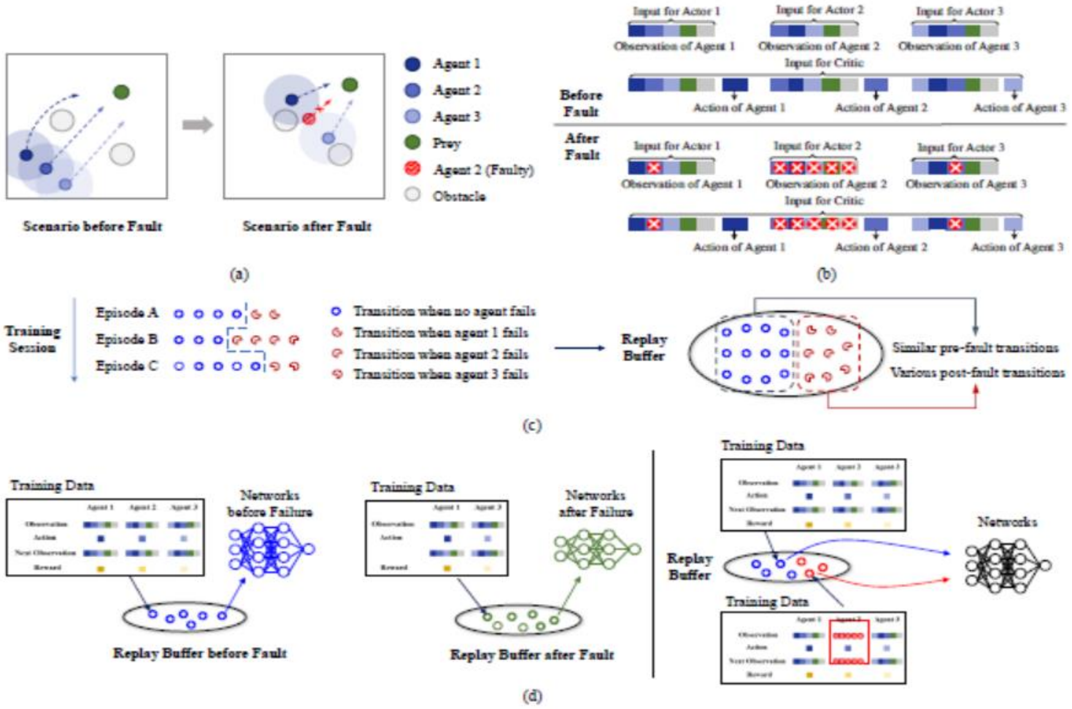
Source: *Adopted from: (Islam et al., 2024)*

The inputs to the DRL traffic light Controller are the local traffic situation variables. For each traffic light, there is incoming traffic and outgoing traffic. These volumes of traffic in the number of vehicles per min (Veh/min) represent the input variables. The output is the duration of green-time for that particular traffic light.

Fault-Tolerant Model with Attention on Actor and Critic

We introduce an attention-guided fault-tolerant method for MARL as in (Shi et al., 2024) to tackle the chaotic state space, named Fault-Tolerant Model with Attention on Actor and Critic, (AACFT) for short. Before introducing AACFT, there are two natural remedial ideas to deal with faults. The first idea is to manually distinguish the training data and the networks before and after the agent fault (Fig 3. (d)Right). In such a case, experiences before and after faults are stored in different replay buffers, and actors and critics are separately set and trained. The drawback of this method is that it requires multiple replay buffers and actor-critic networks, which can be quite cumbersome. The second idea is to identify the invalid information within the input automatically by the neural network. In such a case, experiences before and after faults are stored in the only replay buffer, and a unique actor-Critic network distinguishes whether a fault has occurred by the training data. Nevertheless, the presence of invalid information could lead to the learning of a suboptimal policy. Unlike the above two ideas, our proposed AACFT is capable of automatically identifying unexpected faults while appropriately tackling the special information within the chaotic state space. Specifically, we have carefully devised a method for configuring the input of the critic and actor networks and have integrated an attention module into the networks, building upon the MADDPG framework. In the critic, the observation of the faulty agent is no longer meaningful, and the attention module can

Figure 3: (a) An Illustration of a Predator-prey System bBefore and After the Agent Fault. (b) An Illustration of the Inputs for the Actor and Critic Before and After Fault. (c) An Illustration of Two Natural Ideas of Handling Faults. (d) An Illustration of a Replay Buffer with Transitions in 3 Episodes



Source: *(Shi et al., 2024)*

then prioritize shifting attention away from the observations of the faulty agent and focus on other relevant information within the input. Within the actor, the observation of each agent encompasses the states of other agents, enabling the recording of the state when a fault occurs. The attention module can then dynamically modulate the level of attention assigned to the fault information based on its impact on the system.

C. Agent Design

In the following sections, the action space and reward function setup, as well as the design of the state representation for the traffic environment, are discussed in detail.

Observation of intersections (O): The traffic environment variables that an intelligent agent measures at an intersection are derived from its local state observation.

Action (A): According to this study, agent I's action at a specific junction j defines the timing of the current phase. It can be converted into signal phase lengths by using Equation 1.

$$t = 0.5.(1+a).(t_{\max} - t_{\min}) + t_{\min}$$

(1)

Where t is the phase timing duration and is the actual action applied to the environment. The maximum and minimum green light durations for the intersection's signal phases are represented by $t_{max}$and t_{min}, respectively.

Reward (R): The reward, which accounts for the fair allocation of transport resources in the study, is the negative sum of the intersection queue lengths and the Sum of Absolute Deviations (SADs) of vehicle numbers across each entrance lane. A shorter queue duration indicates less congestion, and a smaller total deviation indicates a more equitable distribution of vehicles between lanes. The prize is given at the conclusion of each phase. A neighboring agent incentive has been incorporated to promote cooperation among agents. This incentive is calculated for each individual agent using Equations 2 and 3.

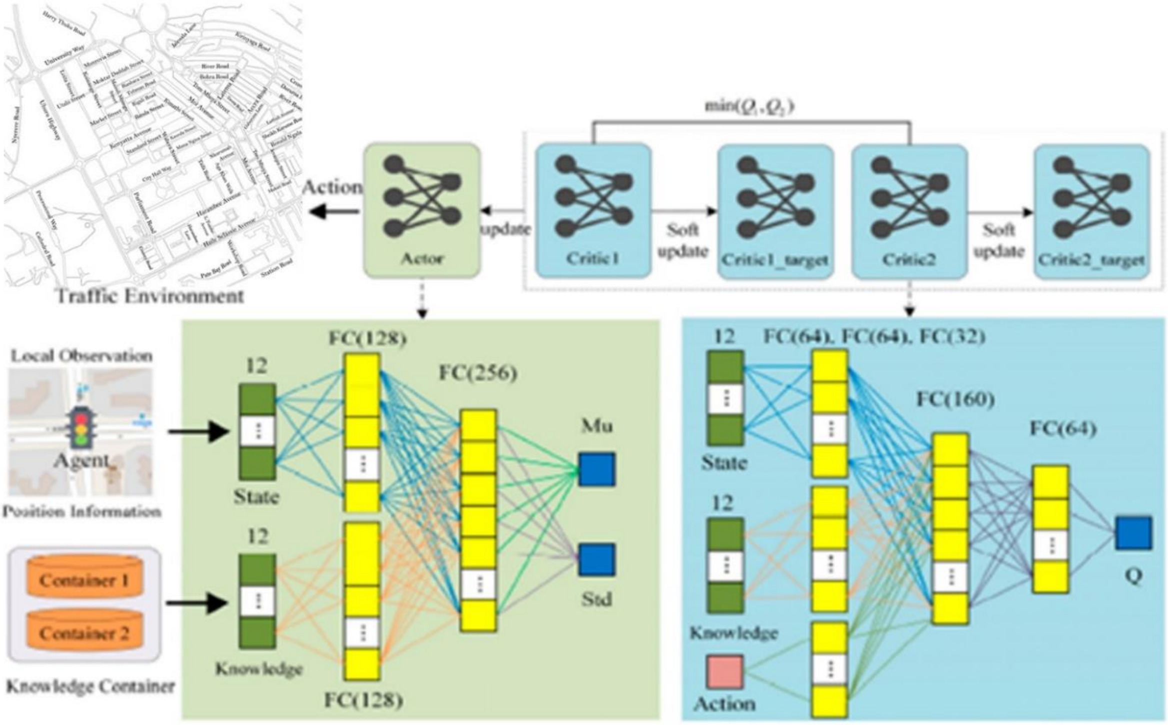
$$r_L = -\left(\sum_{i=1}^{n} |x_i - \overline{x}| + \sum_{i=1}^{n} x_i \right) \tag{2}$$

$$r = c_1.r_L + c_2.\overline{r}_N \tag{3}$$

Where x_i is the number of cars in the intersection's i^{th} entry lane. The local reward is denoted by the phrase r_L , whilst the average value of the nearby rewards is shown by the term $\overline{r}_N$. As weighting factors, the constants c_1 and c_2 have values of 0.6 and 0.4, respectively.

Figure 4 shows how the agent's value and policy networks are designed. To parameterize a Gaussian distribution, the actor network produces the mean (Mu) and standard deviation (s.d). Actions are sampled from this distribution to introduce stochasticity and promote exploration.

Figure 4: The Agent's Particular Network Architecture



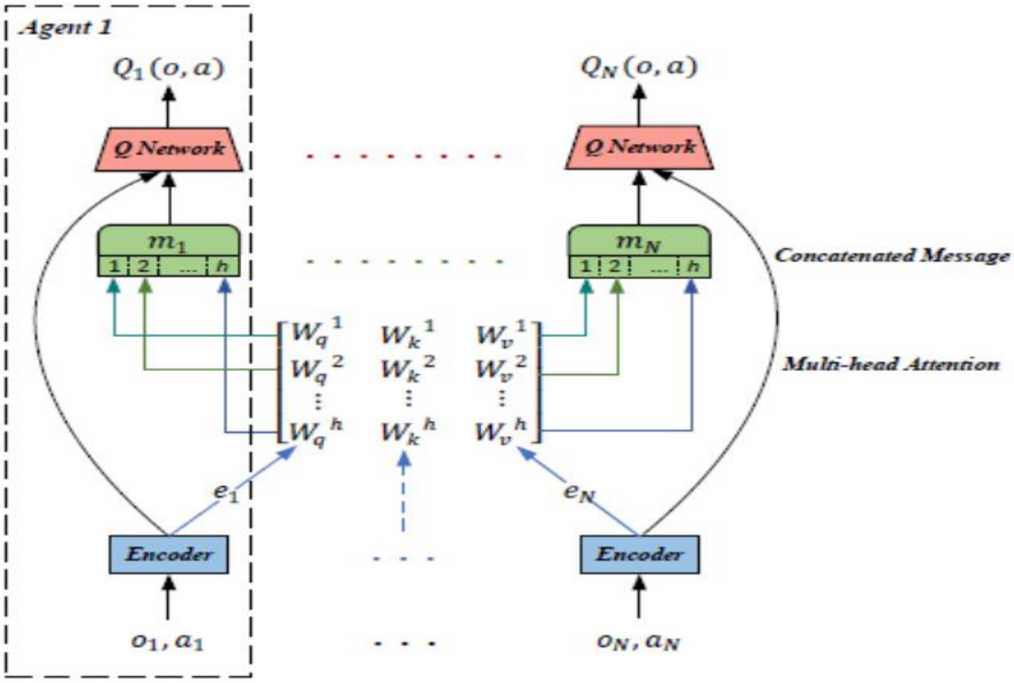
Source: *Adopted from: (Fan et al., 2025)*

We present a multi-head attention mechanism that selectively attends to the opinions of other agents in order to learn the critic for each actor. The key elements of our methodology are illustrated in Figure 5.

To identify agent interactions and choose relevant observations, we employ multi-head dot-product attention. Each agent naturally enquires about the observations and behaviors of other agents in order to assess its value function, after which it takes into account the relevant information. The value is estimated as shown in Equation 4, taking into account the contributions of other agents as well as the observation and activity of agent i denoted as.

$$Q_i^{\psi}\left(o,a\right)=f_i\left(g_i\left(o_i,a_i\right)m_i\right)$$
(4)

Figure 5: Framework of FT-Attn. MADDPG Algorithm



Source: *(Gu et al., 2021b)*

In this case, g_i denotes the encoder function and f_i the Q-Network. The weighted sum of each agent's value represents the contribution from other agents as shown in Equation 5.

$$m_i = \sigma\left(Concat\left[\sum_{j\in\backslash i} \alpha_{ij}^h W_v^h e_j, \forall h \in \mathrm{H} \right] \right)$$

(5)

Where e_j is the embedding represented by the g_j function and h is an attention head. e_j is converted into a "value" by W_v^h. \i is the representation of the set of all agents other than i, and j is the index. Each independent attention head projects each agent's input feature to the query, key, and value representation in order to determine the weight a_{ij}^h. The relationship between i and j for attention head h is calculated as presented in Equation 6.

$$\alpha_{ij}^h = \frac{\exp\left(\tau . W_q^h e_i . \left(W_k^h e_j\right)^T\right)}{\sum_{r\in\backslash i} \exp\left(\tau . W_q^h e_j . \left(W_k^h e_r\right)^T\right)}$$

(6)

Where W_q^h turns e_i into a "key" and τ, a scaling factor, W_k^h turns e_j into a "query."

FT-Attn. MADDPG Algorithm

Following the flow of the FT-Attn. MADDPG algorithm (Algorithm 1), the pseudo-code statements in lines 8 and 16 are designed to identify two discrete time points: the start and finish times of the green light phase.

Algorithm 1: FT-Attn. MADDPG Algorithm for N Interacting Agents

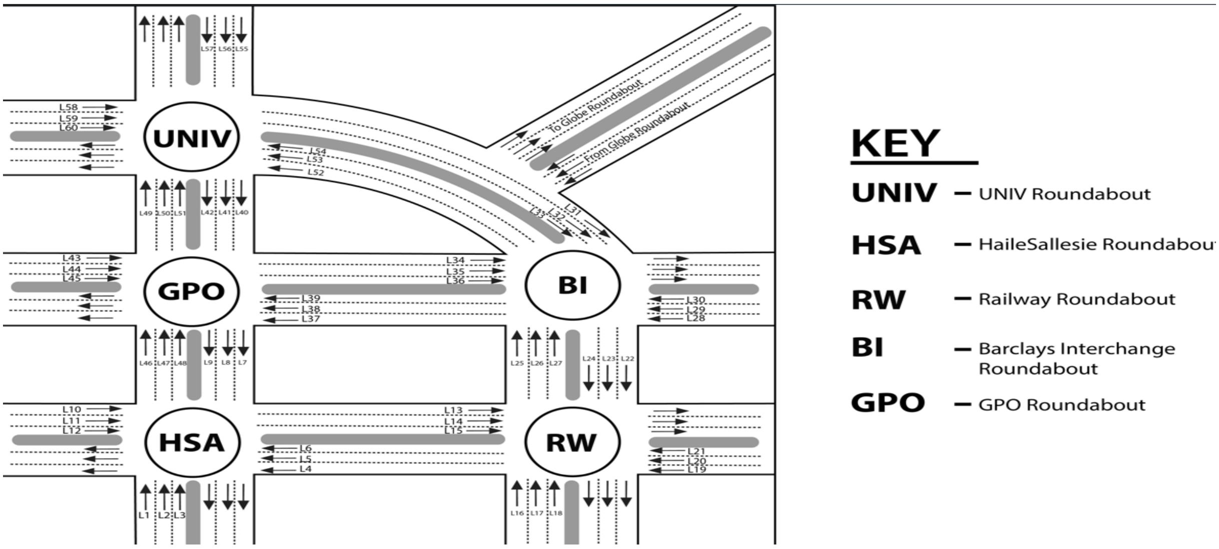
```
Initialize actor networks $(\pi_{\theta_1}, ..., \pi_{\theta_N})$ and twin critic networks $(Q_{\psi_{1,1}}, Q_{\psi_{1,2}}, ..., Q_{\psi_{N,1}}, Q_{\psi_{N,2}})$
Initialize target critic networks $(Q'_{\psi_{1,1}}, Q'_{\psi_{1,2}}, ..., Q'_{\psi_{N,1}}, Q'_{\psi_{N,2}})$ with $\psi'_{i,j} \leftarrow \psi_{i,j}, \forall_i \in \{1, ..., N\}, j \in \{1, 2\}$
Initialize replay buffers $(D_1, ..., D_N)$
for episode = 1 to E do
    Initialize knowledge $k_r$ and $k_c$ , and receive initial state x
    for t = 1 to max episode length do
        for agent i = 1 to N do
            If yellow phase step = 3 then
                Receive observation $o_i$ and position information $p_i$, set x = $(o_i, p_i)$
                Receive the collective knowledge $k_r$, set $k_i$ =$k_r$
                Generate knowledge obtaining vector $u_i$
                Generate knowledge updating vector $\hat{k}_i$
                Store the new knowledge in the container $k_r \leftarrow \hat{k}_i$
                Select action $a_i$ and execute action a = $a_i$
            end if
            if green phase step = a then
                Receive observation$o_i^{\prime}$ and position information $p_i$, set $x' = (o_i', p_i')$
                Receive the collective knowledge $k_c$, set $k_i$ =$k_c$
                Generate knowledge obtaining vector $u_i^{\prime}$ using Equation (10)
                Generate knowledge updating vector $\hat{k}_i$ using Equation (11)
                Store the new knowledge in the container $k_r \leftarrow \hat{k}_i$
                Select action $a_i$ and execute action a = $a_i$
            end if
            Set $u = (u_i, u_i')$
            Store $(x, x', a, u, r)$ in $D_i$
        end for
    end for
for $i_{ep} = 1...num$ episodes do
    Observe initial state $o_i$ for each agent i,
    for t = 1 . . . steps per episode do
        Select actions $a_i \sim \pi_i(.|o_i)$ for each agent i.
        Execute the action $a_i$ and get $o_i'$ , $r_i$ for all agents.
        Store transitions $(o_1...N, a_1...N, r_1...N, o_1'...N)$ in D
        Sample minibatch $B \leftarrow m \times (o_1...N, a_1...N, r_1...N, o_1'...N) \in D$ , and unpack.
        Calculate $Q_i^{\psi}(o_1...N, a_1...N)$ for all i in parallel, $a_i' \sim \pi_i^{\theta}(o_i')$, using target policies, $Q_i^{\bar{\psi}}(o_1'...N, a_1'...N)$
```

```
Set $y_i = r_i + \gamma E_{\alpha' \sim \pi_\theta(o')}\left[Q_i^{\bar{\psi}}\left(o', a'\right)\right]$,
Update critic by minimizing the loss:
$L_Q(\psi) = \sum_{i=1}^{n} E_{(o,a,r,o') \sim D}\left[\left(Q_i^{\psi}(o.a) - y_i\right)^2\right]$,
Update policy:
$\nabla_\theta J(\pi_\theta) = E_{\alpha \sim \pi_\theta}\left[\nabla_{\theta_i} \log\left(\pi_{\theta_i}(a_i \mid o_i)\right)\left(b\left(o, a_{\setminus i}\right) Q_i^{\psi}(o, a)\right)\right]$
Update target critic and policy parameters:
$\bar{\psi} = \tau\bar{\psi} + (1-\tau)\psi$
$\bar{\theta} = \tau\bar{\theta} + (1-\tau)\theta$
        end for
end for
```

D. Experiments

The performance of the proposed FT-Attn. MADDPG (TSC) model was evaluated using a traffic simulation environment built with SUMO (Simulation of Urban MObility) version 1.18.0 (Guastella & Bontempi, 2023). SUMO is an open-source tool developed by the German Aerospace Center which enables detailed, realistic traffic simulations by modeling individual vehicle behavior and integrating real-world data. Its flexibility allows for testing of autonomous vehicle coordination, intelligent transportation systems, and traffic control strategies. It also provides detailed outputs like trip times, emissions, and congestion, making it useful for policy and infrastructure planning. The simulation focused on traffic flow at five nodes in Nairobi's central business district. The traffic flow model for the five nodes in Nairobi's central business district is shown in Figure 6.

Figure: 6: Traffic Flow Model of the Five-nodes


UNIV
GPO
HSA
BI
RW
To Globe Roundabout
From Globe Roundabout
L1 L2 L3 L4 L5 L6 L7 L8 L9 L10 L11 L12 L13 L14 L15 L16 L17 L18 L19 L20 L21 L22 L23 L24 L25 L26 L27 L28 L29 L30
L31 L32 L33 L34 L35 L36 L37 L38 L39 L40 L41 L42 L43 L44 L45 L46 L47 L48 L49 L50 L51 L52 L53 L54 L55 L56 L57 L58 L59 L60
KEY
UNIV – UNIV Roundabout
HSA – HaileSallesie Roundabou
RW – Railway Roundabout
BI – Barclays Interchange Roundabout
GPO – GPO Roundabout

In order to fairly compare the suggested FT-Attn. MADDPG model with other deep reinforcement learning techniques and traditional methods, we construct an intricate and realistic traffic simulation environment using Simulation of urban Mobility (SUMO). Five intersections (i.e., Haile Salesia, Railway, Moi Avenue Interchange, University way and GPO Roundabouts) in the

Nairobi Central Business District of Kenya served as the model for the testing scenarios (see Figure 7). Parameter values were established as shown in Table 1.

Figure 7: Five Regulated Intersections Make Up the Actual Road Network in Nairobi, Kenya's Central Business District.

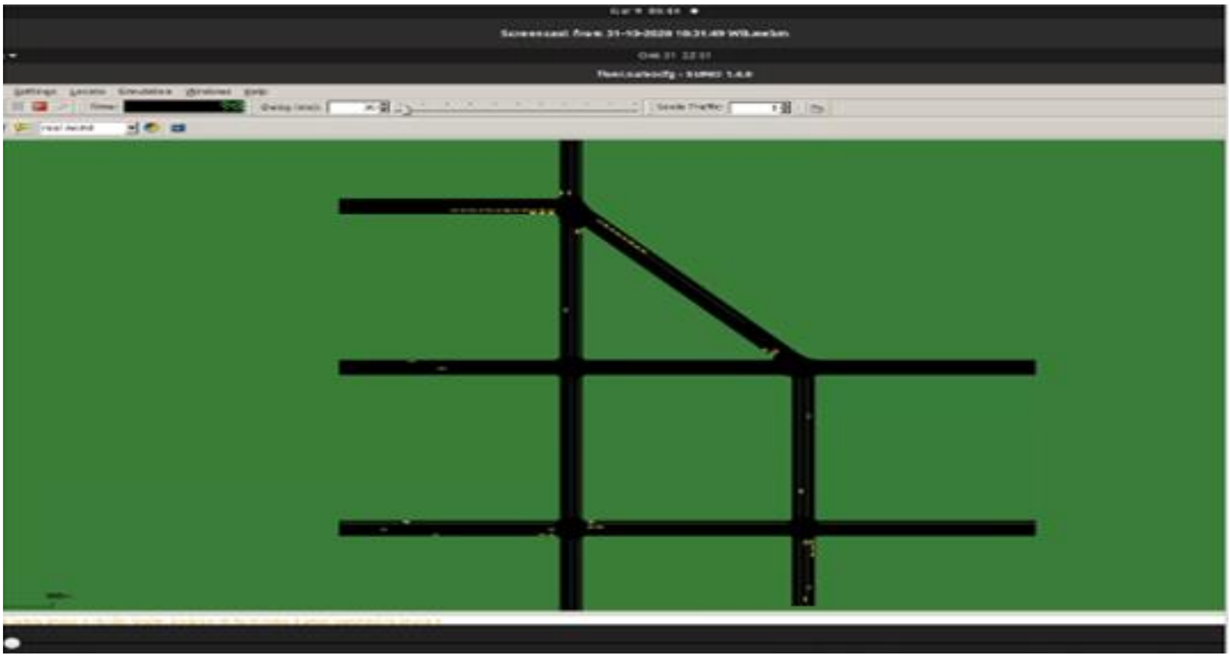


Table1: Settings of Parameters in the FT-Attn. MADDPG (TSC) Model

Hyper Parameter	Value
Episode Time Lengths in seconds	1200
Discount Factor	0.99
Target network parameter update rate	0.01
Initial alpha	0.01
Target entropy	-1.0
Alpha learning rate	0.001
Batch size	128
Size of replay memory	1×10^5
Learning rate of the Actor network	0.001
Learning rate of the critic network	0.002
Minimum green (s)	6
Maximum green (s)	30
Yellow(s)	3

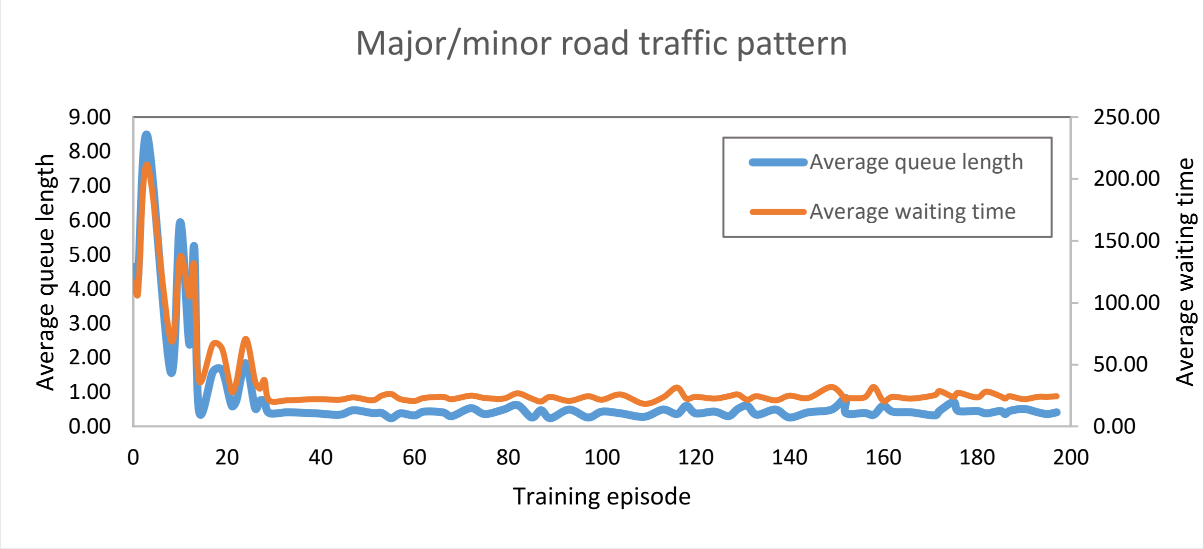
E. Comparison Baseline

This study evaluates and validates the effectiveness of the proposed MADDPG model and its improved version, FT-Attn. MADDPG, by including baseline comparisons between deep reinforcement learning (DRL) control strategies and traditional traffic signal management techniques. Fixed-Time Control (FTC), the industry standard for traffic signal control, is one of the most widely utilized strategies in traffic signal management. FTC employs pre-established signaling schemes to regulate traffic flows and offers advantages in terms of cost-effectiveness and implementation simplicity. All algorithms were trained using 400 episodes, and the simulation had a maximum of 1200 steps.

RESULTS AND ANALYSIS

For several reinforcement learning techniques, Figure 8 displays the average reward patterns during training. Smoothing the curves is done using a moving average method with a window size of 10. From locally optimal solutions to random strategies, these reward curves demonstrate how the models are optimized. In the initial training episodes, the reward curves of all algorithms show a sharp increase, followed by a steady convergence.

Figure 8: Major/minor Road Traffic Pattern



Results and Analysis with Low Traffic Demand

Table 2 illustrates results of average waiting time with low traffic demand across the three algorithms. The average number of vehicles with low traffic demand at intersection of highway was found to be 972. This value was a little lower than that found by Gonzales, et al. (2009). It was 19.65% less than the 1163 found for the North Eastern junction of Uhuru Highway–Kenyatta Avenue intersection at morning peak hours between 7 and 8 am.

Table 2: Average Waiting Time with Low Traffic Demand (seconds)

Algorithm Implementation	Number of vehicles	All junctions	gne0	gne1	gne2	gne3	gne4
Fixed	**972**	420	317	288	350	312	244
MADDPG	**972**	249	274	242	238	232	260
FT Attn. MADDPG	**972**	171	181	178	168	156	172

We must examine measures of central tendency (such as means) and measures of variability (such as standard deviation or confidence intervals) in order to talk about the statistical significance of the data presented. We can examine the data as follows because we have provided fixed numerical findings for seven trials or measurements for three techniques (Fixed, MADDPG, and FT Attn. MADDPG).

1. Raw Data Summary

Method	Mean	Std. Dev
Fixed	321.83	54.27
MADDPG	249.17	14.55
FT Attn. MADDPG	171.00	8.27

2. 95% Confidence Intervals:

Assuming these are sample means from approximately normal distributions (or using the Central Limit Theorem), and sample size n=6, we can calculate the 95% Confidence Interval (CI) as:

$$CI = \bar{x} \pm t_{0.025,n-1} \cdot \frac{s}{\sqrt{n}}$$

Using $t_{0.025,5}$ ≈2.57:

Method	CI
Fixed	[264.82, 378.84]
MADDPG	[233.89, 264.45]
FT Attn. MADDPG	[162.31, 179.69]

3. Interpretation of Statistical Significance

- **No Interval Overlap:**

The lack of overlap between the confidence intervals indicates that, at the 95% level, the differences between Fixed > MADDPG > FT Attn. MADDPG are statistically significant.

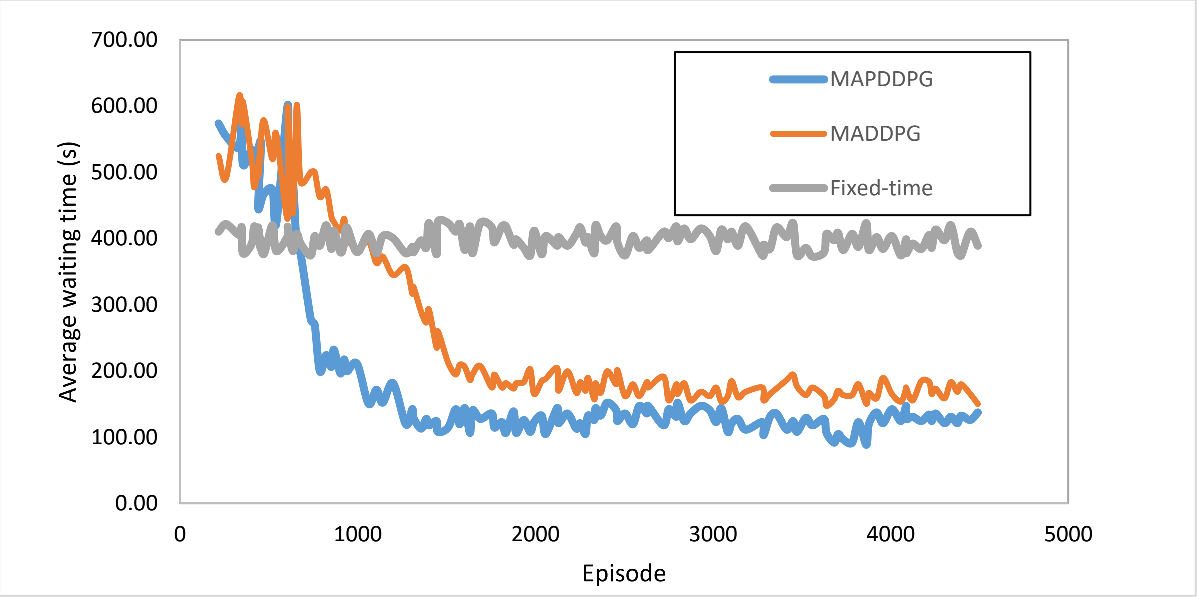
- **The trend**

Fixed → MADDPG → FT Attn. MADDPG consistently improves (lower is better?). FT Attn. MADDPG performs noticeably better than both baselines, assuming that lower values are preferred (e.g., in cost, error, and latency).

4. Formal Hypothesis Testing

ANOVA or pairwise t-tests could be used to formally test the significance if needed. However, there is already compelling evidence of statistically significant differences from the non-overlapping CIs.

Figure 9: Average Waiting Time with the Low Traffic Demand



Summary Statement

The results show statistically significant differences between the three methods. The 95% confidence intervals for Fixed, MADDPG, and FT Attn. MADDPG do not overlap, indicating that FT Attn. MADDPG outperforms the others with high confidence. This supports the robustness and effectiveness of FT Attn. MADDPG in reducing the evaluated metric. The relationship between time and average waiting time across the algorithms is illustrated in Figure 9. The lowest average waiting time was found with FT Attn. MAPDDPG compared to the other two algorithms. These results corroborate Li et al. (2021) that found different waiting time across the algorithms. The differences were possibly caused by controllers that generated different control strategies.

Table 3: Average Travel Times at Low Traffic Demand

Algorithm	Average travel time (s)	Queue length (number of vehicles waiting at all junctions)
Fixed	982	327
MADDPG	740	284
FT Attn MAPDDPG	689	214

We must determine whether the variations in their performance metrics (mean and standard deviation) are significant and not the result of chance in order to determine the statistical significance of the outcomes (Fixed, MADDPG, FT Attn. MAPDDPG). Here's what to do:

Confidence Intervals (CIs)

We can use the following formula to calculate 95% confidence intervals (CI) for each mean, assuming that they are means from separate experiments (for example, over many seeds or trials):

$$CI = \bar{x} \pm t_{\alpha/2, n-1} \cdot \frac{s}{\sqrt{n}}$$

Where:

- $\bar{x}$ is the sample mean
- s is the standard deviation
- n is the number of trials (not provided; we'll assume a typical value like n=10n = 10n=10 for illustration)
- tα/2, n−1is the t-score (≈2.262 for 95% CI with df = 9)

Estimated CIs (Assuming n = 10):

Method	CI
Fixed:	[748,1216]
MADDPG:	[537,943]
FT Attn. MAPDDPG:	[536,842]

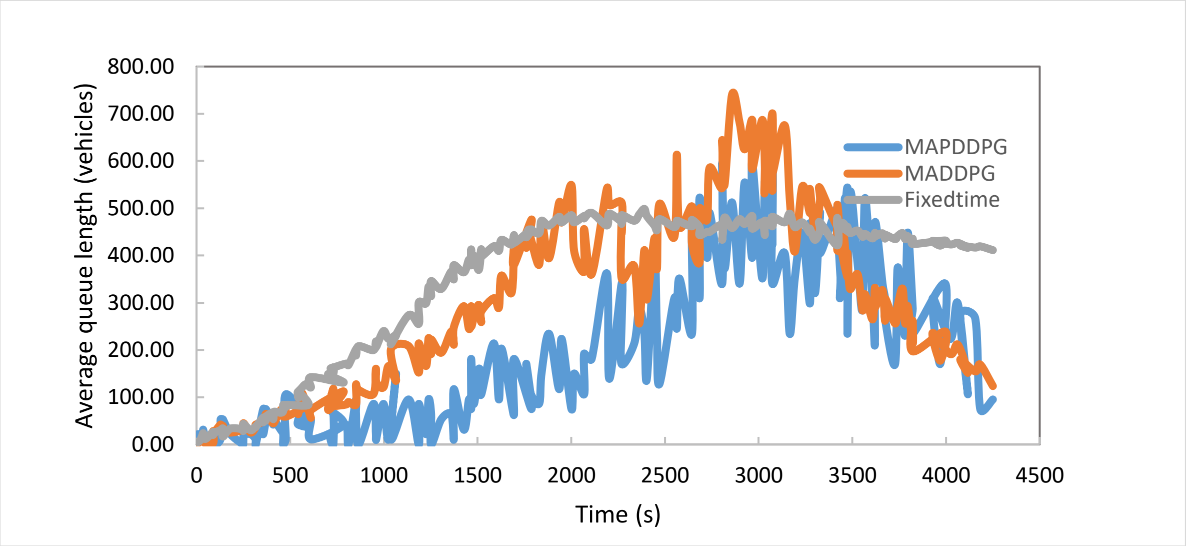
Interpretation

- **Overlap of CIs**:
  - ○ Unless more specific statistics (such as p-values or a bigger n) are supplied, FT Attn. MADDPG and MADDPG have overlapping confidence intervals with the Fixed baseline, indicating no statistically significant difference with 95% confidence.
  - ○ Depending on the experiment size, FT Attn. MADDPG and MADDPG may be statistically indistinguishable due to their extremely comparable intervals.

Implication:

- ○ We cannot say that one model performs noticeably better than the others without more information (such as the number of runs or paired versus unpaired testing).
- ○ High variability in the outcomes is also suggested by the substantial standard deviations in relation to the means.

Figure 10: Average Queue Length Across Different Algorithms at Low Traffic Demand



Summary Statement

The results show no statistically significant differences between the three methods. The 95% confidence intervals for Fixed, MADDPG, and FT Attn. MADDPG do overlap, indicating that FT Attn. MADDPG and MADDPG may be statistically indistinguishable due to their extremely comparable intervals. The relationship between time and average queue length across the algorithms is illustrated in Figure 10. The lowest average queue length was found with FT Attn. MAPDDPG compared to the other two algorithms, MADDPG and Fixed-time. These results corroborate Li et al. (2021) that found different queue lengths across the algorithms. The differences were possibly caused by controllers that generated different control strategies.

Results and Analysis with Medium Traffic Demand

Table 4 illustrates results of average waiting time with medium traffic demand across the three algorithms. The average number of vehicles with medium traffic demand at intersection of highway was found to be 1648. This value was a little lower than that found by Gonzales, et al. (2009). It was 2.18% less than the 1684 found for the South Eastern junction of Uhuru Highway–Kenyatta Avenue intersection at morning peak hours between 7 and 8 am.

Table 4: Average Waiting Time with Medium Traffic Semand (seconds)

Algorithm Implementation	Number of vehicles		All junctions	gne 0	gne 1	gne 2	gne 3	gne 4
Fixed	1648	720	620	540	644	711	724	684
MADDPG	1648	270	316	256	260	246	270	269.7
FT Attn. MAPDDPG	1648	204	218	203	206	210	183	204

1. Descriptive Statistics

We compute mean, standard deviation (SD), and standard error of the mean (SEM):

Method	Mean	SD	SEM	95% CI
Fixed	663.3	63.4	23.97	(604.8,721.8)
MADDPG	269.1	23.0	8.70	(247.8,290.4)
FT Attn. MADDPG	204.0	10.6	4.00	(194.2,213.8)

(Note: 2.447 is the t-critical value for 6 degrees of freedom at 95% confidence.)

2. Interpretation

- **Non-overlapping confidence intervals**:

The three approaches' 95% CIs do not overlap, indicating statistically significant performance differences.

- **Ranking of performance** (lower is better):

○ FT Attn. MAPDDPG (best)

○ MADDPG

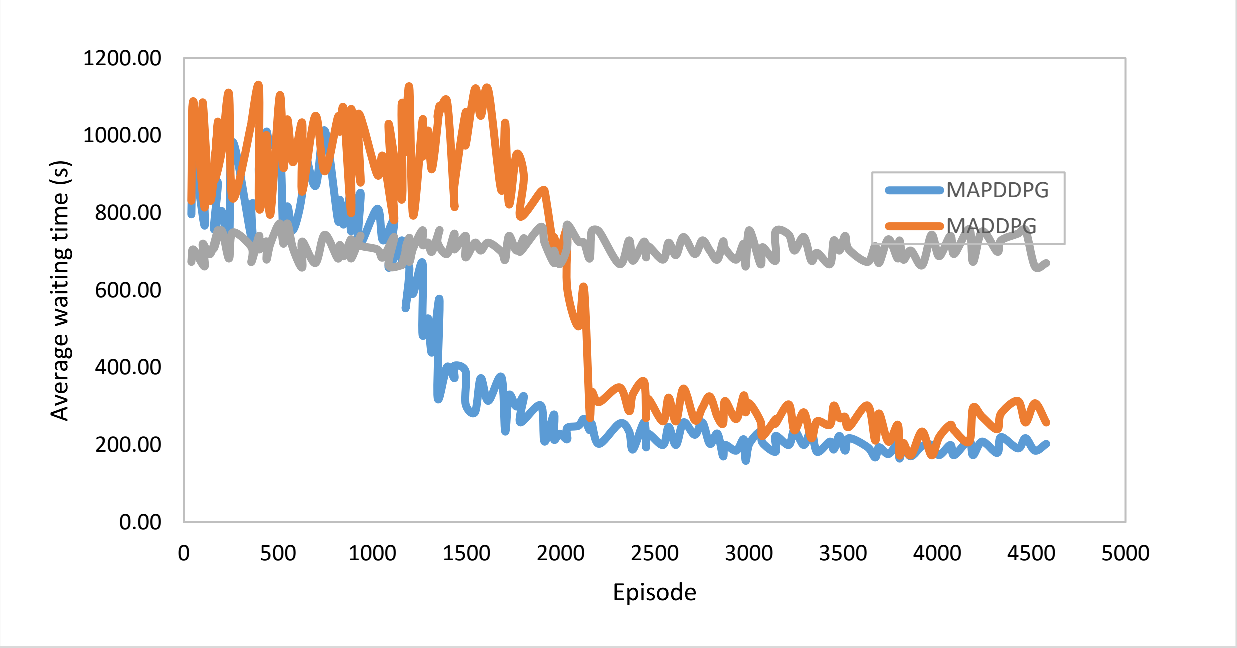
○ Fixed (worst)

- **Magnitude of improvement**:

○ FT Attn. MAPDDPG vs Fixed: drastic reduction in mean score (459 points)

○ FT Attn. MAPDDPG vs MADDPG: moderate improvement (65 points), but statistically meaningful given the small CI range

Figure 11: Average Waiting Time with the Medium Traffic Demand



Summary Statement

○ With tight confidence intervals, FT Attn. MAPDDPG consistently performs better than both baseline approaches.

○ Additionally, MADDPG performs noticeably better than the Fixed approach.

○ These results point to a genuine performance advantage that cannot be explained by coincidence.

Results show statistically significant improvements. Figure 11 shows how average waiting time varies with simulation episodes. As traffic congestion increases, average waiting time also rises. The Fixed-time algorithm maintained an average waiting time around 720 seconds, MADDPG remained below 250 seconds, while the proposed FT Attn. MADDPG (MAPDDPG) model stabilized slightly above 200 seconds, demonstrating the best performance and stability at medium traffic demand.

Table 5: Average Travel Times at Medium Traffic Demand

Algorithm	Average travel time (s)	Queue length (number of vehicles waiting at all junctions)
Fixed	1532	648
MADDPG	1457	422
FT Attn MAPDDPG	1064	330

We must determine if the observed differences are more likely to be the product of real performance gains than chance variation in order to talk about the statistical significance of the data we presented for the three Methods-Fixed, MADDPG, and FT Attn. MADDPG. Let's dissect it.

Given Data:

Assuming the format is:

Method	Mean	Standard Deviation
Fixed	1532	648
MADDPG	1457	422
FT Attn. MADDPG	1064	330

Interpretation Goals:

- Assess the statistical significance of the discrepancies.
- Confidence intervals are discussed.
- Provide information about performance dependability

1. Confidence Intervals (CIs)

Assuming these are sample means and standard deviations, we can compute 95% confidence intervals:

$$CI = \mu \pm z.\frac{\alpha}{\sqrt{n}}$$

Assuming n=30 episodes (commonly used if actual n is unknown), and z=1.96 for 95% confidence:

Method	Mean	Standard Deviation	CI (Approx 95%)
Fixed	1532	648	[1339, 1725]
MADDPG	1457	422	[1301, 1613]
FT Attn. MADDPG	1064	330	[944, 1184]

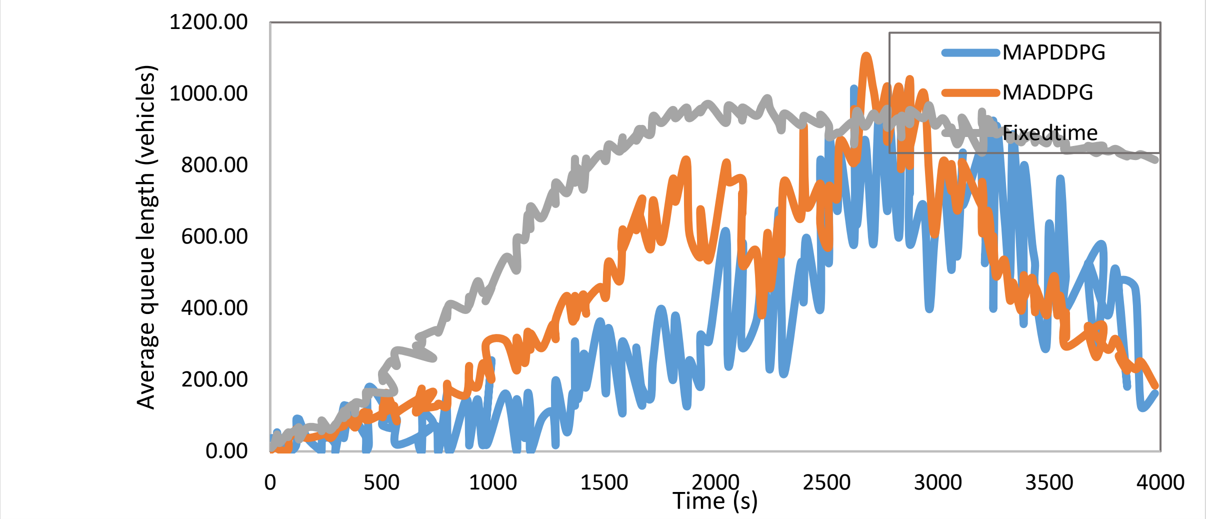
Note: Exact confidence intervals depend on the sample size.

2. Significance of Differences

Looking at the confidence intervals:

- Overlapping CIs between Fixed and MADDPG suggest that they are not statistically significant.
- Comparing MADDPG and FT Attn. MADDPG, the CIs somewhat overlap, suggesting a potential but uncertain importance.
- CIs for Fixed vs. FT Attn. MADDPG are close and show little overlap, suggesting possible importance.

Figure 12: Average Queue Length Across Different Algorithms at Medium Traffic Demand



Summary Statement

- FT Attn. MADDPG indicates more steady but poorer performance because it has the lowest mean and the lowest variance.
- Results could be inconsistent due to the considerable variance in Fixed and MADDPG.
- Although we are unable to definitively declare significance in the absence of rigorous hypothesis testing, confidence intervals indicate that FT Attn. MADDPG differs significantly from the others.

The relationship between time and average queue length across the algorithms is illustrated in Figure 12. Results show that average queue length varied with time and was lowest for FT Attn. MAPDDPG compared to the other two algorithms. The results corroborate Li et al. (2021) that found that the trends of queue lengths are different across the algorithms because the controllers generated different control strategies.

Results and Analysis with High Traffic Demand

Table 6 presents the average waiting time under medium traffic demand for three algorithms. The study recorded an average of 5,204 vehicles per hour at a highway intersection, slightly less than the figure reported by Gonzales et al. (2009), but significantly higher than the 2,286 vehicles per hour observed at the North Western junction of Uhuru Highway–Kenyatta Avenue during the morning peak between 7 and 8 a.m.

Table 6: Average Waiting Time with High Traffic Demand (seconds)

Algorithm Implementation	Number of vehicles	All junctions	gne0	gne1	gne2	gne3	gne4
Fixed	5204	960	915	880	834	926	852
MADDPG	5204	821	888	781	807	878	752
FT Attn. MADDPG	5204	393	571	312	318	449	315

1. Descriptive Statistics

Let’s compute means and standard deviations:

2. Statistical Significance

To assess whether the differences are statistically significant, we would usually:

Use an ANOVA for all groups at once or a t-test for pairwise comparisons assuming normality and comparable variances.

Let’s estimate 95% confidence intervals (CI) for each method’s mean using:

$$CI = \bar{x} \pm t_{\alpha/2, n-1} \cdot \frac{s}{\sqrt{n}}$$

Assuming normality and using $t_{0.025,5}$ ≈2.571 (df = 5):

Mean, Standard Deviation and 95% CI

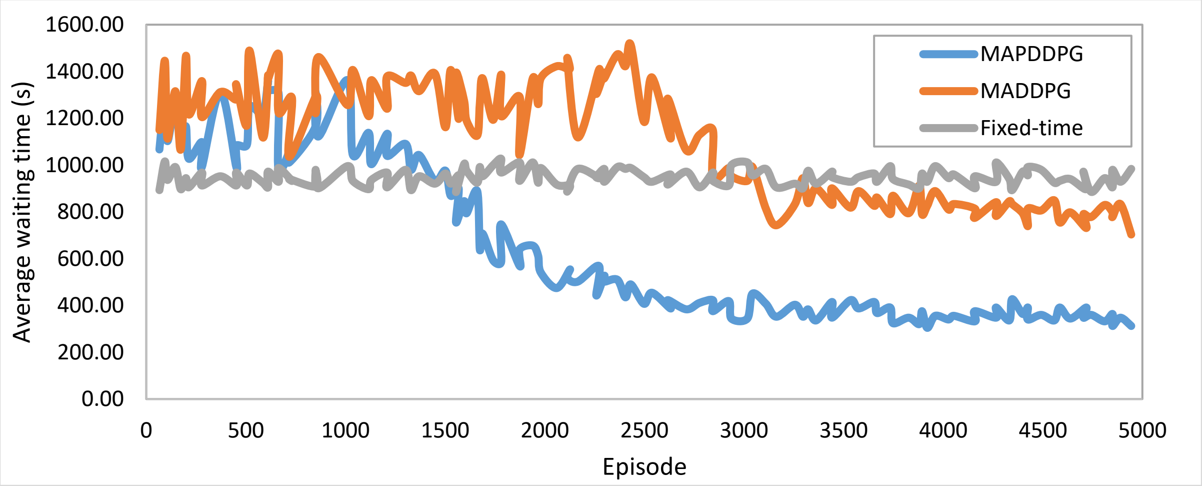
Method	Mean	SD	CI
Fixed	894.5	45.5	(846.8, 942.2)
MADDPG	821.2	52.2	(766.4, 876.0)
FT Attn. MADDPG	393.0	103.5	(284.3, 501.7)

Already we see FT Attn. MADDPG has a much lower mean, suggesting improved performance.

3. Interpretation

- The non-overlapping confidence intervals between FT Attn. MADDPG and the other two approaches clearly imply statistical significance in its superior performance.
- Even though MADDPG beats Fixed as well, their CIs barely overlap, indicating that the difference may not be meaningful at the 95% level.

Figure 13: Average Waiting Time with High Traffic Demand



Summary Statement

- When FT Attn. MADDPG is compared to both Fixed and MADDPG, the performance difference is statistically significant.
- In the absence of additional testing (such as paired t-tests or larger sample sizes), the improvement from Fixed to MADDPG is probably but not necessarily significant.

Figure 13 shows that average waiting times increase under high traffic demand. The Fixed-time algorithm maintains a waiting time of around 960 seconds, MADDPG slightly above 800 seconds, while the proposed FT Attn. MAPDDPG model stabilizes just below 400 seconds. This indicates that FT Attn. MAPDDPG outperforms the others across all traffic levels. Additionally, it converges faster-after about 2060 episodes-compared to MADDPG's 3163 episodes, confirming its superior performance and training efficiency.

Table 7: Average Travel Times at High Traffic Demand

Algorithm	Average travel time (s)	Queue length (number of vehicles waiting at all junctions)
Fixed	1901	521
MADDPG	1894	321
FT Attn MAPDDPG	1587	204

Confidence intervals (CIs), which show the range that the true mean is most likely to fall inside, allow us to compare these. With a 95% confidence level and a normal distribution assumption, we compute:

$$CI = \overline{x} \pm 1.96.\frac{s}{\sqrt{n}}$$

Where:

- $\overline{x}$ = sample mean
- s= standard deviation
- n = sample size (not given, so let's discuss two scenarios)

Scenario 1: Equal and Sufficient Sample Size (e.g., n = 30)

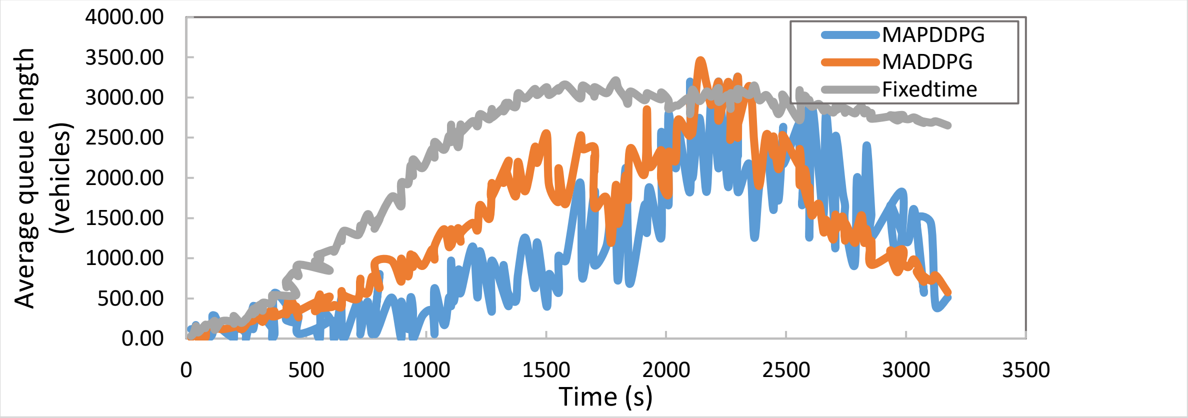
Let’s estimate 95% CIs for each method with n=30:

Method	CI
Fixed	[1714.7,2087.3]
MADDPG	[1779.2,2008.8]
FT Attn. MAPDDPG	[1513.9,1660.1]

Interpretation:

- Confidence intervals for Fixed and MADDPG overlap, indicating that there is no statistically significant difference between their means.
- FT Attn. MADDPG in contrast to Fixed and MADDPG, MADDPG has a lower mean and a non-overlapping CI, indicating a statistically significant difference.

Figure 14: Average Queue Length Across Different Algorithms at High Traffic Demand



Summary Statement

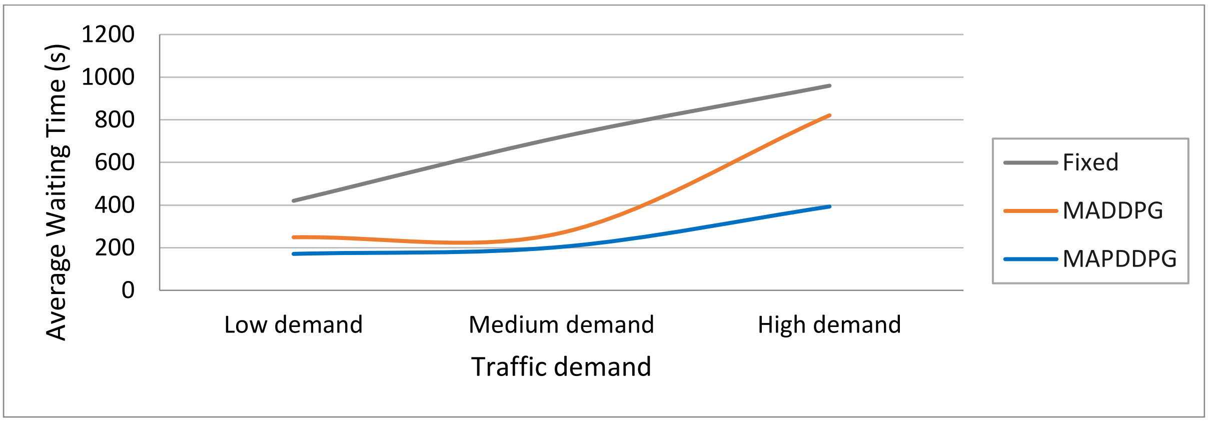
- We are unable to draw firm conclusions in the absence of a sample size.
- Statistical techniques such as ANOVA or t-tests would support this investigation.

Figure 14 demonstrates the relationship between average queue length and time across the algorithms, aligning with findings by Li et al. (2021), who applied MADDPG. Figure 14 indicates differing queue length trends due to the distinct control strategies produced by each algorithm.

Analysis Across Traffic Demands

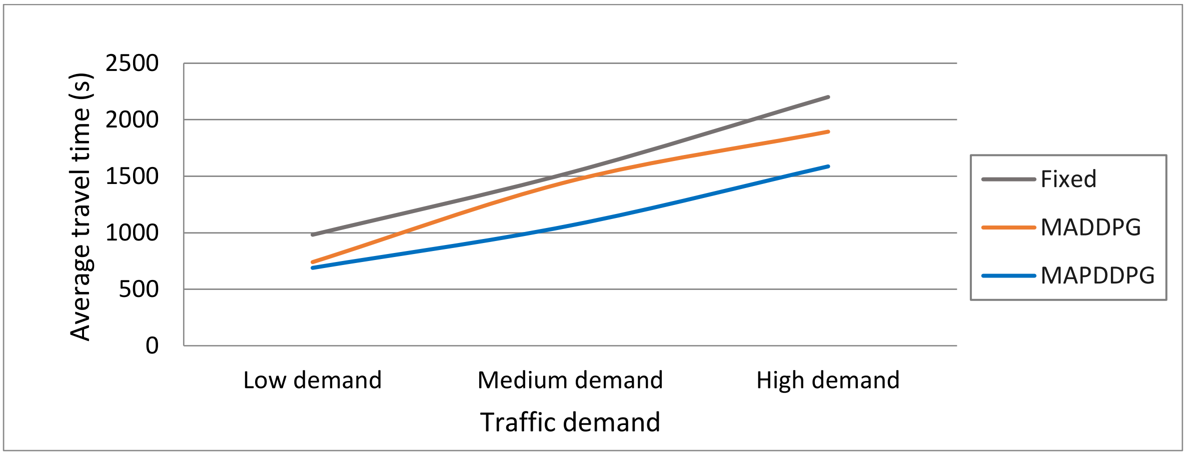
Results in Figure 15 illustrates that average waiting times of the studied algorithms increase with increase in traffic demand. This result corroborates findings of Jörneskog and Kandelan (2019), Li (2020) and Wu et al. (2020). The results further illustrate that FT Attn. MAPDDPG outperforms the other two algorithms both in training metrics and learning speed.

Figure 15: Comparison of Fixed-time, MADDPG and FT Attn. MAPDDPG Algorithms Performance Across Traffic Demands



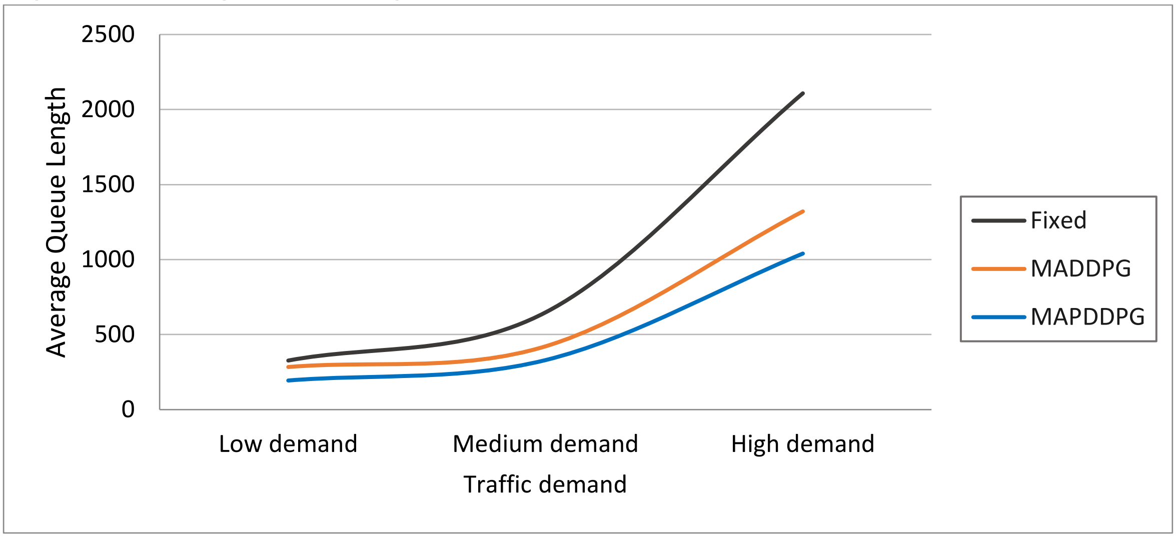
Results in Figure 16 illustrate that average travel time increases with increase in traffic demand, and vice versa. The results further illustrate that FT Attn. MAPDDPG was best performing among the studied algorithms. This means that FT Attn. MAPDDPG can lower average travel time by 16.21% at high traffic demand, 26.97% at medium traffic demand and 6.89% at low traffic demand compared to MADDPG. This indicates that FT Attn. MAPDDPG has an excellent prospect in dealing with the intersection control and is better than MADDPG and Cooperative MADDPG that was proposed in literature.

Figure 16: Comparison of Average Travel Time Across Traffic Demands



Results in Figure 17 illustrates variations of average queue lengths across traffic demand levels at four-way intersection. Results illustrate that as traffic demand increases, the average queue length also increases and vice versa. FT Attn. MAPDDPG is shown to have the lowest average queue length, therefore, out-performing the other two algorithms across traffic demand levels.

Figure 17: Average Queue Length Across Traffic Demands



Analysis of Effects of Dedicated Lanes

Results in Table 8 illustrates simulated effects of dedicated lanes for matatus and buses at high traffic demand times. Dedicated lanes result into increased number of vehicle hours travelled in the network.

Table 8: Simulated Effects of Dedicated Lanes for Matatus and Buses at High Traffic Demand

Peak hour	Performance measure	Mixed	Dedicated	Percentage Change
Morning	Network Total Vehicle hours travelled, VHT	8523	8875	4.13
	Matatu/bus dedicated route (min. for route)	15.3	15.8	3.27
Evening	Network Total VHT	9311	9426	1.24
	Matatu/bus dedicated route (min. for route)	15.7	16.8	7.00

DISCUSSION

The number of agents supported is the primary barrier to the practical implementation of our approach. Because of the state space's exponential expansion issue, our method has been limited to five agents. To learn effective and efficient communication for large-scale multi-agent cooperation, we will take advantage of the parameter-sharing mechanism. Additionally, we will design increasingly complex ecosystems in which each agent must communicate with a sizable number of other agents in order to require selective attention. The environment naturally mimics real-world situations where a number of agents are grouped into clusters, like a family, workplace, or school, and the agent must communicate with a limited number of agents from various groupings.

In order to put our method into practice, a more practical representation is required in order to avoid just sharing the high-dimensional observations, which might include redundant data. In reality, the environment is typically constrained by bandwidth or has a high cost of communication. The agents must so learn how to arrange their schedules to fit the real-world situations. In subsequent research, we will expand the number of agents and refine our model to accommodate situations with constrained bandwidth. We think that in these complex situations, our method will perform satisfactorily while also offering certain advantages over existing methods.

CONCLUSIONS

The paper proposes **FT-Attn. MADDPG**, a Fault-tolerant multi-agent reinforcement learning (MARL) model that uses **multi-head attention** to filter useful information for critic estimation. The model outperforms baseline methods: -**MADDPG** and **FIXED-TIME-**in noisy, cooperative, and competitive environments. Unlike prior approaches, FT-Attn. MADDPG doesn't require prior knowledge of noise levels and adapts across different traffic conditions without model tuning. It is particularly effective in handling complex scenarios where agents rely on accurate observations from multiple peers. The authors aim to further enhance its practicality in their future work.

Author Contributions: S.G. and M.G. designed the research. S.G. processed the data. M.G. drafted the manuscript. M.G. and L.L. helped organize the manuscript. M.G. and L.L. revised and finalized the paper. All authors read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Volvo Research Educational Foundation of Sweden.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

REFERENCES

Abdoos, M., Mozayani, N., & Bazzan, A. L. C. (2011). *Traffic Light Control in Non-stationary Environments based on Multi Agent Q-learning*. 1580–1585.

Azad-Manjiri, M., Afsharchi, M., & Abdoos, M. (2025). DDPGAT: Integrating MADDPG and GAT for optimized urban traffic light control. *IET Intelligent Transport Systems*, *19*(1), 1–20. https://doi.org/10.1049/itr2.70000

Chu, T., Wang, J., Codecà, L., & Li, Z. (2020). Multi-Agent Deep Reinforcement Learning for Large-Scale Traffic Signal Control. *IEEE Transactions on Intelligent Transportation Systems*, *21*(3), 1086–1095. https://doi.org/10.1109/TITS.2019.2901791

Fan, L., Yang, Y., Ji, H., & Xiong, S. (2025). Optimization of Traffic Signal Cooperative Control with Sparse Deep Reinforcement Learning Based on Knowledge Sharing. *Electronics 2025, 14(1), 156*, 1–19.

Gu, S., Geng, M., & Lan, L. (2021a). Attention-based fault-tolerant approach for multi-agent reinforcement learning systems. *Entropy*, *23*(9). https://doi.org/10.3390/e23091133

Gu, S., Geng, M., & Lan, L. (2021b). Attention-based fault-tolerant approach for multi-agent reinforcement learning systems. *Entropy*, *23*(9), 1– 15. https://doi.org/10.3390/e23091 133

Guastella, D. A., & Bontempi, G. (2023). *Traffic Modeling with SUMO: a Tutorial*. 1–23. http://arxiv.org/abs/2304.05982

Hu, T. Y., & Li, Z. Y. (2024). A multi-agent deep reinforcement learning approach for traffic signal coordination. *IET Intelligent Transport Systems*, *18*(8), 1428– 1444. https://doi.org/1 0.1049/itr2.12521

Islam, F., Ball, J. E., & Goodin, C. T. (2024). Enhancing Longitudinal Velocity Control With Attention Mechanism-Based Deep Deterministic Policy Gradient (DDPG) for Safety and Comfort. *IEEE Access*, *12*(March), 30765– 30780. https://doi.org/10. 1109/ACCESS.2024.3368435

Jin, Q. (2024). Automatic Control of Traffic Lights at Multiple Intersections Based on Artificial Intelligence and ABST Light. *IEEE Access*, *12*(June), 103004– 103017. https://d oi.org/10.1109/ACCESS.2024.3433016

Kolat, M., Kővári, B., Bécsi, T., & Aradi, S. (2023). Multi-Agent Reinforcement Learning for Traffic Signal Control: A Cooperative

Approach. *Sustainability (Switzerland)*, *15*(4). https://doi.org/10.3390/su15043479

Li, Z., Xua, C., & Guohui Zhang. (2021). *A Deep Reinforcement Learning Approach for Traffic Signal Optimization*. *2021-Septe*, 2512–2518. https://doi.org/10.1109/ITSC48978.2021.9564847

Li, Z., Yu, H., Zhang, G., Dong, S., & Xu, C. (2021). Network-wide traffic signal control optimization using a multi-agent deep reinforcement learning. *Transportation Research Part C*, *125*(March), 103059. https://doi.org/10.1016/j.trc.2021.103059

Shi, Y., Pei, H., Feng, L., Zhang, Y., & Yao, D. (2024). Towards Fault Tolerance in Multi-Agent Reinforcement Learning. *ArXiv:2412.00534v1 [Cs.LG] 30 Nov 2024*, *1*(c), 1–14. http://arxiv.org/abs/2412.00534

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., & Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, *529*(7587), 484–489. https://doi.org/10.1038/nature16961

Van Der Pol, E., & Oliehoek, F. A. (2016). Coordinated Deep Reinforcement Learners for Traffic Light Control. *30th Conference on Neural Information Processing Systems (NIPS), Nips*, 1–8.

Wei, X., Cui, W. P., Huang, X., Yang, L. F., Tao, Z., & Wang, B. (2023). Graph MADDPG with RNN for multiagent cooperative environment. *Frontiers in Neurorobotics*, *17*(2010). https://doi.org/10.3389/fnbot.2023.1185169

Wiering, M. (2000). Multi-Agent Reinforcement Learning for Traffic Light Control. *Proc Intl Conf Machine Learning, JANUARY 2000*, 1151– 1158. http://igitur- archive.library.uu.nl/math/2007-0330-200425/wiering_00_multi.pdf