Multi-Agent Deep Reinforcement Learning for Traffic Optimization through Multiple Road Intersections using *Live* Camera Feed

Deepeka Garg, Maria Chli and George Vogiatzis

***Abstract*— Traffic signals provide one of the primary means to administer conflicting traffic flows. Existing signal control strategies, operating on hand-crafted rules, fail to efficiently, autonomously adapt to the changing traffic patterns. Each signal control system independently manages one intersection at a time and regulates navigation of vehicles through that intersection. Current systems cannot co-operate to optimize aggregate traffic flows through multiple road intersections. Consequently, they are susceptible to making myopic signal control decisions that might be effective locally, but not globally. Instead, we propose a system of multiple, coordinating traffic signal control systems. This paper presents the first application of multi-agent deep reinforcement learning (DRL) to achieve traffic optimization through multiple road intersections *solely* based on raw pixel input from CCTV cameras in real time. This set of traffic control agents is shown to significantly outperform independently operating (both DRL-trained and loop-induced) adaptive signal control systems, by increasing traffic throughput and reducing the average time a vehicle spends in an intersection. Additionally, this paper, introduces attention-based visualization to interpret and validate the proposed multi-agent signal control methodology.**

I. Introduction

Traffic congestion is a serious problem, costing substantially to drivers in terms of wasted fuel and time. Among others, in the urban road networks, inadequate traffic signal timings is one of the repeated causes of congestion. Exerting real-time, adaptive control is potentially useful in a variety of intelligent transportation systems applications, including signal control. The primary limitation of conventional signal control methods is the need for pre-specified models of the traffic environment. The purpose of having pre-specified models is to effectively visualize the picture of present or imminent traffic conditions. The pre-specified traffic environment specifications are required to be constructed by the domain experts. Furthermore, these models must be generic enough to cover a variety of traffic conditions, as it can be impractical to have a separate model independently demonstrating each potential traffic situation. However, a generalized traffic model may lack the ability to reliably reflect the full range of traffic flow patterns. For instance, TRANSYT; one of the state-of-the-art adaptive signal control systems, only uses a platoon-dispersion model to determine the arrival pattern of vehicles, irrespective of the prevailing traffic conditions.

Fig. 1: A view of Traffic3D's graphical display (the simulation environment used in our experiments).

Reinforcement Learning (RL) [1] is a successful paradigm, that obviates the need of pre-specification of the environment model. The environment RL agents operate in, is not known in advance. Instead, RL agents monitor their environment through perception, influence it by implementing actions and learn by observing the outcomes of their actions. RL was first applied to traffic signal control in the 1990s, with the first techniques limited to tabular Q-learning [2]. Traditional RL methods suffered from limited scalability and optimality in practice. However, in recent years, deep neural networks (DNNs) have proven their effectiveness in significantly improving the performance of RL methods; facilitating end-to-end learning (i.e. mapping from sensory inputs to action outputs) and completely eliminating the need for hand engineering of task-specific features by domain experts [3]. To accomplish a particular task, the DRL (Deep Reinforcement Learning) agent *learns* the set of environment features that are significant in each task. These agents derive efficient representations of high-dimensional, raw sensory data (such as videos and images) and subsequently utilize these to generalize the past experience to new unseen situations. In recent papers [4], [5], we presented a truly adaptive signal control agent that directly responds to the actual traffic conditions in a single intersection; achieving effective signal control in the face of complex, imprecise traffic environment. In this paper, to further establish the resilience of our DRL-based signal control approach, we investigate the utilization of multi-agent DRL to real-time adaptive traffic signal control through a *network* of road intersections.

A number of RL tasks (such as autonomous driving and robotic manipulation) can be naturally modelled as cooperative multi-agent systems. However, RL agents attempting to single-handedly solve these tasks perform poorly, as their joint state and action spaces grow exponentially,

The authors are based at the Computer Science Department, School of Engineering and Applied Science, Aston University, Birmingham, United Kingdom, e-mail: {gargd, m.chli, g.vogiatzis}@aston.ac.uk.

leading to dimensionality explosion. This paper, for the first time, establishes network-level coordination between multiple DRL-based signal control agents operating on visual traffic data (i.e. the agents *solely* operate on camera feed to optimize an aggregate of traffic flows through a network of intersections). Having the ability to visually perceive the prevailing traffic state enables our signal control agents to extensively process the traffic environment and subsequently learn its intricate feature representations (including vehicle types and their precise positions) that would otherwise be tedious to determine using widely-used traffic data collection methods (such as induction loops and microwave detector [6]).

Incorporating the concepts and perspectives from recent work in the field of multi-agent planning [7], [8], to achieve network-level coordination between individually operating local signal control agents, we apply an *actor-critic* [9] RL approach. We implement a centralised critic that enables global learning, while each actor's execution is local. To achieve network-level optimality, the centralised critic operates on all the available state information (i.e. concatenation of local states of the collaborating signal control agents). In contrast, each actor (i.e. each participating signal control agent) operates exclusively on its own limited local observation of the environment. We compare our proposed centralised learning method against baseline methods: (1) fully-decentralised learning (outlined in Sec. V-A), (2) fully-independent learning (outlined in Sec. V-B), and (3) loop-induced signal control (outlined in Sec. V-C). Our experiment-based evaluations reveal that our research approach leads to a positive emergence of coordinated behavior between individual signal control agents; resulting in significant performance improvement over above mentioned baseline methods. To further demonstrate the effectiveness of our research methodology, in this paper, we work towards explainable Artificial Intelligence and for the first time illustrate the interpretability of our signal control agents' decisions using attention-based visualization (Sec. VII-D).

II. Related Work

Conventional signal control methods; independently optimizing the flow of traffic through one intersection at a time, operate on pre-programmed *signal regime plan*. The phase time interval may change based on the peak or quiet hours, but they are not otherwise optimized. However, over the years, as the volatility of traffic patterns outpaced the effectiveness of pre-programmed signal control methods, interdisciplinary methods (such as RL) are being studied to adaptively configure signal regimes. There exists a large body of work on RL-based adaptive signal control, however, the majority of recent studies are conducted on single junctions, using relatively simplified traffic state information (e.g. a vector specifying the presence of vehicles at an intersection and their respective speed information) [10], [11]. Our work, in contrast, utilizes visual inputs, rendering an extensive representation of the prevailing traffic state (including flows, types of vehicles, weather conditions, etc.) to decide the configuration of signal regimes across a network of junctions.

Only a handful of studies address signal control optimization through multiple intersections. In [12], tabular Q-learning is applied to each intersection in a multiple intersection traffic environment. This work is further extended in [13], in which traffic regions are dynamically clustered to improve observability. In [14], both Q-learning and SARSA are used, with traffic state observability enhanced using neighbourhood information sharing. Tantawy et al. [15] implemented a heuristic communication between tabular Q-learning-based intersection control agents, in which each message consisted of the estimated neighboring agents' signal control policies. Chu et al. [16] used the max-sum communication for Q-learning-based intersections, in which each message signified the impact of a neighbouring intersection on each local Q-value. Most of these research studies implement value function-based approaches (Q-learning, SARSA) for traffic optimization. Value function-based methods are often criticized for being unstable and in practice are difficult to use. They are inclined towards finding deterministic policies, whereas in a dynamic environment like traffic, an effective policy is expected to be stochastic.

Closest to our work [17], traffic is optimized through a network of intersections in a decentralized fashion. The authors devise a fully-decentralized multi-agent signal control method. In each local agent's state observation information, observations and fingerprints of neighboring agents are included such that each local agent is more aware of regional traffic distribution, while we use a centralized critic and decentralised actors to perform centralised learning and decentralised execution. Furthermore in [17], handcrafted traffic state features (i.e. cumulative delay of first vehicle and number of vehicles approaching an intersection within 50m range to the intersection) are used. In contrast, our signal control methodology is end-to-end trainable and, to our knowledge, is the first to depend *solely* on camera feed for traffic optimization in real time. Deep learning models are known to offer insights that go well beyond human understanding. In this paper, we analyse our agents' decision-making through specialised visualisation techniques (Sec. VII-D).

III. Background and Notation

In this section, we introduce our signal control agents' implementation-based on deep reinforcement learning.

A. Deep Reinforcement Learning (DRL)

In a basic RL setting, an agent aims to achieve a goal by interacting with an uncertain environment. A standard RL framework is mathematically modelled as a Markov Decision Process (MDP), which can be represented as a tuple $< S, A, T, R, \gamma >$, where S and A are the state and action spaces respectively. $\gamma \in (0, 1)$ denotes the discount factor, which models the relevance of immediate rewards over the future rewards. After observing a state, an agent working under the policy $\pi : S \mapsto A$ produces an action.

Given current state s_t and action a_t, the transition function $T : S \times A \times S \mapsto \mathbb{R}^+$ determines the distribution of the next state s_{t+1}. The reward function R is determined by $R : S \times A \mapsto \mathbb{R}$. An episode $\tau \sim \mathcal{M}$ with horizon H is a sequence of state, action, reward $(s_0, a_0, r_0, \ldots, s_H, a_H, r_H)$ at every time-step t. The discounted episodic return of τ is determined by $R_t = \sum_{t=0}^{H} \gamma^t r_t$. Given the agent's policy π, the expected episodic return is defined by $E_\pi[R_\tau]$. The expected episodic return is maximized by optimal policy π^*

$$\pi^* = \arg \underset{\pi}{\max} E_{\tau\sim\mathcal{M},\pi}[R_\tau]. \tag{1}$$

A deep neural network (π_θ) with parameters θ in high-dimensional RL settings represents policy π^*. The agent aims to learn θ^* that achieves highest expected episodic return,

$$\theta^* = \arg \underset{\theta}{\max} E_{\tau\sim\mathcal{M},\pi}[R_\tau]. \tag{2}$$

In actor-critic RL [9], the actor is the policy $\pi^\theta(a|s)$ with parameters θ, based on which actions are estimated, while the critic computes value functions to help the actor in learning. Action and value function are estimated using function approximators and the gradient is estimated from trajectories sampled from environment. R_t is replaced by an expression equivalent to $Q(s_t, u_t) - b(s_t)$, where $b(s_t)$ contributes in reducing the variance. If R_t is replaced by $A(s_t, u_t)$, then $b(s_t) = V(s_t)$. $R(t)$ can also be replaced by the *temporal difference* error; $r_t + \gamma V(s_{t+1}) - V(s)$, which is an unbiased estimate of $A(s_t, u_t)$.

B. *Multi-agent Reinforcement Learning (MARL)*

In this paper, we consider a network of signal control agents (forming a multi-agent system). In this setting, our goal is to train signal control agents to effectively participate in optimizing traffic flows at a global network level. We consider a multi-agent extension of the Markov Decision Processes (MDPs); defined by a tuple $E =< S, U, P, r, Z, O, n, \gamma >$, where n agents (represented by $a \in A \equiv [1, ..., n]$) act in the environment E. The true state of the environment is represented as $s \in S$. At each time-step, each agent independently, simultaneously chooses an action $u^a \in U$, forming a joint action space $u \in U \equiv U^n$, which produces a transition in the environment (represented by $P(s\prime \mid s, u)\colon S \times U \times \mapsto [0, 1]$). For their individual selected actions, the agents receive their individual rewards; $r(s, u)\colon S \times U \mapsto \mathbb{R}$ and $\gamma \in (0, 1)$ denotes the discount factor. Given the real-world traffic complexity, we consider partially observable traffic settings, where each agent acts on its local observations $z \in Z$ (based on the observation function $O(s, a)\colon S \times A \mapsto Z$). Each agent depends on an action-observation history (represented by $\tau^a \in T \equiv (Z \times U)^*$), based on which it conditions a stochastic policy $\pi^a(u^a \mid \tau^a)\colon T \times U \mapsto [0, 1]$. The discounted return is denoted by $R_t = \sum_{t=0}^{\infty} \gamma^l r_{t+l}$. The agents' goal is to learn a policy that maximizes their expected discounted returns.

IV. Our Autonomous Multi-Intersection Signal Control Methodology

In this section, we describe the implementation for our signal control agents; including the MDP settings; state, action, reward specifications.

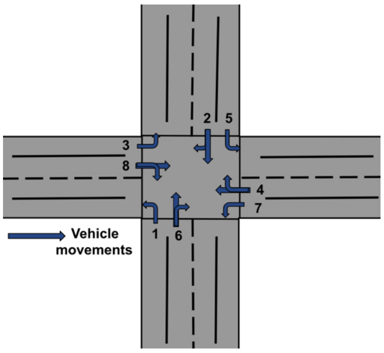


Fig. 2: Possible Signal Phases and Vehicle Movements.

A. *Problem Definition*

Under our research methodology, we consider a multi-intersection road scenario in which a set of signal control agents act in parallel. Each signal control agent controls one intersection in the network by directly mapping RGB images (describing the prevailing traffic state) to actions (controlling the traffic signals). Our goal is to achieve effective coordination of agents' actions such that their joint utility is maximized.

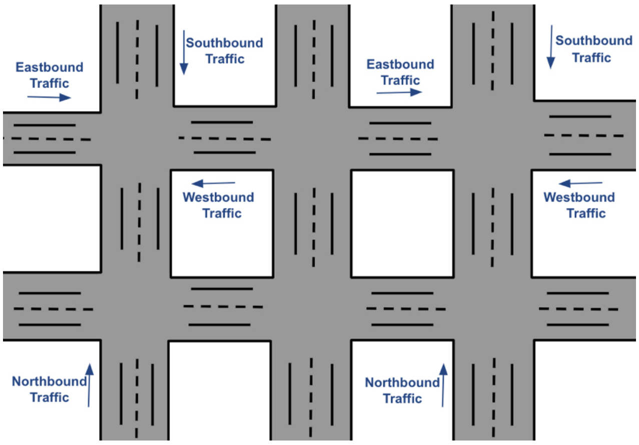


Fig. 3: An illustration of Intersection Grid.

B. *Traffic Model Simulation*

Due to economic and safety concerns, an agent cannot be trained via DRL to autonomously control traffic signals in real world. Simulation is deemed as a safe, cost-effective, controlled tool catalyzing protocol development. All the experiments presented in this paper are conducted using our novel traffic simulation environment; Traffic3D [18], [19]. Traffic3D is publicly available at https://traffic3d.org/en/latest/. For the current work, we simulated 3D four-way intersection scenarios with microscopic

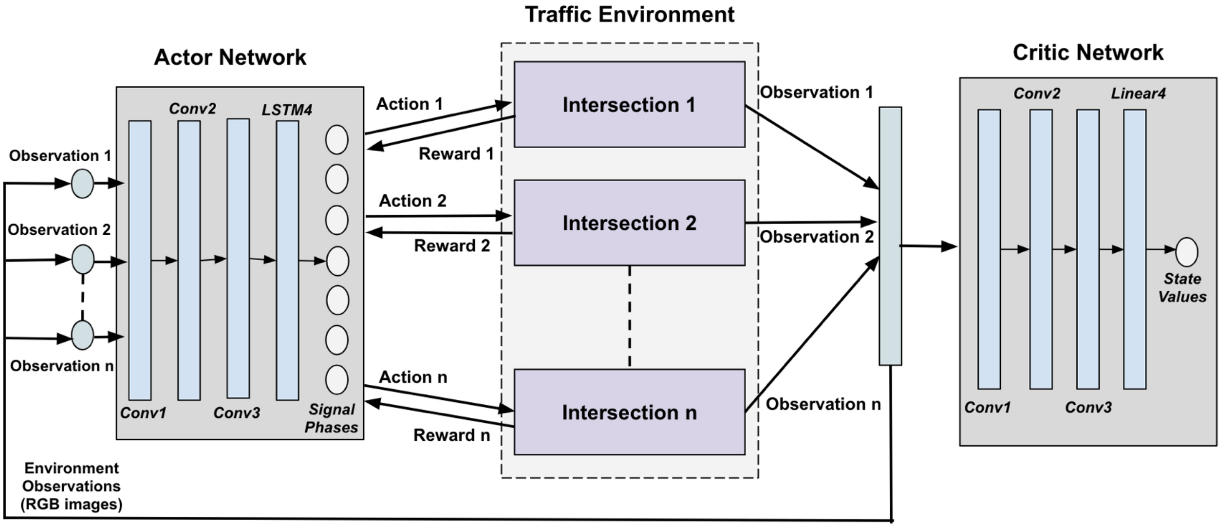


Fig. 4: Our Multi-Intersection *Actor-Critic* Network Framework. We use network parameter sharing (described in Sec. IV-F) to implement one *actor* and one *critic* network, which is shared by all the agents.

traffic properties. We investigated real-world traffic data and conducted sensitivity analysis of key traffic environment (such as weather and lighting conditions) and vehicle simulation (such as distribution of maximum speeds, lane and car-following behavior etc.) parameters to calibrate our simulator's parameters - both physical (supported by NVIDIA's Physx [20]) and visual (using real-time global illumination [20]).

C. Traffic Movement Simulation

Traffic movement is defined as the vehicles navigating across an intersection (from an entrance lane to an exit lane). Based on real-world guidelines, we define a set of possible, non-conflicting vehicle movements to allow their safe passage through the intersections (illustrated in Fig. 2) [6]. Signal phases are configurable and it is possible to have simultaneous execution of more than one phases. Vehicles can either go straight or turn right/left, route selection probability is parameterizable in our simulator. Fig. 3 illustrates our intersection network grid. Each intersection is a four-way intersection.

D. Learning Environment Setup: MDP Settings

Our simulated traffic environment is illustrated in Fig. 1. At each MDP time-step, concurrently operating signal control agents interact with the traffic environment every t seconds (i.e the agents sense the prevailing state of the traffic environment using the visual data, based on which they configure traffic signals in real time for t seconds). The smaller the t, the more often the agents will be asked to make a decision about the configuration of traffic signals. In the current work, to ensure greater adaptiveness, we set t to 10s, which implies that at each MDP step, we have a minimum green signal time duration of 10s. After 10s elapses, based on the prevailing state of the traffic, the agents may decide to have the same signal configuration or change it. Real-world minimum/maximum signal time durations dictated by traffic regulation rules, can also be conveniently accommodated by our simulation model. Following are the MDP settings for our signal control agent; including state, action spaces and reward design.

1) State Space: Each actor (i.e local signal control agent) operates *solely* on camera footage to achieve signal control in real time. Actors only perceive the current state of the traffic environment in and around the intersections that they are controlling. In contrast, the centralised critic operates on global state of the traffic (i.e. concatenation of local observations of all actors). For faster computation, we downsize the input images to a compact resolution of 100 x 100, having experimentally verified that this does not impair our agents' decision making.

2) Action Space: At each MDP time-step, each signal control agent selects one of the available phases, to be implemented for a duration of t seconds. Based on the set of admissible vehicle movements (illustrated in Fig. 2), *signal phases* are configured [6]. We define a set of discrete actions A such that each computed action corresponds to each phase. For instance, an action a_1 corresponds to a phase p_1 (i.e. $< a_1 \mapsto p_1 >$). At each MDP time-step, given the current state of the traffic, the signal control agents share the common goal to select the signal phase that best serves the existing traffic demand.

3) Reward Design: To evaluate/optimize the overall efficiency of road networks, both delay and throughput are considered as acceptable metrics. In this paper, we focus on optimizing joint traffic throughput across the network of intersections and subsequently, reducing the average time a vehicle spends in an intersection. To accomplish this task, we define two reward functions: (1) a success reward of +1 for every vehicle passing safely through an intersection; and (2) a penalty of -1 for every vehicle waiting at the start of an intersection.

E. Network Architecture

Fig. 4 illustrates our actor-critic network framework. Given the nature of input data (i.e. vision-based), both our actor and critic networks comprise three convolutional layers (*Conv1*, *Conv2* and *Conv3*). Along with the convolutional layers, our critic network includes a linear layer (*Linear4*). In contrast, for our actor network, we use *long-short term memory* (*LSTM*) as the last layer to memorize a short history. Traffic flows form a complex spatial-temporal structure, resulting in non-stationary MDP if the agents do not have access to any previous data to rely on. *LSTM* networks provide an implicit memory that improves performance in partially-observable environments. As seen in Fig. 4, the actor network takes an RGB image as input (depicting the current traffic state of a signal control agent) and produces action probabilities as output (from which an action deciding the signal phase is sampled). The critic network takes an RGB image as input (depicting the current traffic states of all the participating signal control agent) and produces state values as output.

F. Network Parameter Sharing

Agents may learn successful policies more efficiently using parameter sharing, as it allows learning simultaneously, based on all agents' experiences. Furthermore, parameter sharing enables large-scale application of the proposed multi-intersection optimization approach, as it is infeasible to have a separate actor and critic network for each intersection in a multi-intersection scenario. In this paper, to improve learning efficiency and economise on training time, the agents are allowed to share parameters among each other, i.e. we implement one actor network and one critic network, which are shared by all the agents (illustrated in Fig. 4). However, the agents still demonstrate their respective independent behaviors, as each agent receives different observations based on the prevailing traffic situation in and around the intersection it is controlling.

G. Single Agent Credit Assignment in a Multi-Agent Environment

One of the primary challenges of multi-agent environments is marginalization of each agent's individual contribution towards a global reward. In a recent multi-intersection signal control implementation [17], at each time-step, all signal control agents receive the same global reward (i.e. total aggregated reward through a network of intersections); keeping them oblivious to their true individual contribution towards network-level traffic optimization. In contrast, in the current work, the agents operating under both, our proposed centralised learning method (Sec. IV-H) and baseline methods (Sec. V) are allowed to observe their individual local rewards. From our research perspective, deducing each signal control agent's individual reward is fairly straightforward. Almost all real-world traffic intersections are equipped with induction loops or rely on cameras, which are used to count the vehicles. Since our reward signal includes traffic throughput; thus deducing each signal control agent's independent contribution towards the global network reward is possible.

H. Centralised Signal Control Learning Protocol (our method)

Within urban road networks, following a decentralised framework, any local signal control agent might be susceptible to *myopic* signal control decisions that work effectively locally, but fail to globally optimize traffic on the network level. To avert this possibility, we implement an actor-critic approach such that our critic is centralised, that conditions on the combined observations of all the actors to output a consensual value estimate. While each actor (i.e. each signal control agent) acts independently-based on its private, local observation of the traffic environment without knowing the state of other actors (illustrated in Fig. 4). Our actor network (shared to all actors) represents the policy π, parameterized by θ. Given a team of actors (i.e. signal control agents) consisting of N agents, let $\mathbf{u} = \{u_1, ..., u_N\}$ represents all agents' actions and $\mathbf{o} = \{o_1, ..., o_N\}$ represents all agents' observations. The gradient of the expected return for an agent i, $J(\theta) = \mathbb{E}[R]$ is represented as:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\mathbf{s}\sim\rho, \mathbf{u}\sim\pi(\mathbf{s})}[\nabla_\theta \log \pi_\theta(\mathbf{u}|\mathbf{o}) V^\pi(\mathbf{o})], \quad (3)$$

where, V^π represents a centralised critic network that takes as input the concatenated state information of all participating signal control agents and outputs a centralised state value (i.e it produces a single state-value function after considering the observations of all agents). The policy is updated in the direction of the gradient, illustrated as:

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta), \quad (4)$$

where, α is a step-size parameter.

V. Baselines for Comparison

We compare our multi-intersection signal control strategy with both RL-based and conventional signal control-based methods.

A. Fully-Decentralized Signal Control using Augmented State and Local Rewards [17]

In contrast to our centralised signal control methodology (outlined in Sec. IV-H), here we implement an actor-critic based completely decentralized protocol for traffic optimization through a network of intersections. Signal control agents communicate with one another in the absence of any central controller. At each MDP time-step, each signal control agent independently executes an action based on its local information and the information shared by its neighbors, which increases observability of each local agent. This approach, using information sharing, aims at diffusing local state observations of agents across the network of intersections.

B. Fully-Independent Signal Control using Local State and Local Rewards [9]

A straightforward method to implement actor-critic DRL for autonomous signal control is to have each signal control agent control its individual intersection by independently learning its own policy and the corresponding state-value function. Learning is independent in this setup, without any central controller or interaction between local agents. At each MDP time-step, both actor and critic networks operate on same local observations.

C. Loop-Induced Signal Control (no learning involved) [6]

Lastly, we compare our research findings against the standard induction loop-based adaptive signal control [6]. In loop-induced adaptive signal control, a loop detects approaching vehicles along each incoming lane, within 50m to the junction, that are idling overhead and an electronic impulse is sent to the signal circuit - to switch the red light to green.

All our baseline methods use same configuration of signal phases, outlined in Sec. IV-C.

VI. Evaluation Metrics

We define the following performance metrics used to evaluate our research findings;

A. Traffic Throughput

At each MDP time-step, traffic throughput gives the aggregate number of vehicles that manage to pass through the network of intersections. Higher throughput corresponds to a larger number of vehicles passing through the intersections; indicating a superior multi-intersection signal control method.

B. Journey Travel-Time

At each MDP time-step, journey travel-time is defined as the time interval between vehicles arriving at an intersection stop-line and reaching the end of the intersection. Lower journey travel-times indicates a better multi-intersection signal control method.

VII. Experiments and Results

We simulated a realistic multi-intersection traffic environment with time-variant traffic flows. A view of the traffic environment used for experimentation is illustrated in Fig. 1. To our knowledge, this is the first study considering optimization of traffic flows through multiple intersections based on visual traffic data. Given the same number of signal control agents, in this section we empirically investigate the performance of our multi-intersection signal control strategy (outlined in Sec. IV-H) in contrast to various relevant baseline methods (outlined in Sec. V). The evaluation metrics used in all experiments are outlined in Sec. VI.

A. Centralised signal Control (our method)

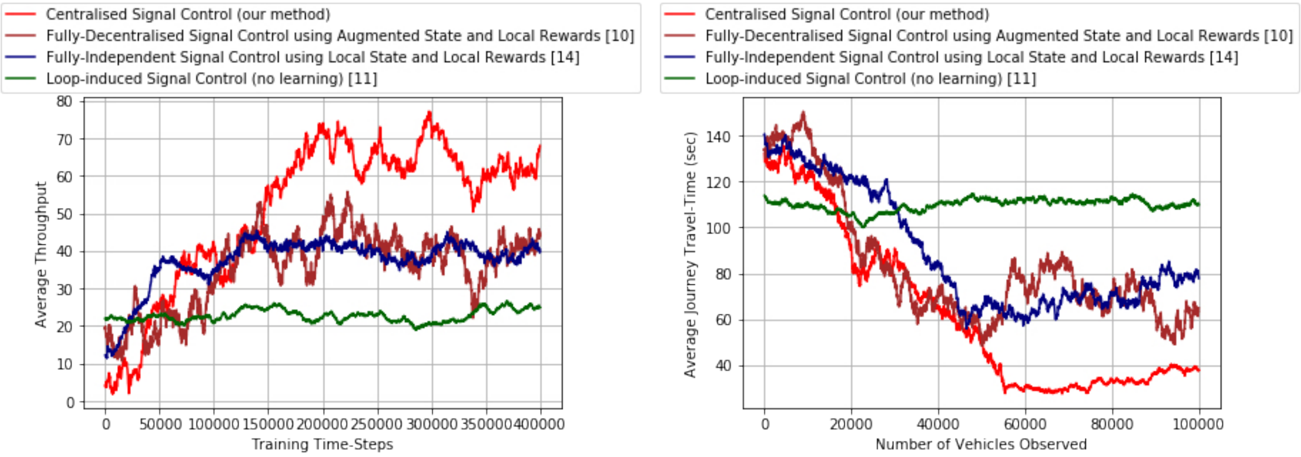
Following the framework illustrated in Fig. 4 and discussed in Sec. IV-H, we implement centralised learning of decentralised policies. At each MDP time-step, our centralised critic network acts on the global traffic state i.e. concatenation of local states of all signal control agents. In contrast, every actor (i.e. every signal control agent) acts on its individual local observation of the prevailing traffic state of the intersection under consideration. As seen in Fig. 5(a) and Fig. 5(b), average traffic throughput (red line) is highest and average junction travel-time (red line) is lowest using our centralised signal control strategy. Centralised critic acting on the global traffic state observation; is able to perceive the overall state of the traffic environment at the network level (i.e. around the network of intersections). With respect to the value function (outputted by critic network), agents efficiently determine the jointly optimal actions (i.e. signal control agents harmonize to maximize the total return). Furthermore, since the centralised critic is aware of the traffic distribution at the network level, this mitigates the known non-stationarity problem of multi-agent environments. Our results signify that multiple agents operating in the same environment do not always require to have an explicit communication amongst themselves to learn coordinated behaviors. This implies that collaboration/cooperation is still possible without information sharing among the agents.

B. Fully-Decentralized Signal Control using Augmented State and Local Rewards

In this setup, based on the method discussed in Sec. V-A, we implement a fully-decentralised learning of agent (actor) policies. Each local agent has access to an augmented state information including regional traffic distribution and cooperative strategy, i.e. observations and fingerprints (current policy) of neighboring agents. At each MDP time-step, both actor and critic networks receive as inputs, this augmented state information. As seen in Fig. 5(a) and Fig. 5(b), both average traffic throughput (brown line) and average junction travel time (brown line) get worse in decentralised learning scenario. This indicates that having access to information from neighboring agents is not always beneficial, it can interfere in learning. Furthermore, agents having access to their individual rewards in a decentralised multi-agent environment can become greedy and tend not to sacrifice for the greater good. Therefore, having a centralised controller with a wider picture of environment, may be useful. In general, finding a globally optimal solution for multiple agents operating with partial information of their environment without any central overseer, is considered intractable [21].

C. Fully-Independent Signal Control using Local State and Local Rewards

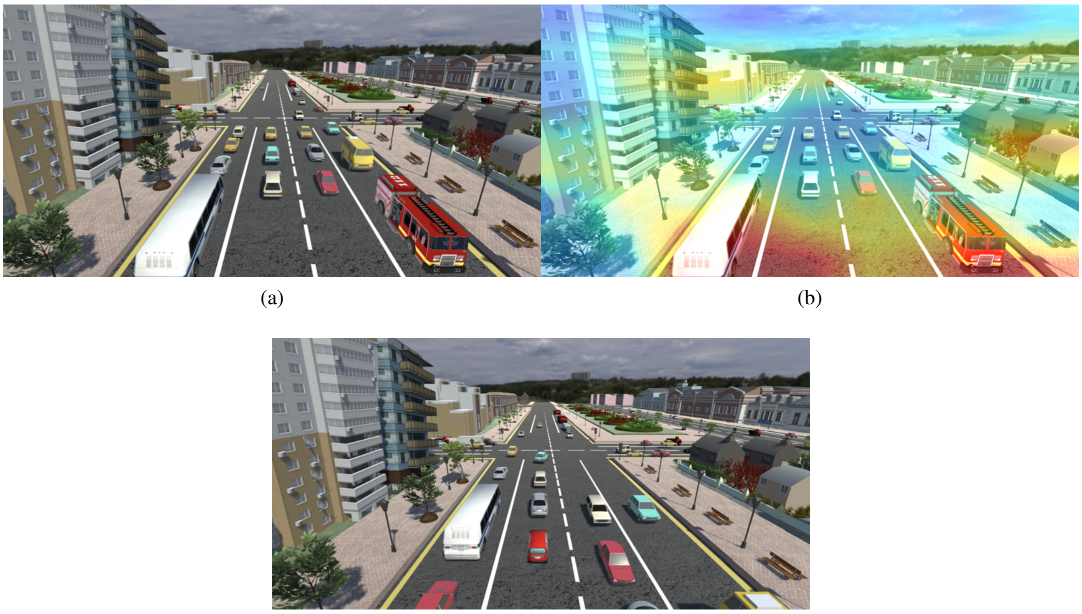
In this setup, based on the method discussed in Sec. V-B, the learning is completely independent without any central controller or any communication between the agents. Agents only operate on their local observations. At each MDP time-step, both actor and critic networks are fed with same local observations of the traffic environment. As seen in Fig. 5(a) and Fig. 5(b), independent learning results in both average traffic throughput (blue line) and average junction travel time (blue line) getting as worse as in the case of decentralised learning. This indicates that agents learning independently-based on their local field-of-view, are susceptible to myopic decisions which leads to overall inferior performance of the signal network.



(a)

(b)

Fig. 5: Graphs demonstrating our centralised signal control method vs fully-decentralised, fully-independent and loop-induced signal control). (a) Average Throughput. (b) Average Journey Travel Time.



(a)

(b)

(c)

Fig. 6: Visualization results. (a) Original Image (b) Grad-cam Activation-Firetruck and Public Bus (c) Next Signal Phase Activated (bus given priority to pass through the intersection).

Loop-induced signal control (green line) performs the worst in all cases, as this method fails to; (1) extensively view the traffic environment due to induction loops' operational narrow range, and (2) continuously modify agents' traffic optimization decisions-based on the dynamically changing traffic flow patterns, as there is no learning involved. Due to brevity, we could not share all our research findings. In a more expanded version of paper, we will include results on effectiveness of the proposed multi-intersection signal control approach around different network topologies, varying weather and lighting conditions.

D. Signal Control Decisions' Interpretability

With the help of deep learning, DRL applied to visual traffic data from road intersections eliminates the need to have pre-engineered features describing the traffic environment. However, deep learning models; Deep Neural Networks (DNNs) are known to be black boxes and visualizing these models can help in understanding the high-level behavior of our signal control agent and discover the parts of the visual input that influenced a certain signal control decision. Greydanus et al. [22] explored the utility of visual stimuli in making decisions in the Atari domain using saliency maps.

To the best of our knowledge, DNN models used for signal control optimization have not been previously visualized. In the current work, we take a step towards explainable Artificial Intelligence and illustrate the interpretability of our signal control agent's decisions. We implement Grad-CAM [23] to produce heatmap on top of the input image to depict the critical area that dominates a certain signal control decision. Grad-CAM facilitates the visual explanations of convolutional neural networks (CNNs) using gradient-based localization. This approach uses the gradients flowing into the final convolutional layer to generate a coarse localization, highlighting the important areas in the visual input leading to a certain network output. Grad-CAM has been previously applied to produce visual explanations for the tasks such as image classification, image captioning and visual question answering. To our knowledge, in this paper, Grad-CAM is applied for the first time to produce visual explanations for a reinforcement learning and a traffic optimization task.

Our visualization results signify that our signal control method, apart from prioritizing the swift movement of traffic based on the prevailing traffic demand captured by wide-range cameras (for brevity we could not include these results here), it also attends to different types of vehicles. As seen in Fig. 6(b), our DRL-based signal control method shows activation around emergency and public vehicles to prioritizes their movement through the intersections. Our visualization research findings reflect that DRL applied to visual traffic data enables signal control based on key traffic features (such as vehicle type and their relevance) that would otherwise be infeasible to explore or exploit using popular traffic data collection methods (such as induction loops). It also validates the benefit of using visual traffic data in providing more flexibility (for e.g. larger detection areas) compared to typically used induction loops with narrow operational range. Furthermore, visualizing the output of DNN algorithms has the potential of breaking barriers in machine learning research. We expect that this DNN visualization will help transportation engineers gain further trust in applying deep learning paradigms to autonomous transportation.

VIII. Conclusion

This paper presents the first application extending deep reinforcement learning methods to optimize traffic through multiple intersections based *solely* on visual traffic data, without hand-crafted traffic state features. We demonstrate a centralised controller that is able to bring about a principled learning strategy between the signal control agents, resulting in positive emergence of cooperative behavior among them in a scenario where each agent has access only to the partial state of the traffic environment. In future work, we plan to introduce algorithms addressing the limitations arising from centralisation: while it results in effective traffic optimization in multi-intersection scenarios, as the number of agents/intersections increases, the centralised critic's state input dimensionality increases exponentially and is susceptible to single-point-of-failure.

References

[1] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," 2011.

[2] T. L. Thorpe and C. W. Anderson, "Traffic light control using sarsa with three state representations," Citeseer, Tech. Rep., 1996.

[3] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[4] D. Garg, M. Chli, and G. Vogiatzis, "Deep reinforcement learning for autonomous traffic light control," in *2018 3rd IEEE International Conference on Intelligent Transportation Engineering (ICITE)*. IEEE, 2018, pp. 214–218.

[5] ——, "A deep reinforcement learning agent for traffic intersection control optimization," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 4222–4229.

[6] P. Koonce and L. Rodegerdts, "Traffic signal timing manual." United States. Federal Highway Administration, Tech. Rep., 2008.

[7] L. Kraemer and B. Banerjee, "Multi-agent reinforcement learning as a rehearsal for decentralized planning," *Neurocomputing*, vol. 190, pp. 82–94, 2016.

[8] J. N. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, "Counterfactual multi-agent policy gradients," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[9] V. R. Konda and J. N. Tsitsiklis, "Actor-critic algorithms," in *Advances in neural information processing systems*, 2000, pp. 1008–1014.

[10] J. Gao, Y. Shen, J. Liu, M. Ito, and N. Shiratori, "Adaptive traffic signal control: Deep reinforcement learning algorithm with experience replay and target network," *arXiv preprint arXiv:1705.02755*, 2017.

[11] X. Liang, X. Du, G. Wang, and Z. Han, "Deep reinforcement learning for traffic light control in vehicular networks," *arXiv preprint arXiv:1803.11115*, 2018.

[12] M. Wiering, "Multi-agent reinforcement learning for traffic light control," in *Machine Learning: Proceedings of the Seventeenth International Conference (ICML'2000)*, 2000, pp. 1151–1158.

[13] T. Chu, S. Qu, and J. Wang, "Large-scale traffic grid signal control with regional reinforcement learning," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 815–820.

[14] H. A. Aziz, F. Zhu, and S. V. Ukkusuri, "Learning-based traffic signal control algorithms with neighborhood information sharing: An application for sustainable mobility," *Journal of Intelligent Transportation Systems*, vol. 22, no. 1, pp. 40–52, 2018.

[15] S. El-Tantawy, B. Abdulhai, and H. Abdelgawad, "Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (marlin-atsc): methodology and large-scale application on downtown toronto," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1140–1150, 2013.

[16] T. Chu and J. Wang, "Traffic signal control by distributed reinforcement learning with min-sum communication," in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 5095–5100.

[17] T. Chu, J. Wang, L. Codecà, and Z. Li, "Multi-agent deep reinforcement learning for large-scale traffic signal control," *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[18] D. Garg, M. Chli, and G. Vogiatzis, "Traffic3d: A new traffic simulation paradigm," in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2019, pp. 2354–2356.

[19] ——, "Traffic3d: A rich 3d-traffic environment to train intelligent agents," in *International Conference on Computational Science*. Springer, 2019, pp. 749–755.

[20] S. Blackman, *Beginning 3D Game Development with Unity 4: All-in-one, multi-platform game development*. Apress, 2013.

[21] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein, "The complexity of decentralized control of markov decision processes," *Mathematics of operations research*, vol. 27, no. 4, pp. 819–840, 2002.

[22] S. Greydanus, A. Koul, J. Dodge, and A. Fern, "Visualizing and understanding atari agents," *arXiv preprint arXiv:1711.00138*, 2017.

[23] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 618–626.