Learning the max pressure control for urban traffic networks considering the phase switching loss

Xingmin Wang[a], Yafeng Yin[a,b], Yiheng Feng[c] and Henry X. Liu[a,d,*]

[a]Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, Michigan, United States
[b]Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan, United States
[d]University of Michigan Transportation Research Institute, Ann Arbor, Michigan, United States
[c]Lyles School of Civil Engineering, Purdue University, West Lafayette, Indiana, United States

ARTICLE INFO

Keywords:
Traffic Signal Control
Max Pressure Control
Traffic Networks
Switching Loss
Reinforcement Learning
Policy Optimization

ABSTRACT

Previous studies have shown that the max pressure control is a throughput-optimal policy that can stabilize the store-and-forward traffic network when the demand is within the network capacity. However, most of the existing studies do not consider the loss of capacity associated with phase switching, which will undermine the stability of the network. This work proposes a novel framework that utilizes reinforcement learning algorithms to optimize a max pressure controller considering the phase switching loss. We first modify the max pressure control by introducing a switching curve and prove that the proposed control method is throughput-optimal in a store-and-forward network. Then the theoretical control policy is extended by using a distributed approximation and position-weighted pressure so that the policy-gradient reinforcement learning algorithms can be utilize to optimize the parameters in the policy network including the switching curve and the weight curve. Simulation results show that the proposed control method greatly outperforms both the conventional max pressure control and vehicle-actuated control. The proposed framework combines the strengths of the data-driven method and the theoretical control model by utilizing reinforcement learning algorithm to optimize the max pressure controller, which is of great significance for real-world implementations because the proposed control policy can be generated in a distributed fashion based on local observations.

1. Introduction

Traffic signal control and optimization methods have been an active research topic for the past decades and recent literature can be roughly divided into three different categories: 1) optimization or optimal control methods based on different traffic models and formulations, 2) artificial intelligence algorithms such as the reinforcement learning, and 3) max pressure control for a general signalized network. Most optimization or optimal control methods are based on a receding-horizon optimization established on traffic flow models such as the cell transmission model or the variational formulation (Lo, 2001; Aboudolas et al., 2009; Wada et al., 2017). However, such receding-horizon optimization methods usually suffer from heavy computational cost especially when dealing with large-scale networks as well as long planning horizon; hence it is usually challenging to deploy these methods in the real world.

Reinforcement learning (RL) algorithms have also been extensively used for traffic signal control optimization during the past decade (Arel et al., 2010; Khamis and Gomaa, 2014; Yau et al., 2017; Chu et al., 2019; Wei et al., 2019b). By training offline, RL can directly learn an end-to-end control policy from the observation by interacting with the simulation environment. Most of the existing literature using RL for traffic signal control focused on the design of the input state space and reward (Wei et al., 2019b), while utilizing different RL techniques such as the multi-agent algorithms (Chu et al., 2019). However, the control policy obtained by RL is usually expressed by a neural network. Due to the issue of the generalization ability of the neural networks, it would not be preferable to directly apply RL policy learned offline in a simulation environment to the real world without additional adjustments.

The max pressure control, which is also known as the back pressure or max weight control, is originally studied in the communication network domain with respect to routing and scheduling (Tassiulas and Ephremides, 1990; Neely, 2010; Srikant and Ying, 2013). It was firstly introduced to traffic network signal control by Varaiya (2013), and followed

*Corresponding author
xingminw@umich.edu (X. Wang); yafeng@umich.edu (Y. Yin); feng333@purdue.edu (Y. Feng); henryliu@umich.edu (H.X. Liu)
ORCID(s): 0000-0003-0435-2786 (X. Wang); 0000-0003-3117-5463 (Y. Yin); 0000-0001-5656-3222 (Y. Feng); 0000-0002-3685-9920 (H.X. Liu)

by various extensions and evaluations (Le et al., 2015; Xiao et al., 2014; Zaidi et al., 2016; Sun and Yin, 2018; Li and Jabari, 2019; Chen et al., 2020). The max pressure control for urban traffic networks has drawn tremendous attention in recent years since it can provide appealing theoretical guarantee of stabilizing the store-and-forward network as long as the demand is within the network capacity. Besides, it is a decentralized control policy in which each intersection makes its own decision based on the upstream and downstream queue lengths.

However, the max pressure control introduced in most literature (Varaiya, 2013; Le et al., 2015; Xiao et al., 2014; Zaidi et al., 2016) is derived based on a store-and-forward network model, which contains some strong assumptions such as infinite link capacity, no link travel time, and no switching loss. In particular, it is well known that, due to the phase switching loss, the phase switching frequency should decrease with the increase of the traffic demand so that the network queue lengths can be stabilized under higher traffic volume without suffering much phase switching loss. Nonetheless, the conventional max pressure control fails to adjust its phase switching frequency dynamically according to the varying traffic demand; and hence it is no longer throughput-optimal over a network model with phase switching loss (Celik et al., 2016).

In this work, we propose to utilize the policy-gradient reinforcement learning methods to learn a max pressure control policy that considers the phase switching loss. We first propose an extended max pressure control policy named SCMP, short for Switching-Curve-based Max Pressure control. It can be proved that, under the network model with the phase switching loss, SCMP is *throughput-optimal*, meaning that it can stabilize the network queue lengths as long as the traffic demand is (strictly) within the network capacity. SCMP extends the original max pressure control by introducing a switching curve that could help the controller dynamically adjust the phase switching frequency according to the current traffic loads. To adapt to the real-world traffic which is much more complicated than the store-and-forward point-queue model, we further modify SCMP by using a distributed approximation and the position-weighted pressure scheme. This modified max pressure control, which will be referred as ESCMP (Extended-SCMP), is a more practical and general version of SCMP with the variant weight curves and the switching curves. While the switching curve determines the switching behavior of the controller, the weight curve enables the controller to consider the vehicles at different locations differently, so that it could implicitly improve the coordination among intersections.

Furthermore, we utilize the policy-gradient RL algorithms to optimize the two parametric curves in ESCMP including the switching curve and the weight curve. ESCMP where the parametric curves are optimized by policy-gradient RL algorithms is named as LESCMP (Learned-ESCMP). Compared with other RL-based methods utilizing neural networks to represent the actor, LESCMP uses the max pressure control policy network, which is interpretable and derived based on a control policy that has certain theoretical guarantee over a simplified network model. One seemingly similar method to LESCMP that combines the RL and max pressure control comes from Wei et al. (2019a), which integrated the "pressure" into the reward function. However, the difference is quite obvious: we directly utilize the max pressure controller as the actor instead of setting the pressure as the reward. As a distributed control policy in which each intersection makes its own decision solely based on its upstream and downstream observation, LESCMP would be also of great significance for the real-world implementation, especially in dealing with large-scale traffic networks.

The rest of this paper is organized as follows. Section 2 describes the network model used in this paper that adds the phase switching loss to the original store-and-forward model. Sections 3 introduces the proposed max pressure controllers, from SCMP, ESCMP, to LESCMP. Section 4 proves that the proposed SCMP is throughput-optimal under the network model with switching loss. Section 5 shows the simulation results while Section 6 concludes this paper.

2. Network model with switching loss

In the theoretical analysis of the proposed max pressure control methods, we extend the store-and-forward model used by prior studies (Aboudolas et al., 2009; Varaiya, 2013) to further capture the phase switching loss. Figure 1 is an illustration of the store-and-forward network model. Let $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ be a general traffic network where $\mathcal{N}$ is the set of the nodes (intersections) and $\mathcal{L}$ represents all the links. Usually, a link contains three movements: through, left-turn, and right-turn. It is assumed that different movements of the same link are separate and do not block each other. A movement can be defined as a tuple composed of the origin link and the destination link $\mathcal{M} = \mathcal{L} \times \mathcal{L}$. We further divide the movements into two categories: ordinary movements $\mathcal{M}^o$ and exit movements $\mathcal{M}^e$. The exit movements will not be considered in the analysis since the vehicle of the exit movements can be freely discharged without additional downstream constraints. For each movement $ij \in \mathcal{M}^o$, let $x_{ij}(t)$ be the queue lengths at time slot t and c_{ij} be the saturation flow rate, which is treated as a constant.

For the traffic demand and signal constraints, let $a_{ij}(t)$ be the exogenous demand of movement $ij \in \mathcal{M}$, which is

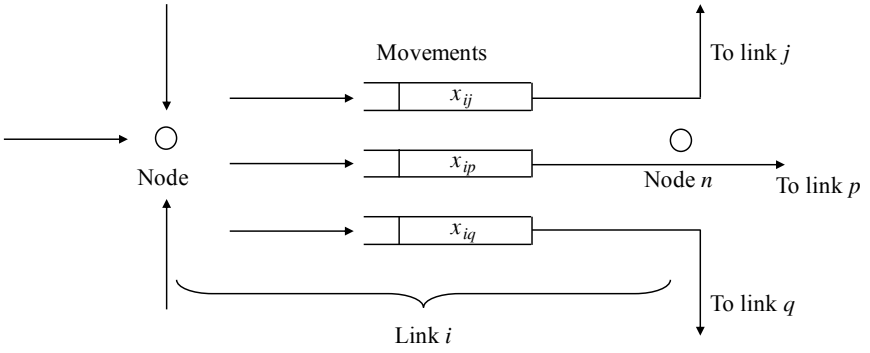

Figure 1: Store-and-forward model, reproduced from Varaiya (2013)

assumed to be i.i.d. with the expectation $\mathbb{E}a_{ij}(t) = a_{ij}$ and the maximum value $a_{\max}$ ($0 \le a_{ij} \le a_{\max}$). Let $r_{ij}(t)$ be the turning ratio from link i to link j, which is also i.i.d. with the expectation $\mathbb{E}r_{ij}(t) = r_{ij}$. For the traffic signal timing plan, we define $s_{ij}(t) \in \{0, 1\}$ as the traffic signal state for the movement ij where 0 corresponds to the red light and 1 represents the green light. Generally, the signal constraints can be formulated as a linear constraint:

$$\boldsymbol{s}(t) \in \{\boldsymbol{s} \mid \boldsymbol{K} \cdot \boldsymbol{s} \le \boldsymbol{h}\} = \mathcal{S}, \tag{1}$$

where $\boldsymbol{s}$ is the column vector that represents the traffic signal state for each movement. $\boldsymbol{K}$ is a matrix and $\boldsymbol{h}$ is a column vector with proper dimensions. For example, if there is only an isolated intersection with two conflict through movements, then the signal constraint can be written as $s_1 + s_2 \le 1$, which can be expressed by Equation (1) with $\boldsymbol{s} = [s_1, s_2]^T$, $\boldsymbol{K} = [1, 1]$, and $h = 1$. It is easy to verify that the set $\mathcal{S}$ is a polyhedron with integer-valued vertices.

Figure 2 shows the flowchart of modeling the switching loss. Let $\lambda_{ij}(t) \in \{0, 1\}$ be the indicator; $\lambda_{ij}(t) = 1$ means that the movement ij is in the discharge mode at time t while $\lambda_{ij}(t) = 0$ corresponds to the switching mode. Let $\chi_{ij}(t)$ be the count down timer that stores the remaining duration of the switching mode and we have

$$\lambda_{ij}(t) = \begin{cases} 1 & \chi_{ij}(t) = 0 \\ 0 & \chi_{ij}(t) > 0 \end{cases}. \tag{2}$$

Let $\mathcal{M}_n^o$ be the set of ordinary movements that enter the node n; whenever the signalized node n switches to the switching mode from the discharge mode, we have:

$$\chi_{ij}(t) = T^r \quad \forall (i, j) \in \mathcal{M}_n^o \tag{3}$$

which means that all the movements entering the node i will switch to the switching mode. T^r is the total number of time slots for the switching loss. For each time slot, the count down timer $\chi_{ij}(t+1)$ is updated as:

$$\chi_{ij}(t+1) = \begin{cases} \chi_{ij}(t) - 1 & \lambda_{ij}(t) = 0 \\ \chi_{ij}(t) & \lambda_{ij}(t) = 1 \end{cases}. \tag{4}$$

Combining the switching loss model given by Equations (2-4) and the store-and-forward model (Aboudolas et al., 2009; Varaiya, 2013) yields the following dynamics of the queue lengths:

$$\begin{aligned} x_{ij}(t+1) = x_{ij}(t) + a_{ij}(t) + \sum_k r_{ij}(t) \min\{x_{ki}(t), c_{ki} s_{ki}(t)\} \cdot \lambda_{ij}(t) \\ - \min\{x_{ij}(t), c_{ij} s_{ij}(t)\} \cdot \lambda_{ij}(t) \qquad \forall (i, j) \in \mathcal{M}^o \end{aligned} \tag{5}$$

which is equivalent to the matrix form:

$$\boldsymbol{x}(t+1) = \boldsymbol{x}(t) + \boldsymbol{a}(t) - (\boldsymbol{I} - \boldsymbol{R}(t)) \cdot \boldsymbol{\Lambda}(t) \cdot \min\{\boldsymbol{x}(t), \boldsymbol{C}\boldsymbol{s}(t)\}, \tag{6}$$

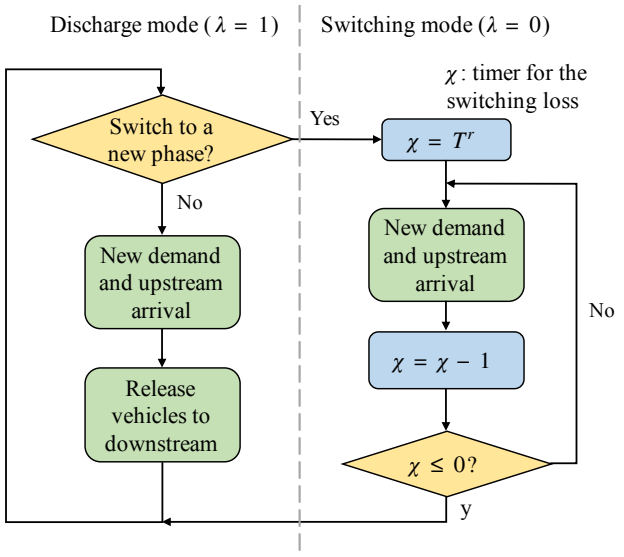

Figure 2: Flowchart of network dynamics considering the switching cost.

where $\boldsymbol{x}(t)$ is the queue lengths of all the ordinary movements. $\min\{\cdot,\cdot\}$ is the entry-wise minimization of the two vectors. $\boldsymbol{C}$ is a diagonal matrix with $\boldsymbol{C}_{mm} = c_m, \forall m = (i,j) \in \mathcal{M}^o$. $\boldsymbol{R}(t)$ is the matrix containing all the turning ratio; $\boldsymbol{I}$ is the identical matrix; $\boldsymbol{\Lambda}(t)$ is a diagonal matrix with $\{\boldsymbol{\Lambda}(t)\}_{mm} = \lambda_{mm}(t), \forall m = (i,j) \in \mathcal{M}^o$. Compared with the dynamics in Varaiya (2013), Equations (5-6) have an extra λ or $\boldsymbol{\Lambda}$ term so that the vehicle will only be allowed to pass the intersection in the discharge mode.

When there is no switching loss, the whole system is a Markov chain with the state representation $\boldsymbol{x}_t$. The system is more complicated with the switching loss; the timer vector $\boldsymbol{\chi}(t)$ as well as the signal state $\boldsymbol{s}_t$ needs to be augmented to the state to maintain the Markovian property. The change of the Markovian property will influence the proof of the stability in Section 4: instead of considering the Lyapunov drift step by step, the Lyapunov drift between switching times will be considered to prove the stability.

3. Proposed max pressure control methods

3.1. Switching-Curve-based Max Pressure control (SCMP)

Based on the modified store-and-forward model with extra consideration of the phase switching loss, SCMP is similar to the switching-curve-based (SCB) method introduced in Celik et al. (2016) but will be extended in the two following aspects: 1) from one intersection (single-hop) to a general network (multi-hop); 2) the weight/pressure will be a more general function of queue lengths. SCMP includes two parts: 1) how to switch, and 2) when to switch. For the first aspect, whenever the switching is activated, the new signal timing plan is chosen as:

$$\boldsymbol{s}^* = \arg\max_{\boldsymbol{s}\in S} \mathrm{pr}\left(\boldsymbol{x}_t, \boldsymbol{s}\right) = \arg\max_{\boldsymbol{s}\in S} \boldsymbol{w}\left(\boldsymbol{x}_t\right)^T \boldsymbol{C}(\boldsymbol{I}-\boldsymbol{R})\boldsymbol{s} \tag{7a}$$

$$= \arg\max_{\boldsymbol{s}\in S} \sum_{n\in\mathcal{N}} \left(\sum_{ij\in\mathcal{M}_n^o} s_{ij}c_{ij} \cdot \left(w_{ij}(x_{ij}) - \sum_{jk\in\mathcal{M}} r_{jk} w_{jk}(x_{jk}) \right) \right) \tag{7b}$$

where $\mathrm{pr}(\boldsymbol{x}, \boldsymbol{s})$ represents the network *pressure* function under the state $\boldsymbol{x}$ and control policy $\boldsymbol{s}$. $\boldsymbol{w}(\cdot)$ is to apply function $w_i(\cdot)$ to each entry of the column vector within it. The $w_i(t)$ is a weight function that satisfies the following conditions:

1. Function $w_i(x)$ is increasing and continuous for $x \geq 0$, and $w_i(0) = 0$;
2. $w(x) \to \infty$ when $x \to \infty$;
3. With bounded constant $0 < B_0 \leq B_1 < \infty$:

$$w_i(x) + B_0\Delta x \leq w_i(x+\Delta x) \leq w_i(x) + B_1\Delta x \tag{8}$$

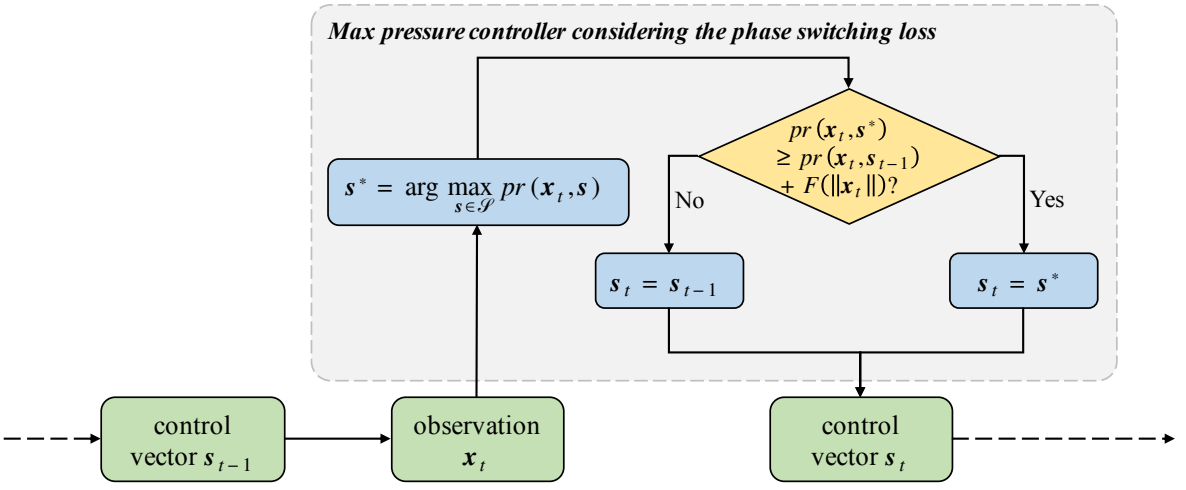

Figure 3: Flowchart of network dynamics considering the switching cost.

For example, $w_i(x)$ can be any sublinear or piece-wise linear functions that monotonically increase with the queue lengths. The max pressure policy given by Equation (7b) is similar to the max pressure control given by Varaiya (2013). The only difference is that we generalize the queue lengths in the pressure to a more general function. Noted that Equation (7a) is essentially a linear program (LP) that will reach the global optimum at an integer-valued vertex. This naturally leads to a discrete signal control policy for each time slot, which suits the case in practice. Another discussion in terms of Equation (7) is that it is a distributed algorithm since both the network pressure function and the signal constraints ($\boldsymbol{s} \in \mathcal{S}$) are separable among intersections. Therefore, finding the control policy of the network according to Equation (7) is equivalent to find the max pressure control policy for each intersection:

$$\boldsymbol{s}_n = \arg\max_{\boldsymbol{s}_n \in \mathcal{S}_n} \sum_{ij \in \mathcal{M}_n^o} s_{ij} c_{ij} \cdot \left(w_{ij}(x_{ij}) - \sum_{jk \in \mathcal{M}} r_{jk} w_{jk}(x_{jk}) \right) \qquad \forall n \in \mathcal{N} \tag{9}$$

where each intersection n can determine its control policy $\boldsymbol{s}_n$ solely based on its upstream observation $\{x_{in} \mid \forall i, in \in \mathcal{M}\}$ and downstream observation $\{x_{nj} \mid \forall j, nj \in \mathcal{M}\}$. Therefore, the max pressure control policy given by Equation (7) or Equation (9) has two major strengths: i) it is a distributed control policy among intersections; ii) it is an end-to-end control policy that directly generates the control policy given the upstream and downstream observation.

For the switching condition, we refer to Celik et al. (2016) and define the switching function as:

$$\begin{aligned} \psi(t) &= \max_{s \in S} \operatorname{pr}\left(\boldsymbol{x}_t, \boldsymbol{s}\right) - \operatorname{pr}\left(\boldsymbol{x}_t, \boldsymbol{s}_{t-1}\right) - F\left(\|\boldsymbol{x}_t\|\right) \\ &= \max_{s \in S} \boldsymbol{w}\left(\boldsymbol{x}_t\right)^T \boldsymbol{C}(\boldsymbol{I} - \boldsymbol{R})\left(\boldsymbol{s} - \boldsymbol{s}_{t-1}\right) - F\left(\|\boldsymbol{x}_t\|\right), \end{aligned} \tag{10}$$

where $\|\cdot\|$ is 1-norm of the column vector that equals the summation of all the queue lengths. $F(\cdot)$, which is defined as the *switching curve* in this paper, is a sublinear function satisfying:

$$\lim_{x \to \infty} F(x) = \infty \qquad \lim_{x \to \infty} \frac{F(x)}{x} = 0. \tag{11}$$

Based on this switching curve function, the switching is only activated when $\psi(t) \geq 0$, that is,

$$\boldsymbol{s}_t = \begin{cases} \boldsymbol{s}_{t-1} & \psi(t) < 0 \\ \boldsymbol{s}^* & \psi(t) \geq 0 \end{cases}. \tag{12}$$

To sum up, Figure 3 shows the flowchart of the overall SCMP controller given by Equation (7-12). For each time slot, the controller first observes the current system state $\boldsymbol{x}_t$ and then finds the timing plan $\boldsymbol{s}^*$ that maximizes

the pressure function. Instead of switching to the new timing plan $\boldsymbol{s}^*$ immediately, it is only activated whenever the maximized pressure $\text{pr}(\boldsymbol{x}_t, \boldsymbol{s}^*)$ exceeds the pressure of the original signal timing plan $\text{pr}(\boldsymbol{x}_t, \boldsymbol{s}_{t-1})$ by a certain value $F(\|\boldsymbol{x}_t\|)$, which is a sublinear function of the total queue lengths.

We will prove later in Section 4 that SCMP is throughput-optimal as long as the weight function and the switching curve satisfy the corresponding conditions. This means that SCMP actually refers to a family of max pressure controllers with variant weight functions and switching curves. Although all the controllers of this family can stabilize the network queue lengths when the demand is within the network capacity; they might lead to different system delay performance. Therefore, in the following subsections, we propose to utilize the RL algorithms to further improve SCMP by optimizing the parameters.

3.2. Practical implementation: ESCMP

Before we go to the control policy optimization using the RL algorithms, we will first modify SCMP to a more practical and general version called Extended-SCMP (ESCMP). Although SCMP is proved to be throughput-optimal over a store-and-forward model with phase switching loss, it might not suit the real-world traffic very well since the store-and-forward model is essentially a simplified point-queue model with some strong assumptions. However, it is usually intractable and much more difficult to provide the theoretical analysis such as the stability based on a more realistic traffic flow model. Therefore, in this paper, we restrict the theoretical analysis to SCMP while modifying SCMP to ESCMP without the proof of the stability. ESCMP modifies and extends SCMP in two aspects: 1) from a centralized switching to an approximated distributed switching; 2) from the weight or pressure defined by the function of the queue lengths to a more general position weighted pressure.

Distributed switching For SCMP, although the selection of $\boldsymbol{s}^*$ given by Equation (7) is distributed among intersections, the switching rule given by Equation (11-12) requires the switching time to be determined in a centralized fashion. The main reason for choosing a centralized switching rule is to simplify the proof of the global stability. If each intersection decides its own switching time, it would be difficult to analyze the Lyapunov drift of intersections with different switching times. In Hsieh et al. (2017), a superframe is pre-determined by collecting the queue lengths of all the intersections and then individual intersections are allowed to switch more frequently within the superframe. However, it would be better if the switching rule is decentralized which means that each intersection can decide to switch or not only using the local information, making it easier for the real-world implementation. Besides, although the centralized switching is proved to be a stable policy, it has the effect of forcing the intersections with lower traffic volumes to switch less frequently to be synchronized with those congested intersections. This might increase the delay of the low volume intersections.

Therefore, ESCMP uses an approximated distributed switching to replace the centralized switching rule; each signalized node decides to switch whenever the function $\psi^n(\cdot)$ defined below is greater than zero:

$$\psi^n(t) = \max_{\boldsymbol{s}^n \in S^n} \text{pr}\left(\boldsymbol{x}_t^n, \boldsymbol{s}^n\right) - \text{pr}\left(\boldsymbol{x}_t^n, \boldsymbol{s}_{t-1}^n\right) - F\left(\|\boldsymbol{x}_t^n\|\right) \qquad \forall n \in \mathcal{N}, \tag{13}$$

where the superscript n refers to the corresponding value of the node n. Specifically, $\boldsymbol{x}^n$ and $\boldsymbol{s}^n$ represent the queue lengths column vector and traffic signal states of all the movements that enter or exit the node n accordingly.

Position weighted pressure Similar to other max pressure controllers proposed before (Varaiya, 2013; Le et al., 2015; Zaidi et al., 2016), SCMP is derived based on a store-and-forward point-queue network model. One of the major limitations of the store-and-forward model is that it does not consider the vehicle distribution along the link, nor the spatial propagation. To address this problem, Li and Jabari (2019) proposed a position weighted back pressure (PWBP) control, which used a position weighted method (a linear weight) to calculate the pressure for each movement. We will use the similar idea to get the pressure for each movement. Figure 4 illustrates the position-weighted pressure as well as different weight curves. The moving vehicle and stopped vehicle might be considered separately, the red lines are weight curves for the stopped vehicle while the blue lines correspond to the moving vehicle. The key idea of the position weighted pressure proposed by Li and Jabari (2019) is that vehicles at different locations of the movement might contribute differently to the movement pressure. Let $\{l_{ij,p}, \forall p\}$ be the set of the locations (represented by the traveled distance from the start of the movement) where there is a vehicle p along the movement ij and $w_{ij}^*(\cdot)$ be the position-based weight function. For the position weighted pressure, the $w(\boldsymbol{x}_{ij})$ in Equation (9) is calculated as:

$$w(\boldsymbol{x}_{ij}) = \sum_p w_{ij}^*(l_{ij,p}) \tag{14}$$

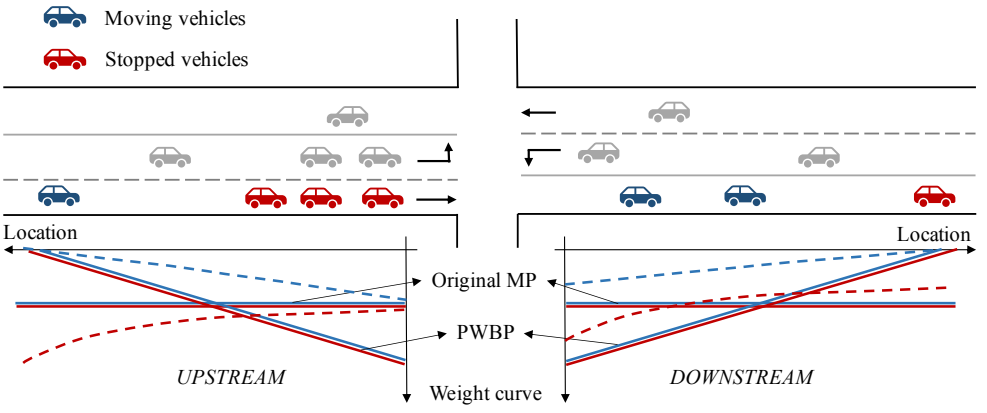

Figure 4: Position-weighted curve to calculate the pressure. Original MP: original max pressure proposed by Varaiya (2013), PWBP: position-weighted back pressure proposed by Li and Jabari (2019). As shown by the dashed lines, this paper regards the position-weighted curves as the parameters to be optimized.

Under this position weighted pressure scheme, Li and Jabari (2019) used a linear curve to calculate the movement pressure as shown in Figure 4, which means that the closer a vehicle to the intersection, the more it contributes to the movement pressure. As a comparison, the original max pressure control proposed by Varaiya (2013) directly used the queue lengths as the pressure for each movement. Under the store-and-forward model, the queue length is essentially the number of vehicles within the movement. Therefore, the vehicles of the same movement exert equal pressure to the whole movement as shown in the horizontal lines in Figure 4. Both the original MP and PWBP do not distinguish the moving vehicle and the stopped vehicle; and hence the blue line and the red line are overlapped. In this paper, as shown by the dashed line, we will treat the weight curve as the parameter to be optimized and split the vehicles to moving vehicles and stopped vehicles. Intuitively, an increasing function would be preferable for moving vehicles. On the contrary, a decreasing function might be better for the stopped vehicles since it can penalize the long queues. Although we will not provide the theoretical analysis of the stability of this position weighted pressure scheme based on a first-order traffic flow model like Li and Jabari (2019), we do use a weight function of the queue lengths to get the pressure under the store-and-forward model, trying to mimic the similar effect.

With these two adaptations, Figure 5 is an illustration to the overall ESCMP controller. As a distributed controller, each intersection has a controller of the same structure as shown in the figure but they would have different parameters due to the different geometry and demand patterns. The road segment is discretized into cells with equal length so that the observation is the number of vehicles within each cell. The weight curve is represented by a vector that has the same dimension as the observation. Then the movement (corresponds to a controlled lane) pressure is obtained by performing an inner product over the observation and the weight curve. A phase is defined as the set of movements that are allowed to pass at the same time, corresponding to a feasible control policy vector $\boldsymbol{s}^n \in \mathcal{S}^n$ of the node n. Then the phase pressure is the summation of the pressure of the movements that are allowed to pass during this phase. Before getting the phase weight layer, the phase pressure should add another switching cost layer, which determines the switching frequency of the controller. At last, the phase with the maximum weight will be chosen as the action of the current time slot.

3.3. Parameter optimization using policy-gradient methods: LESCMP

With the policy network of ESCMP given by Figure 5, we are able to leverage the policy-gradient methods to optimize the parameters including the weight curve and the switching curve. Unlike most the literature that used the neural networks as the actor in the RL algorithms (Chu et al., 2019; Yau et al., 2017; Khamis and Gomaa, 2014; Arel et al., 2010), we use RL to optimize a policy network that has a pre-determined max pressure structure. In this paper, ESCMP controller that is further optimized by RL is named as Learned-ESCMP (LESCMP) in this paper.

RL algorithms can further improve ESCMP in two aspects. Firstly, RL can implicitly take other factors not included in the store-and-forward model into consideration, such as the spill-over and coordination among intersections. Both the spill-over and coordination will influence the system total delay, which relates to the reward of the RL. RL can

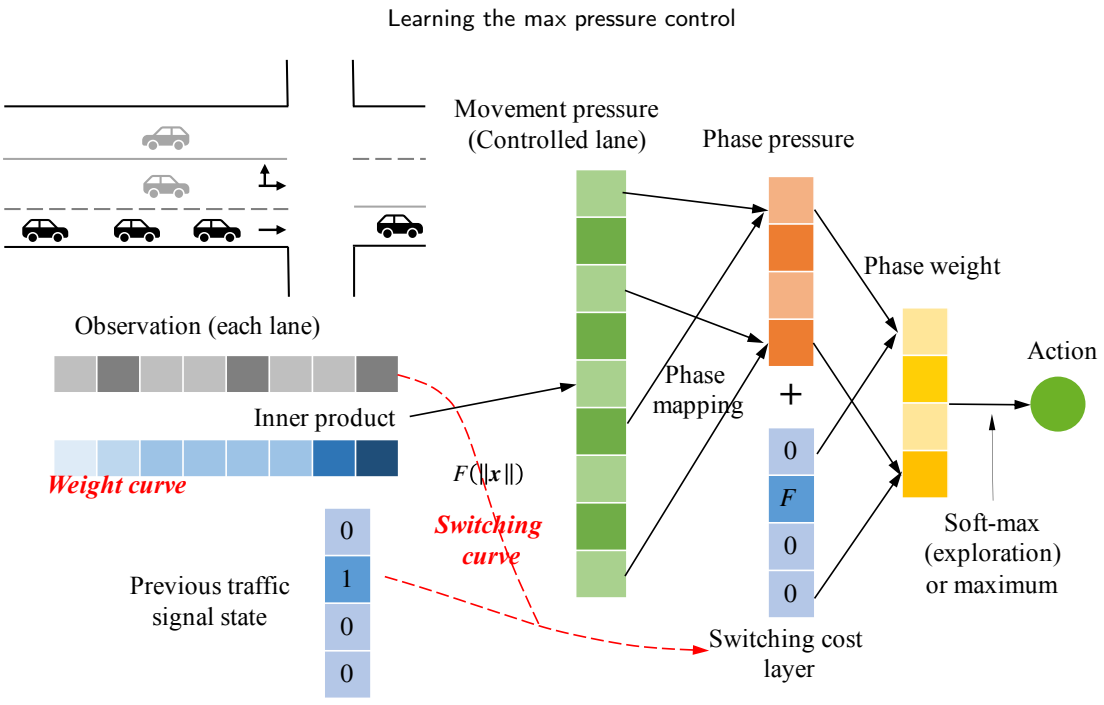

Figure 5: The max pressure control policy network (ESCMP).

take account these factors by adjusting the weight curve and the switching curve when trying to maximize the system total reward. Secondly, the theoretical analysis with regard to the max pressure controller only concerns the stability of the system, which means that the total queue lengths are bounded or the traffic demand can be served in the long run. According to Little's law (Little and Graves, 2008), the bounded total queue lengths guarantee a bounded total delay but not the optimal total delay. Therefore, RL algorithms can be utilize to further optimize the delay performance of the system, which turns out to be hard to deal with in the theoretical analysis.

With the switching loss, the system is still a Markov chain by augmenting the traffic signal state and the count-down timer to the traffic state representation. Let S_t be the augmented system state at time slot t. The weight curves and switching curve in Figure 5 can be parameterized by $\theta \in \Theta$. Since RL requires a stochastic control policy to perform the exploration, the deterministic max pressure control policy can be easily converted to a stochastic version by changing the maximization operator to a softmax (logit model) when selecting the final action as shown in Figure 5. Let $\pi_\theta(\cdot \mid S_t)$ be the probability distribution of the action $a_t \sim \pi_\theta(\cdot \mid S_t)$ that will be taken given state S_t. Let $\tau = [S_0, a_0, S_1, a_1, ...]$ be a realization or a trajectory of the Markov process. To further minimize the delay performance of the system, we can choose the parameter θ as:

$$\theta^* = \arg\min_{\theta\in\Theta} \mathbb{E}_{\tau\sim\pi_\theta} \sum_{t=1}^{T} \|\boldsymbol{x}_t\| = \arg\max_{\theta} \mathbb{E}_{\tau\sim\pi_\theta} R(\tau) = \arg\max_{\theta} J(\theta), \tag{15}$$

where $R(\tau)$ is defined as the reward of the trajectory τ:

$$R(\tau) = -\sum_{t} \|\boldsymbol{x}_t\|, \tag{16}$$

and $J(\theta)$ is the expected reward by taking the expectation with regard to the trajectory τ:

$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta} R(\tau) = -\mathbb{E}_{\tau\sim\pi_\theta} \sum_{t} \|\boldsymbol{x}_t\|. \tag{17}$$

To optimize the parametric control policy in a MDP (Markov Decision Process, aka, controlled Markov chain), in this paper, we adopt the policy-gradient methods (Sutton and Barto, 2018) that update the parameter by using its

gradient:

$$\theta_{k+1} = \theta_k + \alpha \nabla_\theta J(\theta) \mid_{\theta=\theta_k} \tag{18}$$

where α is the learning rate while the gradient is given by:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_t \pi_\theta(a_t \mid S_t) R(\tau)\right], \tag{19}$$

which means that the gradient of the policy can be estimated using the Monte Carlo method. Based on Equation (18-19), there have been different extensions proposed for the policy-gradient methods. For example, trust region policy optimization (TRPO) (Schulman et al., 2015) updates the policy by taking the largest step satisfying a special constraint on the distance between the new and old policies quantified by the KL divergence. To simplify the TRPO which solves a constrained optimization problem for each iteration, the proximal policy optimization (PPO) (Schulman et al., 2017) solves a proximal unconstrained optimization problem for each update. This paper will use the PPO to optimize the parameters in the max pressure policy network shown by Figure 5.

4. Network stability

4.1. Basic concepts for queue lengths strong stability

This section will prove that SCMP given by Equation (7-12) is throughput-optimal under the store-and-forward model with the phase switching loss described in Section 2. Before that, we will introduce some preliminary concepts of the network queue length stability. The strong stability of the global network queue lengths is given by the following definition:

Definition 1. *The network queue lengths are **strongly stable** if:*

$$\lim_{T\to\infty} \sup \frac{1}{T}\mathbb{E}\sum_{t=1}^{T} \|\boldsymbol{x}(t)\| < \infty \tag{20}$$

By definition, the strong stability means that average total queue lengths are bounded in the infinite horizon, which indicates that all the demand will be served in the long run. According to Little' law (Little and Graves, 2008), bounded total queue lengths indicate a bounded system total delay. Therefore, the strong stability can guarantee a bounded total delay but not the optimum. This is one of the reasons that the reinforcement learning is used to optimize the network delay performance as aforementioned in Section 3.3.

In Section 2, we have defined the signal constraints given by Equation (1). The feasible polyhedron of the signal state $\mathcal{S}$ determines the admissible demand region defined below:

Definition 2. *The **admissible demand region** $\mathcal{D}$ is defined as:*

$$\mathcal{D} = \{(\boldsymbol{a}, \boldsymbol{R}) \mid \boldsymbol{a} \preceq (\boldsymbol{I} - \boldsymbol{R})\boldsymbol{C}\boldsymbol{s}, \exists \boldsymbol{s} \in \mathcal{S}\}. \tag{21}$$

The admissible demand region defines the feasible average exogenous demand and turning ratio pair $(\boldsymbol{a}, \boldsymbol{R})$ that can be served by the network. Based on this definition of the admissible demand region, a control policy is called *throughput-optimal* if it can stabilize the network queue lengths as long as the demand belongs to the interior of the admissible demand region $(\boldsymbol{a}, \boldsymbol{R}) \in \text{int}\mathcal{D}$. The following theorem shows that only if the demand is within the admissible demand region, the network queue lengths can be stabilized.

Theorem 1. *The network queue lengths can only be stabilized when $(\boldsymbol{a}, \boldsymbol{R}) \in \mathcal{D}$. With the switching loss, the new admissible demand region has to be in the interior of the original admissible demand region $(\boldsymbol{a}, \boldsymbol{R}) \in int\mathcal{D}$.*

The proof of Theorem 1 can be seen in Appendix A. This theorem indicates that the throughput-optimal control policy can stabilize the network queue length as long as there exists a control policy that can stabilize it.

4.2. Sufficient condition for the queue lengths stability with switching loss

Before we show the stability of SCMP, we will first provide a sufficient condition for the network queue length stability considering switching loss. This sufficient condition is modified from Celik et al. (2016), which is extended in two aspects: 1) from a single-hop to a multi-hop network; 2) from the quadratic Lyapunov function to a more generalized Lyapunov function (Srikant and Ying, 2013). Under the generalized weight function $w(\cdot)$, the Lyapunov function of a given network queue length state $\boldsymbol{x}_t$ is defined as:

$$L(\boldsymbol{x}(t)) = \sum_i \int_{\xi=0}^{x_i(t)} w_i(\xi) d\xi, \tag{22}$$

which becomes the quadratic Lyapunov function used in Varaiya (2013) when the weight function $w_i(\xi) = \xi$.

Let τ_k be the time step when the kth switching is activated and $\boldsymbol{s}^k$ be the signal timing plan chosen after kth switching. Δ_{τ_k} is defined as the Lyapunov drift from the kth switching to $(k+1)$th switching:

$$\Delta_{\tau_k} = \mathbb{E}\left(L(\boldsymbol{x}(\tau_{k+1}) \middle| S_{\tau_k} \right) - L(\boldsymbol{x}(\tau_k). \tag{23}$$

where $S_{\tau_k} = (k, \tau_k, \boldsymbol{x}(\tau_k), \boldsymbol{s}^k)$ is the augmented state including the index of the switching k, the switching time of the kth switching τ_k, the network queue lengths state $\boldsymbol{x}(\tau_k)$, and the traffic signal state $\boldsymbol{s}^k$. It is easy to verify that the dynamic system is a Markov chain under such augmented state representation. The following theorem provides a sufficient condition for the network queue length stability under the switching loss.

Theorem 2. *Under the network model with switching loss described by Section 2. Let k denote the index of the switching. For each k, τ'_k is a random stopping time. Given a compact set C, a sublinear function $F(\cdot)$, and a non-negative function $\delta'(\cdot)$ with $\lim_{x\to\infty} \delta'(x) = 0$. Let ϵ, c_1, c_2 be positive constants. Given a control policy that always selects the max pressure policy according to Equation (7) whenever the switching is activated, if the demand belongs to the interior of the admissible demand region $(\boldsymbol{a}, \boldsymbol{R}) \in intD$ and the following conditions with regard to the switching time are satisfied:*

$$\tau_{k+1} \geq \tau'_{k+1}; \tag{24a}$$

$$\mathbb{E}\left[(\tau'_{k+1} - \tau_k) \middle| S_{\tau_k} \right] \geq c_1(1 - \delta'(\|\boldsymbol{x}(\tau_k)\|))F(\|\boldsymbol{x}(\tau_k)\|); \tag{24b}$$

$$\mathbb{E}\left[(\tau'_{k+1} - \tau_k)^2 \middle| S_{\tau_k} \right] \leq T_r^2 + c_2 \left(F(\|\boldsymbol{x}(\tau_k)\|) \right)^2; \tag{24c}$$

$$\mathbb{E}\left[L(\boldsymbol{x}(t+1)) - L(\boldsymbol{x}(t)) \middle| S_t \right] \leq -\epsilon \|\boldsymbol{w}(\boldsymbol{x}(t))\|, \quad \forall \boldsymbol{x}(t) \in C^o, t \in \left\{ \tau'_{k+1}, \tau'_{k+1}+1, ..., \tau_{k+1} \right\}; \tag{24d}$$

then the network queue lengths will be strongly stable.

The temporal axis illustration given by Figure 6 could help to understand this theorem. The second and the third condition given by (24b-24c) restrict the expectation of the first and second order of the temporal difference $\tau'_{k+1} - \tau_k$. For the first condition given by (24a), if this condition holds as equality, then the last condition will be redundant. If the first condition holds as inequality, which means that the $(k+1)$th switching is activated after τ'_{k+1}, then we need an extra condition given by Equation (24d), requiring a step-by-step negative Lyapunov drift from τ'_{k+1} to τ_{k+1} when the $(k+1)$th switching is activated.

With Jensen's inequality, from Equation (24c) we have:

$$\mathbb{E}[\tau'_{k+1} - \tau_k \mid S_{\tau_k}] \leq T_r + \sqrt{c_2} F(\|\boldsymbol{x}(\tau_k)\|) \tag{25}$$

and thus we have the two-sided bound for the temporal difference between the τ_k and the random stopping time τ'_{k+1} given by Equation (25) and Equation (24b).

The proof of this theorem is based on a lemma that bounds the Lyapunov drift from the switching time τ_k and the random stopping time τ'_{k+1} for each k. Let Δ'_{τ_k} be the conditional Lyapunov drift from τ_k to the random stopping time τ'_{k+1}:

$$\Delta'_{\tau_k} = \mathbb{E}\left(L(\boldsymbol{x}(\tau'_{k+1})) \mid S_{\tau_k} \right) - L(\boldsymbol{x}(\tau_k), \tag{26}$$

then the lemma can be written as:

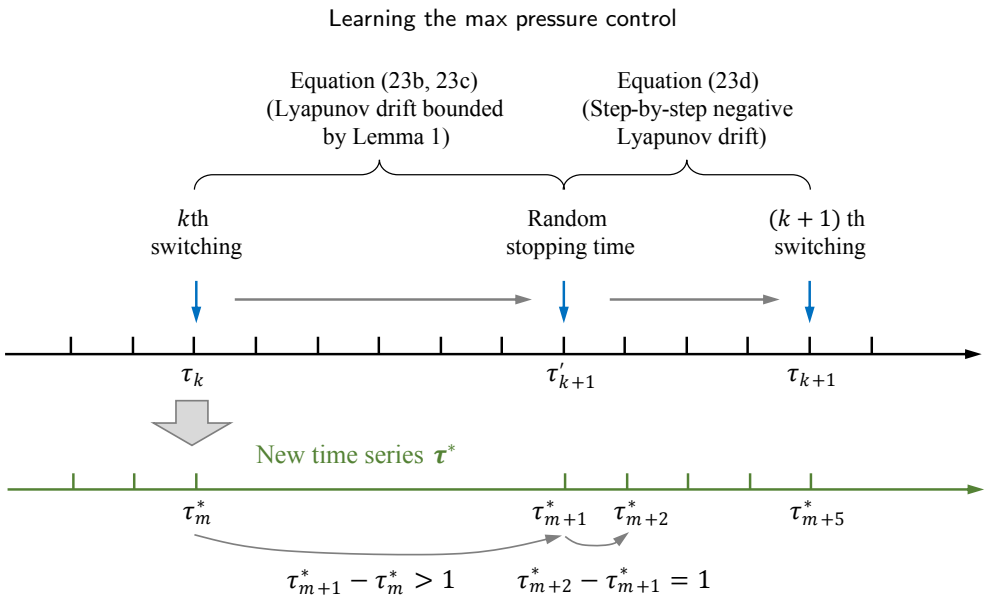

Figure 6: Temporal axis of the switching times.

Lemma 1. *If the max pressure control policy satisfies the conditions given in Theorem 2, then given the demand within the admissible demand region and $\eta > 0, c_3 < \infty$, we have:*

$$\Delta'_{\tau_k} \leq c_3 - \eta F(\|\boldsymbol{x}(\tau_k)\|)\|\boldsymbol{w}(\boldsymbol{x})(\tau_k)\| \tag{27}$$

The proof of Lemma 1 is attached in Appendix B. Lemma 1 essentially provides an upper bound for the Lyapunov drift from the time τ_k to τ'_{k+1} as shown in Figure 6. With Lemma 1, here we give a sketch of the proof of the Theorem 2. As shown on the green axis in Figure 6, let $\boldsymbol{\tau}^*$ be a new time series that skips the time slots when $\tau \in (\tau_k, \tau'_{k+1})$,

$$\boldsymbol{\tau}^* = \left[\tau_1^*, \tau_2^*, \tau_3^*, ...\right]^T = \left[\tau_0, \tau_1', \tau_1' + 1, ..., \tau_1, \tau_2', \tau_2' + 1, ..., \tau_2, ...\right]^T. \tag{28}$$

We will first show that the global queue lengths are strongly stable in this new time series and then complete the proof by extending the results to the whole time series. It is easy to verify that under the new time series, the augmented system state is still a Markov chain given the generalized max pressure policy. Combining the Lemma 1 and the last condition in Theorem 2, we have:

$$\mathbb{E}\left[L(\boldsymbol{x}(\tau_{i+1}^*)) - L(\boldsymbol{x}(\tau_i^*))\,\middle|\,S_{\tau_i^*}\right] \leq c' - \epsilon\|\boldsymbol{w}(\boldsymbol{x}(\tau_i^*))\|, \quad \forall i. \tag{29}$$

This equation comes from Lemma 1 when $\tau_{i+1}^* - \tau_i^* > 1$ and from Equation (24d) when $\tau_{i+1}^* - \tau_i^* = 1$. Taking the expectation for both sides of this equation and summing up all the equations for all i, we will finally get:

$$\frac{1}{T}\sum_{i=1}^{T}\mathbb{E}(\|\boldsymbol{x}(\tau_i^*)\|) < \infty, \quad \forall T \quad \Longrightarrow \quad \lim_{T\to\infty}\sup\sum_{i=1}^{T}\frac{1}{T}\mathbb{E}(\|\boldsymbol{x}(\tau_i^*)\|) < \infty, \tag{30}$$

which means that the network queue lengths under the new time series $\boldsymbol{\tau}^*$ is strongly stable. To prove that the queue lengths are strongly stable of the whole time series, the remaining issue is to prove the queue lengths stability for every skipped period $(\tau_k, \tau'_{k+1}), \forall k$. Here we do not provide the detailed proof, which can be easily derived from Equation (25) and the assumption that the number of arrival for each time slot is bounded. This omitted part is similar to the proof of theorem 1 in Celik et al. (2016).

4.3. Stability of SCMP

With the sufficient conditions of the queue lengths stability given in the previous subsection, this subsection will show that SCMP given by Equation (7-12) is throughput-optimal which means that it could stabilize the network

queue lengths as long as the traffic demand is strictly within the network capacity. Before we prove the stability of the proposed control policy, we will first introduce a lemma which defines a *biased-based* policy (Celik et al., 2016; Hsieh et al., 2017) that can stabilize the network queue lengths by satisfying the sufficient conditions in Theorem 2 with the first condition as an equality.

Lemma 2. *Given a max pressure control that chooses the control policy according to Equation (7), if the switching is activated whenever the the following $\psi'(\cdot)$ function*

$$\psi'(t) = \|\boldsymbol{x}(t) - \boldsymbol{x}(\tau_k)\| - \theta F(\|\boldsymbol{x}(\tau_k)\|), \quad \theta > 0 \tag{31}$$

is greater than zero, the control policy satisfies the conditions in Theorem 2 with the first condition as equality.

We choose not to go through the details of the proof of this Lemma since it is similar to the proof in Celik et al. (2016). Basically, it can be easily derived by using some relaxation tricks to bound first- and second-order moments of the temporal difference between the previous switching time and the time when the $\psi'(\cdot)$ function given by Equation (31) is firstly satisfied. With Lemma 2, the following theorem shows that, when the demand is strictly within the admissible demand region, SCMP given by Equation (7-12) satisfies the condition in Theorem 2, and hence can stabilize the network queue lengths.

Theorem 3. *When the demand is strictly within the admissible demand region, SCMP given by Equation (7-12) satisfies the condition in Theorem 2 with the first condition as inequality, and hence could stabilize the network queue lengths.*

The proof of Theorem 3 is provided in Appendix C. The basic idea of the proof is to first show that before the switching is activated according to the switching rule given by Equation (10), there is always a corresponding biased-based policy given by Equation (31) that the switching is activated in advance. This means that the second and the third condition of Theorem 2 is satisfied by Lemma 2. Then we show that the Lyapunov drift is negative step-by-step after that by using the fact that the condition given by Equation (10) is not satisfied yet. Theorem 3 eventually shows that SCMP is throughput-optimal under the store-and-forward network model with the phase switching loss.

5. Simulation Experiments

We use a simulation model built on SUMO (Krajzewicz et al., 2012) to compare the proposed max pressure control methods including ESCMP and LESCMP with two benchmark methods including the vehicle-actuated control and the original max pressure control . Figure 7 shows the network used in the simulation, which is a corridor with six intersections on Plymouth Road, Ann Arbor, Michigan. The network topology is directly extracted from the OpenStreetMap data set (Haklay and Weber, 2008) while the traffic demand is calibrated from the historical data collected by videos during the evening peak hours. Figure 8 shows the traffic demand pattern used for the reinforcement learning. The duration of each episode is 60 min, which is divided into 5 different periods with variant demand levels or arrival rates. The vehicle arrival follows the Poisson process with a stationary arrival rate within each period. The relative demand level refers to the ratio of the realized demand to the calibrated peak-hour demand value. The green line and area are the mean value and the standard derivation of 10 random samples of the overall vehicle arrival rate of the whole network. For the signal controller, each movement (controlled lane) will has an enforced 3-second yellow time when it is switched from green light to red light and a 2-second all-red clearance time.

In this paper, we use an open-source reinforcement learning library (i.e., Ray rllib) (Liang et al., 2017) for the implementation of LESCMP. The input state includes the traffic state as well as the current signal state; the traffic state is represented by the number of the stopped and running vehicles within each cell of each lane while each lane is divided into cells for every 50 meters. The reward of the environment is chosen as the negative value of the total stop delay while a vehicle is regarded as stopped when its speed is less than a given threshold. The policy network of the PPO is described in Section 3.2 while the switching curve is chosen as

$$F(x) = \alpha \cdot x^{\beta}, \tag{32}$$

where α and β are the parameters. Besides the policy network, the PPO also learns a value network at the same time as the critic to guide the update of the policy network; the value network is simply chosen as a two-layer fully-connected

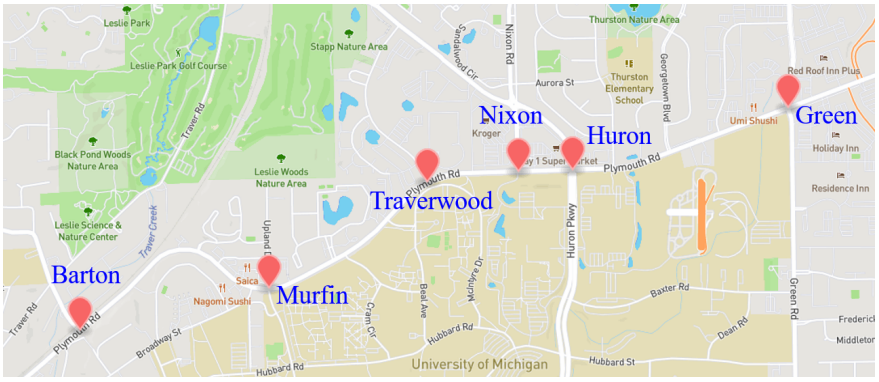

Figure 7: Network topology of Plymouth Rd., Ann Arbor, Michigan.

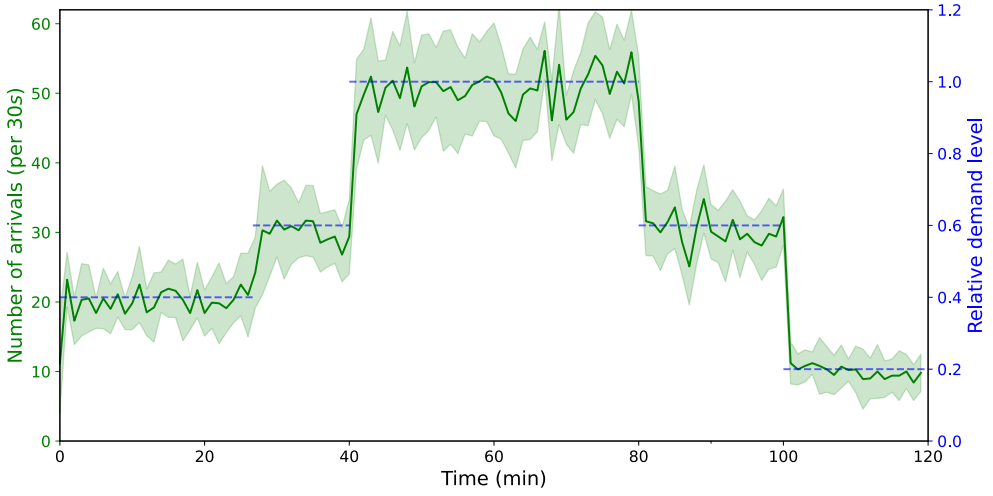

Figure 8: Input demand profile of the simulation environment.

neural network with 256 neurons for each layer. Figure 9 shows the training curve of the PPO with the proposed max pressure policy network. The horizontal axis is the total time steps of the simulation environment while the vertical axis is the average scaled reward for each episode.

Two benchmark controllers are used for the comparison: the actuated control and the max pressure control without consideration of the switching loss. The actuated control tested in this experiment used the default actuated controller provided by the SUMO environment (Krajzewicz et al., 2012). The max pressure control without the consideration of the switching loss is chosen as the position weighted back pressure control (PWBP) proposed by Li and Jabari (2019). For the implementation of PWBP, we slightly change the position-based weight curve in the PWBP by using different curves for the running vehicles and the stopped vehicles as shown in Section 3.2. The weight curve for the running vehicles is chosen as the same as the PWBP while a uniform flat curve is used for the stopped vehicles. The proposed ESCMP tested in this section has the same weight curves with the implementation of PWBP but with an extra switching curve $F(x) = x^{0.4}$, where the parameter 0.4 is selected by a heuristic line search program. To obtain LESCMP, ESCMP is further optimized by using the PPO as aforementioned.

Under the demand profile given by Figure 8, Figure 10 and Table 1 show the comparison of the proposed max pressure methods with the two benchmarks. There are totally four different controllers tested: the actuated control, PWBP, ESCMP, and LESCMP. The evaluation for each controller is repeated 10 times. For each evaluation, the input demand for all the controllers is generated using the same random seed which means that the input demand is

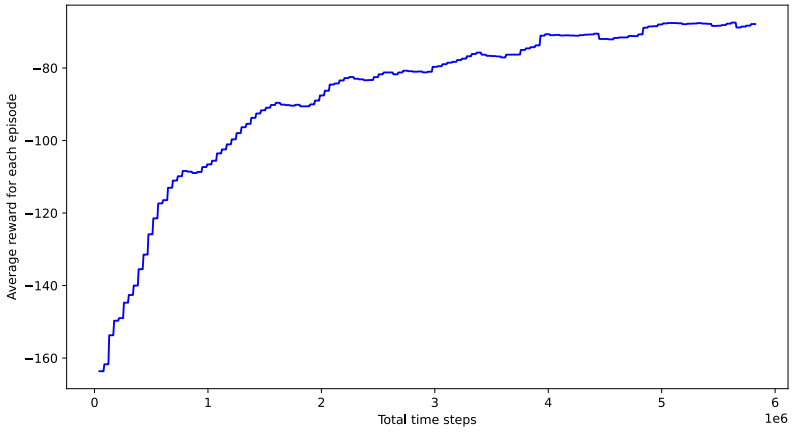

Figure 9: Training curve of the RL using the max pressure policy network.

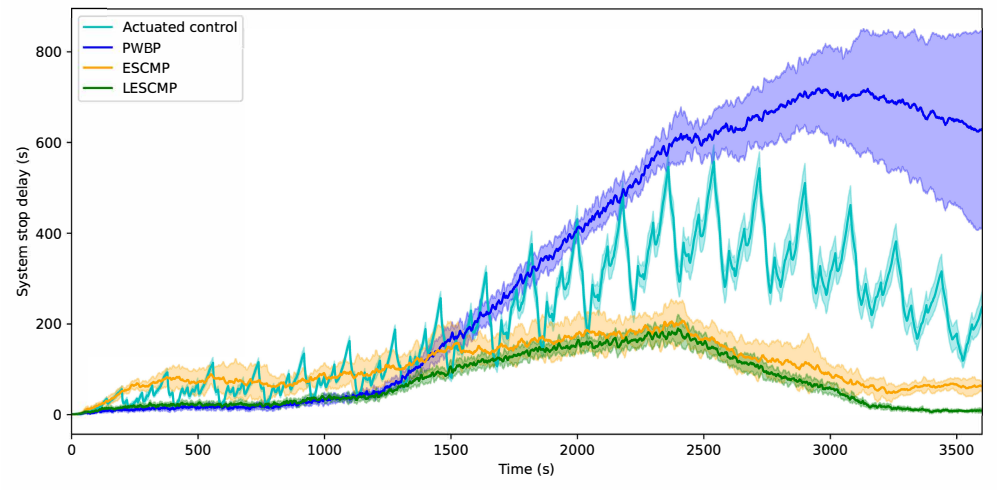

Figure 10: System stop delay under the different traffic signal controllers.

Table 1
Comparison of the system total delay of different controllers. Input demand is given by Figure 8.

Control policy	Average system total delay (h)	Total delay std (h)
CAC	68.56	45.66
PWBP	331.98	37.97
ESCMP	108.87	21.53
LESCMP	71.95	3.63

exactly the same for all the controllers. Figure 10 shows the mean and standard derivation of the system total delay of the four controllers. Compared with the delay curves of the three different max pressure controllers, the delay curve of the actuated control show obvious oscillation since the actuated control is quasi-periodic while the max pressure controllers do not follow the cyclic phase structure. When the traffic demand is low, almost all the max pressure controllers outperform the actuated control since the max pressure control without fixed phase sequence structure is more flexible than the actuated control. However, when the demand increased, the system delay of the PWBP increases significantly. This is because the max pressure control without considering the phase switching loss could lead to very frequent switching so that the traffic demand cannot be served and the queue built up quickly. Compared with the

Table 2
Mean and standard derivation of the system total delay (h) under different demand level (stationary arrival rate).

Control Policy	Different relative demand levels											
	0.2		0.4		0.6		0.8		1.0		1.2	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
CAC	12.6	4.4	26.6	7.7	44.4	11.6	77.5	20.4	185.0	77.7	422.9	219.5
PWBP	5.7	0.2	15.2	0.5	146.2	25.2	588.4	93.0	1292.9	170.6	1835.9	129.3
ESCMP	42.6	2.5	49.1	1.3	62.8	1.2	88.5	3.4	328.1	38.8	1020.9	134.5
LESCMP	9.1	0.1	21.5	0.3	40.9	0.5	79.4	1.1	169.9	6.5	680.4	131.0

PWBP without the switching curve, ESCMP can dynamically adjust the phase switching frequency so that it would switch less frequently under the higher traffic demand. Based on ESCMP, LESCMP performs even better since the parameter is further optimized using the RL algorithms, which is intuitive and consistent with the result in Figure 9. Table 1 lists the mean and standard derivation of the system total delay of the four controllers. ESCMP control outperformed both the actuated control and PWBP significantly and the RL can further improve ESCMP by further decreasing the system total delay by more than 35%.

The evaluation results given by Figure 10 and Table 1 use the same input demand distribution with the training process. Therefore, we design another "out-of-sample" experiment to test the four controllers under the different levels of stationary arrival rates. Similarly, the evaluation is repeated for 10 times for each controller with each demand rate. Table 2 shows the mean and standard derivation of the system total delay under different relative demand levels ranging from 0.2 to 1.2. This result is quite similar to the previous one. Both ESCMP and LESCMP performs very well in all levels of the demand. However, as a more flexible controller without a fixed cyclic phase structure, the max pressure controller suffers a larger variance when the traffic demand was high; although it had a much less mean value compared with the actuated control. The robustness of the max pressure control is not considered in this work and we leave that for future study.

6. Discussions and conclusions

This paper presents a framework that utilizes the policy-gradient reinforcement learning methods to learn a modified max pressure control policy considering the switching loss. The proposed max pressure control (SCMP) uses a switching rule that dynamically adjusts the switching frequency according to the congestion level. It is proved that SCMP is a throughput-optimal policy under the store-and-forward model with the phase switching loss. We also extend the theoretically derived SCMP to a more practical and flexible ESCMP with the weight curve and the switching curve. These two parametric curves are further optimized in LESCMP using the policy-gradient reinforcement learning algorithms. While the switching curve is used to address the switching loss caused by phase switching, the position weighted pressure can implicitly take the coordination between intersections into account. The simulation study established on a calibrated network showed that ESCMP and LESCMP outperformed the original max pressure control and vehicle-actuated control significantly.

The proposed max pressure control methods shows many advantages in the real-world implementation. The control policy is decentralized so that each intersection makes its own decision based on the upstream and downstream traffic state without requiring communication between intersections. The switching rule enables the control policy to dynamically adjust the switching frequency (equivalent to cycle lengths) according to the congestion level so that we do not have to split the whole day into different time of days (TOD) and perform different signal timing plans or adjust the parameters. Besides, the max pressure control is an end-to-end control policy, which directly generates the control policy from the observation; hence it can be efficiently implemented in the real world.

This paper assumes that the locations of all the vehicles are available for the signal controller, which could be obtained by the infrastructure sensor or with 100% penetration rate of the connected vehicles. When the complete traffic state is not available, for example, there are only loop detectors at certain locations or the penetration rate of connected vehicles is not high enough, a traffic state estimation might be required before the implementation of the max pressure control. There have been different methods proposed to estimate the traffic state using different data resources (Liu et al., 2009; Zhao et al., 2019; Wang et al., 2020). Varaiya (2013) has pointed that the stability will not be undermined in the store-and-forward model as long as the estimation of the pressure is unbiased. In fact, it

is easy to verify that, with the infinite link capacity assumption, the stability will not be influenced if the traffic state estimation is unbiased or the estimation error is bounded. However, this situation will change if the assumption does not hold. Therefore, for the store-and-forward model with finite link capacity, we might need additional requirements for the traffic state estimation error to insure the network queue length stability.

Acknowledgments

The authors would like to thank the US Department of Transportation (USDOT) Region 5 University Transportation Center: Center for Connected and Automated Transportation (CCAT) of the University of Michigan for funding this work. Yafeng Yin is also grateful for the support by National Science Foundation (CMMI-1904575). The authors would also like to thank Dr. Tian Mi at University of Michigan for sharing her SUMO simulation environment. The views presented in this paper are those of the authors alone.

References

Aboudolas, K., Papageorgiou, M., Kosmatopoulos, E., 2009. Store-and-forward based methods for the signal control problem in large-scale congested urban road networks. Transportation Research Part C: Emerging Technologies 17, 163–174.

Arel, I., Liu, C., Urbanik, T., Kohls, A.G., 2010. Reinforcement learning-based multi-agent system for network traffic signal control. IET Intelligent Transport Systems 4, 128–135.

Celik, G., Borst, S.C., Whiting, P.A., Modiano, E., 2016. Dynamic scheduling with reconfiguration delays. Queueing Systems 83, 87–129.

Chen, R., Hu, J., Levin, M.W., Rey, D., 2020. Stability-based analysis of autonomous intersection management with pedestrians. Transportation Research Part C: Emerging Technologies 114, 463–483.

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

Haklay, M., Weber, P., 2008. Openstreetmap: User-generated street maps. IEEE Pervasive Computing 7, 12–18.

Hsieh, P.C., Liu, X., Jiao, J., Hou, I.H., Zhang, Y., Kumar, P., 2017. Throughput-optimal scheduling for multi-hop networked transportation systems with switch-over delay, in: Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp. 1–10.

Khamis, M.A., Gomaa, W., 2014. Adaptive multi-objective reinforcement learning with hybrid exploration for traffic signal control based on cooperative multi-agent framework. Engineering Applications of Artificial Intelligence 29, 134–151.

Krajzewicz, D., Erdmann, J., Behrisch, M., Bieker, L., 2012. Recent development and applications of sumo-simulation of urban mobility. International journal on advances in systems and measurements 5.

Le, T., Kovács, P., Walton, N., Vu, H.L., Andrew, L.L., Hoogendoorn, S.S., 2015. Decentralized signal control for urban road networks. Transportation Research Part C: Emerging Technologies 58, 431–450.

Li, L., Jabari, S.E., 2019. Position weighted backpressure intersection control for urban networks. Transportation Research Part B: Methodological 128, 435–461.

Liang, E., Liaw, R., Nishihara, R., Moritz, P., Fox, R., Gonzalez, J., Goldberg, K., Stoica, I., 2017. Ray rllib: A composable and scalable reinforcement learning library. arXiv preprint arXiv:1712.09381 , 85.

Little, J.D., Graves, S.C., 2008. Little's law, in: Building intuition. Springer, pp. 81–100.

Liu, H.X., Wu, X., Ma, W., Hu, H., 2009. Real-time queue length estimation for congested signalized intersections. Transportation research part C: emerging technologies 17, 412–427.

Lo, H.K., 2001. A cell-based traffic control formulation: strategies and benefits of dynamic timing plans. Transportation Science 35, 148–164.

Neely, M.J., 2010. Stochastic network optimization with application to communication and queueing systems. Synthesis Lectures on Communication Networks 3, 1–211.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., Moritz, P., 2015. Trust region policy optimization, in: International conference on machine learning, pp. 1889–1897.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 .

Srikant, R., Ying, L., 2013. Communication networks: an optimization, control, and stochastic networks perspective. Cambridge University Press.

Sun, X., Yin, Y., 2018. A simulation study on max pressure control of signalized intersections. Transportation research record 2672, 117–127.

Sutton, R.S., Barto, A.G., 2018. Reinforcement learning: An introduction. MIT press.

Tassiulas, L., Ephremides, A., 1990. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks, in: 29th IEEE Conference on Decision and Control, IEEE. pp. 2130–2132.

Varaiya, P., 2013. Max pressure control of a network of signalized intersections. Transportation Research Part C: Emerging Technologies 36, 177–195.

Wada, K., Usui, K., Takigawa, T., Kuwahara, M., 2017. An optimization modeling of coordinated traffic signal control based on the variational theory and its stochastic extension. Transportation research procedia 23, 624–644.

Wang, X., Shen, S., Bezzina, D., Sayer, J.R., Liu, H.X., Feng, Y., 2020. Data infrastructure for connected vehicle applications. Transportation Research Record , 0361198120912424.

Wei, H., Chen, C., Zheng, G., Wu, K., Gayah, V., Xu, K., Li, Z., 2019a. Presslight: Learning max pressure control to coordinate traffic signals in arterial network, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1290–1298.

Wei, H., Zheng, G., Gayah, V., Li, Z., 2019b. A survey on traffic signal control methods. arXiv preprint arXiv:1904.08117 .

Xiao, N., Frazzoli, E., Li, Y., Wang, Y., Wang, D., 2014. Pressure releasing policy in traffic signal control with finite queue capacities, in: 53rd IEEE Conference on Decision and Control, IEEE. pp. 6492–6497.

Yau, K.L.A., Qadir, J., Khoo, H.L., Ling, M.H., Komisarczuk, P., 2017. A survey on reinforcement learning models and algorithms for traffic signal control. ACM Computing Surveys (CSUR) 50, 1–38.

Zaidi, A.A., Kulcsár, B., Wymeersch, H., 2016. Back-pressure traffic signal control with fixed and adaptive routing for urban vehicular networks. IEEE Transactions on Intelligent Transportation Systems 17, 2134–2143.

Zhao, Y., Zheng, J., Wong, W., Wang, X., Meng, Y., Liu, H.X., 2019. Various methods for queue length and traffic volume estimation using probe vehicle trajectories. Transportation Research Part C: Emerging Technologies 107, 70–91.

A. Proof of Theorem 1

Define $\boldsymbol{\mu}$ with $\mu_i(t) = \min\{x_{ij}(t), c_{ij}s_{ij}(t)\}$. Obviously, we have $\boldsymbol{\mu} \preceq \boldsymbol{C}\boldsymbol{s}$, that is $\boldsymbol{C}^{-1}\boldsymbol{\mu} \in S$. If there is no switching loss, the dynamics for each time step can be written as:

$$\boldsymbol{x}(t+1) = \boldsymbol{x}(t) + \boldsymbol{a}(t) - (\boldsymbol{I} - \boldsymbol{R})\boldsymbol{\mu}(t). \tag{33}$$

If Equation (21) is not true, then there exists an $i \in \mathcal{M}^o$ and a positive constant $\epsilon > 0$, such that $a_i - \{(\boldsymbol{I}-\boldsymbol{R})\boldsymbol{\mu}\}_i > \epsilon$. Applying 1-norm and taking the expectation on both sides of Equation (33) yield:

$$\mathbb{E}(\|x_i(t+1)\|) \geq \mathbb{E}(\|x_i(t)\|) + \epsilon \geq \ldots \geq \mathbb{E}(\|x_i(0)\|) + (t+1)\epsilon. \tag{34}$$

That is, the queue lengths of this movement i will diverge to infinity when the time keeps increasing. With the switching loss, the inequality will turn to a strict inequality and the network queue lengths will diverge more quickly. □

B. Proof of Lemma 1

Noted that this proof will involve many bounded positive constants such as c_3 in Equation (27); we choose not keep track of the exact mathematical expressions to simplify the proof unless necessary. In fact, all the terms ordered by $\|\boldsymbol{w}(\boldsymbol{x}(\tau_k))\|$ can be put into the bounded constant term, since these bounded constants will not undermine the stability of the network queue lengths. The Lyapunov drift Δ'_{τ_k} defined in Equation (26) can be written as:

$$\begin{aligned}\Delta'_{\tau_k} &= \mathbb{E}\left[\sum_i \int_{\xi=0}^{x_i(\tau'_{k+1})} w_i(\xi)d\xi - \sum_i \int_{\xi=0}^{x_i(\tau_k)} w_i(\xi)d\xi \,\middle|\, S_{\tau_k}\right] \\ &= \mathbb{E}\left[\sum_i w_i\left(x_i(\tau_k) + \beta\cdot\left(x_i(\tau'_{k+1}) - x_i(\tau_k)\right)\right)\left(x_i(\tau'_{k+1}) - x_i(\tau_k)\right) \,\middle|\, S_{\tau_k}\right],\end{aligned} \tag{35}$$

where the second equality comes from the mean-value theorem and $\beta \in [0,1]$. According to the last property of the weight function, there exists a constant B, such that the Lyapunov drift is bounded by:

$$\Delta'_{\tau_k} \leq \mathbb{E}\left[\sum_i w_i\left(x_i(\tau_k)\right)\cdot\left(x_i(\tau'_{k+1}) - x_i(\tau_k)\right) + B\cdot\sum_i\left(x_i(\tau'_{k+1}) - x_i(\tau_k)\right)^2 \,\middle|\, S_{\tau_k}\right] \tag{36}$$

which is equivalent to the matrix form:

$$\Delta'_{\tau_k} \leq \mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T\left(\boldsymbol{x}(\tau'_{k+1}) - \boldsymbol{x}(\tau_k)\right) + B\cdot\left(\boldsymbol{x}(\tau'_{k+1}) - \boldsymbol{x}(\tau_k)\right)^T\left(\boldsymbol{x}(\tau'_{k+1}) - \boldsymbol{x}(\tau_k)\right) \,\middle|\, S_{\tau_k}\right] \tag{37}$$

Let $\Delta'^{(1)}_{\tau_k}$ and $\Delta'^{(2)}_{\tau_k}$ be the first term and second term of Equation (37):

$$\Delta'^{(1)}_{\tau_k} = \mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T(\boldsymbol{x}(\tau'_{k+1}) - \boldsymbol{x}(\tau_k)) \,\middle|\, S_{\tau_k}\right]; \tag{38a}$$

$$\Delta'^{(2)}_{\tau_k} = \mathbb{E}\left[B\cdot(\boldsymbol{x}(\tau'_{k+1}) - \boldsymbol{x}(\tau_k))^T(\boldsymbol{x}(\tau'_{k+1}) - \boldsymbol{x}(\tau_k)) \,\middle|\, S_{\tau_k}\right]. \tag{38b}$$

We will bound these two terms of the Lyapunov drift separately. For $\boldsymbol{x}(\tau'_{k+1}) - \boldsymbol{x}(\tau_k)$ in the first term, we have the dynamics:

$$\boldsymbol{x}(\tau'_{k+1}) - \boldsymbol{x}(\tau_k) = \sum_{t=\tau_k}^{\tau'_{k+1}} \boldsymbol{a}(t) - (\boldsymbol{I} - \boldsymbol{R}')\min\{\boldsymbol{x}(\tau_k), (\tau'_{k+1} - \tau_k - T_r\cdot\boldsymbol{\Pi}_k)\boldsymbol{C}\boldsymbol{s}_k\} \tag{39}$$

where $\boldsymbol{R}'$ is the average turning ratio during the period:

$$R' = \frac{1}{\tau'_{k+1} - \tau_k}\sum_{t=\tau_k}^{\tau'_{k+1}} \boldsymbol{R}(t) \tag{40}$$

$\mathbf{\Pi}_k$ is the diagonal indicator matrix with $\{\mathbf{\Pi}_k\}_{mm} = 1$ if movement m is switched from the red light to the green light during this kth switching. $\boldsymbol{s}_k$ is the selected control policy during this period. By taking the expectation on both sides of Equation (39), and utilizing the fact that $\boldsymbol{a}(t)$ and $\boldsymbol{R}(t)$ are both i.i.d., we will get:

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{x}(\tau'_{k+1}) - \boldsymbol{x}(\tau_k)\right] &= (\tau'_{k+1} - \tau_k)\boldsymbol{a} - (\boldsymbol{I} - \boldsymbol{R})\min\{\boldsymbol{x}(\tau_k), (\tau'_{k+1} - \tau_k - T_r \cdot \mathbf{\Pi}_k)\boldsymbol{C}\boldsymbol{s}_k\} \\ &= (\tau'_{k+1} - \tau_k)\boldsymbol{a} - (\boldsymbol{I} - \boldsymbol{R})\min\{\boldsymbol{x}(\tau_k), \boldsymbol{D} \cdot \boldsymbol{C}\boldsymbol{s}_k\},\end{aligned} \tag{41}$$

where the matrix $\boldsymbol{D}$ is defined as:

$$\boldsymbol{D} = (\tau'_{k+1} - \tau_k) \cdot \boldsymbol{I} - T_r \cdot \mathbf{\Pi}_k \preceq (\tau'_{k+1} - \tau_k) \cdot \boldsymbol{I} \tag{42}$$

The matrix $\boldsymbol{D}$ is also a diagonal matrix with positive diagonal entries. Noted that not all the nodes are necessary to switch to a new phase after each switching but at least there will be one.

By applying the Equation (41) to the first term of the Lyapunov drift given by Equation (38a), it becomes:

$$\begin{aligned}\Delta'^{(1)}_{\tau_k} =& \mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T \cdot \left((\tau'_{k+1} - \tau_k)\boldsymbol{a} - (\boldsymbol{I} - \boldsymbol{R}) \cdot \min\{\boldsymbol{x}(\tau_k), \boldsymbol{D}\boldsymbol{C}\boldsymbol{s}_k\}\right)\middle| S_{\tau_k}\right] \\ =& \mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T \cdot \left((\tau'_{k+1} - \tau_k)\boldsymbol{a} - \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k + \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k - \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{R}) \cdot \min\{\boldsymbol{D}^{-1}\boldsymbol{x}(\tau_k), \boldsymbol{C}\boldsymbol{s}_k\}\right)\middle| S_{\tau_k}\right] \\ =& \mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{R})(\boldsymbol{C}\boldsymbol{s}_k - \min\{\boldsymbol{C}\boldsymbol{s}_k, \boldsymbol{D}^{-1}\boldsymbol{x}(\tau_k)\})\middle| S_{\tau_k}\right] \\ &+ \mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T \left((\tau'_{k+1} - \tau_k) \cdot \boldsymbol{a} - \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k\right)\middle| S_{\tau_k}\right].\end{aligned} \tag{43}$$

For the first item of RHS of the last equality, we have $\forall \boldsymbol{x}(\tau_k)$:

$$\begin{aligned}\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{R})(\boldsymbol{C}\boldsymbol{s}_k - \min\{\boldsymbol{C}\boldsymbol{s}_k, \boldsymbol{D}^{-1}\boldsymbol{x}(\tau_k)\}) &< (\tau'_{k+1} - \tau_k)\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T(\boldsymbol{I} - \boldsymbol{R})(\boldsymbol{C}\boldsymbol{s}_k - \min\{\boldsymbol{C}\boldsymbol{s}_k, \boldsymbol{D}^{-1}\boldsymbol{x}(\tau_k)\}) \\ &< B' \cdot (\tau'_{k+1} - \tau_k).\end{aligned} \tag{44}$$

where $B' > 0$ is a bounded constant. The first inequality of this equation comes directly from Equation (42). For the second inequality: when $\boldsymbol{D}^{-1} \cdot \boldsymbol{x}(\tau_k)$ within the $\min\{\cdot,\cdot\}$ operator is larger than the first term $\boldsymbol{C}\boldsymbol{s}_k$, the whole term $(\boldsymbol{C}\boldsymbol{s}_k - \min\{\cdot,\cdot\})$ will become zero. Therefore, the LHS of Equation (44) could be greater than zero only when the term $\boldsymbol{D}^{-1} \cdot \boldsymbol{x}(\tau_k)$ is less than $\boldsymbol{C}\boldsymbol{s}_k$; under this condition we can easily bound the whole LHS of of Equation (44) with a bounded $\boldsymbol{x}(\tau_k) \leq \boldsymbol{D}\boldsymbol{C}\boldsymbol{s}$.

By applying Equation (44) and (25) to Equation (43), the first term of the Lyapunov drift given by Equation (38a) can be bounded as:

$$\begin{aligned}\Delta'^{(1)}_{\tau_k} &\leq B' \cdot \mathbb{E}\left[\tau'_{k+1} - \tau_k \mid S_{\tau_k}\right] + \mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T \left((\tau'_{k+1} - \tau_k) \cdot \boldsymbol{a} - \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k\right)\middle| S_{\tau_k}\right] \\ &\leq c'_1 + c'_2 F(\|\boldsymbol{x}(\tau_k)\|) + \mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T \left((\tau'_{k+1} - \tau_k) \cdot \boldsymbol{a} - \boldsymbol{D}(\boldsymbol{I} - \boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k\right)\middle| S_{\tau_k}\right],\end{aligned} \tag{45}$$

where $c'_1, c_2 > 0'$ are positive constants.

We can further bound the last term of Equation (45) utilizing the condition that the demand $(\boldsymbol{a}, \boldsymbol{R})$ is within the interior of the admissible demand region. Since $(\boldsymbol{a}, \boldsymbol{R}) \in \text{int}\mathcal{D}$, we have:

$$\boldsymbol{a} - (\boldsymbol{I} - \boldsymbol{R})\boldsymbol{C}\boldsymbol{s}^\pi \preceq -\epsilon\mathbf{1}, \quad \exists \boldsymbol{s}^\pi \in S, \epsilon > 0. \tag{46}$$

Since the control policy $\boldsymbol{s}_k$ is selected according to Equation (7), we have:

$$\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T(\tau'_{k+1} - \tau_k)(\boldsymbol{I} - \boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k \geq \boldsymbol{w}(\boldsymbol{x}(\tau_k))^T(\tau'_{k+1} - \tau_k)(\boldsymbol{I} - \boldsymbol{R})\boldsymbol{C}\boldsymbol{s}^\pi \tag{47}$$

The last term of Equation (45) can be simplified as:

$$\begin{aligned}
&\mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T\left((\tau'_{k+1}-\tau_k)\cdot\boldsymbol{a}-\boldsymbol{D}(\boldsymbol{I}-\boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k)\right)\middle|\, S_{\tau_k}\right]\\
=&\,\mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T(\tau'_{k+1}-\tau_k)(\boldsymbol{a}-(\boldsymbol{I}-\boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k)\mid S_{\tau_k}\right]+\mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T\boldsymbol{\Pi}_k(\boldsymbol{I}-\boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k)\mid S_{\tau_k}\right]\\
\le&\,\mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T(\tau'_{k+1}-\tau_k)(\boldsymbol{a}-(\boldsymbol{I}-\boldsymbol{R})\boldsymbol{C}\boldsymbol{s}^{\pi})\mid S_{\tau_k}\right]+\mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T\boldsymbol{\Pi}_k(\boldsymbol{I}-\boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k)\mid S_{\tau_k}\right]\\
\le&-\epsilon\|\boldsymbol{w}(\boldsymbol{x}(\tau_k))\|\cdot\mathbb{E}\left[(\tau'_{k+1}-\tau_k)\mid S_{\tau_k}\right]+\mathbb{E}\left[\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T\boldsymbol{\Pi}_k(\boldsymbol{I}-\boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k)\mid S_{\tau_k}\right]\\
\le&-\epsilon\|\boldsymbol{w}(\boldsymbol{x}(\tau_k))\|\cdot\left(\mathbb{E}\left[(\tau'_{k+1}-\tau_k)\mid S_{\tau_k}\right]-c'_3\right)
\end{aligned}\tag{48}$$

where the first inequality comes from Equation (47). The last inequality holds since the norm of the vector $\boldsymbol{\Pi}_k(\boldsymbol{I}-\boldsymbol{R})\boldsymbol{C}\boldsymbol{s}_k$ can be easily bounded by a constant.

Applying Equation (24b) to Equation (48) and substituting it to Equation (45), we bound $\Delta^{(1)}_{\tau_k}$ as follows:

$$\Delta'^{(1)}_{\tau_k}\le c'_1+c'_2F(\|\boldsymbol{x}(\tau_k)\|)-\epsilon'(1-\delta''(\|\boldsymbol{x}(\tau_k)\|))F(\|\boldsymbol{x}(\tau_k)\|)\|\boldsymbol{w}(\boldsymbol{x}(\tau_k))\|.\tag{49}$$

where $\delta''(\cdot)$ satisfies $\lim_{x\to\infty}\delta''(x)=0$. So far we have bounded the first term of the Lyapunov drift given by Equation (38a).

For the second term $\Delta'^{(2)}_{\tau_k}$, it can be simply bounded as:

$$\begin{aligned}
\Delta'^{(2)}_{\tau_k}&=\boldsymbol{B}\cdot\mathbb{E}\left[(\boldsymbol{x}(\tau'_{k+1})-\boldsymbol{x}(\tau_k))^T(\boldsymbol{x}(\tau'_{k+1})-\boldsymbol{x}(\tau_k))\middle|\, S_{\tau_k}\right]\\
&\le\boldsymbol{B}\cdot\mathbb{E}\left[(\tau'_{k+1}-\tau_k)^2(\boldsymbol{a}+\boldsymbol{c})^T(\boldsymbol{a}+\boldsymbol{c})\right]\\
&\le c'_4\cdot\left[T_r^2+c'_5\left(F(\|\boldsymbol{x}(\tau_k)\|)\right)^2\right]\\
&\le c'_4\cdot\left(F(\|\boldsymbol{x}(\tau_k)\|)\right)^2+c'_4
\end{aligned}\tag{50}$$

where the first inequality is a loose relaxation from Equation (39). $\mathbb{E}\|\boldsymbol{a}+\boldsymbol{C}\|_2^2$ can be bounded by a constant since the variance of the arrival is assumed to be bounded; the second inequality is from Equation (24c). Summing up the first and second term leads to:

$$\begin{aligned}
\Delta'_{\tau_k}=&\Delta'^{(1)}_{\tau_k}+\Delta'^{(2)}_{\tau_k}\\
=&c'_1+c'_2F(\|\boldsymbol{x}(\tau_k)\|)+c'_4\cdot\left(F(\|\boldsymbol{x}(\tau_k)\|)\right)^2-\epsilon'\left(1-\delta''(\|\boldsymbol{x}(\tau_k)\|)\right)F(\|\boldsymbol{x}(\tau_k)\|)\|\boldsymbol{w}(\boldsymbol{x}(\tau_k))\|\\
\le&c_3-\eta F(\|\boldsymbol{x}(\tau_k)\|)\|\boldsymbol{w}(\boldsymbol{x}(\tau_k))\|
\end{aligned}\tag{51}$$

since $F(\cdot)$ is a sublinear function and $\delta'(\cdot)$ converges to zero when the queue lengths go to infinity. This completes the proof of the Lemma 1. □

C. Proof of Theorem 3

We will first show that under the switching rule defined by Equation (10), for each k, there is always a random stopping time τ'_{k+1}, when the switching is activated according to the switching rule defined in Lemma 2, before the switching condition given by Equation (10) is satisfied. Then according to Lemma 2, the proposed generalized max pressure control policy will satisfy the first three conditions with the first condition as an inequality. Then we only need to check the last condition.

For each sampled trajectory of the system state, we have:

$$\begin{aligned}
&-\boldsymbol{w}(\boldsymbol{x}(\tau_k))^T(\boldsymbol{I}-\boldsymbol{R})(\boldsymbol{s}(\tau_{k+1})-\boldsymbol{s}(\tau_k))\ge 0\\
&\boldsymbol{w}(\boldsymbol{x}(\tau_{k+1}))^T(\boldsymbol{I}-\boldsymbol{R})(\boldsymbol{s}(\tau_{k+1})-\boldsymbol{s}(\tau_k))\ge F(\|\boldsymbol{x}(\tau_{k+1})\|)
\end{aligned}\tag{52}$$

where the first inequality is due to the fact that at time τ_k when a new policy $\boldsymbol{s}(\tau_k)$ is selected, it maximized the weight given by Equation (7). The second inequality follows from the switching rule given by Equation (10). Summing these two equations up yields:

$$[\boldsymbol{w}(\boldsymbol{x}(\tau_{k+1})) - \boldsymbol{w}(\boldsymbol{x}(\tau_k))]^T(\boldsymbol{I} - \boldsymbol{R})(\boldsymbol{s}(\tau_{k+1}) - \boldsymbol{s}(\tau_k)) \geq F(\|\boldsymbol{x}(\tau_{k+1})\|). \tag{53}$$

Then we can have:

$$\begin{aligned} \|\boldsymbol{w}(\boldsymbol{x}(\tau_{k+1})) - \boldsymbol{w}(\boldsymbol{x}(\tau_k))\| \cdot \|(\boldsymbol{I} - \boldsymbol{R})(\boldsymbol{s}(\tau_{k+1}) - \boldsymbol{s}(\tau_k))\| &\geq F(\|\boldsymbol{x}(\tau_{k+1})\|) \\ c\|\boldsymbol{x}(\tau_{k+1}) - \boldsymbol{x}(\tau_k)\| &\geq F(|\|\boldsymbol{x}(\tau_k)\| - \|\boldsymbol{x}(\tau_{k+1}) - \boldsymbol{x}(\tau_k)\||) \\ c\|\boldsymbol{x}(\tau_{k+1}) - \boldsymbol{x}(\tau_k)\| &\geq F(\|\boldsymbol{x}(\tau_k)\|) - \boldsymbol{B} \cdot \|\boldsymbol{x}(\tau_{k+1}) - \boldsymbol{x}(\tau_k)\| \end{aligned} \tag{54}$$

where both $\boldsymbol{B}$ and c are bounded positive constants. The first inequality (first line) follows from the Cauchy-Schwarz inequality; the change of the LHS from the first inequality to the second inequality utilizes the condition that the general weight function has a bounded first-order derivative and the term $\|(\boldsymbol{I} - \boldsymbol{R})(\boldsymbol{s}(\tau_{k+1}) - \boldsymbol{s}(\tau_k))\|$ can be bounded by a constant; the last inequality follows from the condition that $F(\cdot)$ is a sublinear function.

Equation (54) can be simplified as:

$$\|\boldsymbol{x}(\tau_{k+1}) - \boldsymbol{x}(\tau_k)\| \geq \frac{1}{Bc} \cdot F(\|\boldsymbol{x}(\tau_k))\|), \tag{55}$$

which indicates that before the switching is activated according to Equation (10), a corresponding switching activation condition given by Lemma 2 under $\theta = 1/(Bc) > 0$ will be satisfied in advance. Then according to Lemma 2, the first three conditions of the Theorem 2 are satisfied.

Here we will show that the last condition is also satisfied. Before the switching is activated, we have:

$$\max_{\boldsymbol{s}} \boldsymbol{w}(\boldsymbol{x}(\tau_{k+1}))^T(\boldsymbol{I} - \boldsymbol{R})(\boldsymbol{s} - \boldsymbol{s}(\tau_k)) < F(\|\boldsymbol{x}(\tau_{k+1})\|), \tag{56}$$

which means that the current control policy is still close to the max pressure policy. Let Δ_χ be the one-slot conditional Lyapunov drift. Repeating the similar procedure in the proof of Lemma 1 with the only change from the complicated multiple-slots case to one-slot case, we can finally have:

$$\Delta_\chi \leq c - \epsilon\|\boldsymbol{x}(t)\| + F(\|\boldsymbol{x}(t)\|) \tag{57}$$

which is equivalent to the last condition in Theorem 2 since $F(\cdot)$ is a sublinear function. □