

Singapore Management University

Institutional Knowledge at Singapore Management University

Research Collection School Of Computing and Information Systems

School of Computing and Information Systems

10-2020

Online traffic signal control through sample-based constrained optimization

Srishti DHAMIJA

Alolika GON

Pradeep VARAKANTHAM

William YEOH

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research



Part of the [Artificial Intelligence and Robotics Commons](#), [Operations Research, Systems Engineering and Industrial Engineering Commons](#), and the [Transportation Commons](#)

This Conference Proceeding Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email cherylds@smu.edu.sg.

Online Traffic Signal Control through Sample-Based Constrained Optimization

Srishti Dhamija,¹ Alolika Gon,² Pradeep Varakantham,² William Yeoh³

¹Johns Hopkins University ²Singapore Management University* ³Washington University in St. Louis
srishti@jhu.edu, {alolikagon, pradeepv}@smu.edu.sg, wyeoh@wustl.edu

Abstract

Traffic congestion reduces productivity of individuals by increasing time spent in traffic and also increases pollution. To reduce traffic congestion by better handling dynamic traffic patterns, recent work has focused on online traffic signal control. Typically, the objective in traffic signal control is to minimize expected delay over all vehicles given the uncertainty associated with the vehicle turn movements at intersections. In order to ensure responsiveness in decision making, a typical approach is to compute a schedule that minimizes the delay for the expected scenario of vehicle movements instead of minimizing expected delay over the feasible vehicle movement scenarios. Such an approximation degrades schedule quality with respect to expected delay as vehicle turn uncertainty at intersections increases.

We introduce TUSERACT (TUrN-SaMplE-based Real-time trAffic signal ConTrol), an approach that minimizes expected delay over samples of turn movement uncertainty of vehicles. Specifically, our key contributions are: (a) By exploiting the insight that vehicle turn movements do not change with traffic signal control schedule, we provide a scalable constraint program formulation to compute a schedule that minimizes expected delay across multiple vehicle movement samples for a traffic signal; (b) a novel mechanism to coordinate multiple traffic signals through vehicle turn movement samples; and (c) a comprehensive experimental evaluation to demonstrate the utility of TUSERACT over SURTRAC, a leading approach for online traffic signal control which makes the aforementioned approximation. Our approach provides substantially lower (up to 60%) mean expected delay relative to SURTRAC with very few turn movement samples while providing real-time decision making on both real and synthetic networks.

Introduction

Sub-optimal traffic signal control can significantly increase traffic congestion (Chin et al. 2004) due to either: (i) Improper allocation of green time to multiple traffic streams;

(ii) lack of coordination between traffic lights; or (iii) inability to respond to changing traffic patterns in real time. Improvements to traffic signal mechanisms can reduce traffic congestion and therefore increase the effective capacity of existing road networks. Computing an offline static schedule (which indicates the green duration for competing traffic streams at each traffic signal) is not viable, as such static schedules cannot adapt to changing traffic patterns and can result in situations where vehicles travelling in one direction wait at a traffic signal even when there are no vehicles travelling in other directions. Therefore, recent work has focussed on online traffic signal control.

While online traffic signal control can be significantly more useful, it is also a challenging problem because scalable, network-wide control must be achieved under limited planning time, constantly changing traffic patterns, and real-world uncertainty associated with vehicle turn movements (Xie et al. 2014; Cai et al. 2009; Yu and Recker 2006).

Related Work: Owing to the practical benefits and the challenging nature of the underlying control problem, there have been multiple threads of research dedicated to online traffic signal control. First, we have the centralized traffic-responsive signal control systems (Guilliard et al. 2016a; Sims and Dobinson 1980; Robertson and Bretherton 1991; Luyanda et al. 2003) that are adept at reducing network-wide delays. However, they are not scalable since the scale of problems increases exponentially with each additional traffic intersection. Second, we have research (Sen and Head 1997; Gartner, Pooran, and Andrews 2002) that has focused on decentralized adaptive traffic signal control mechanisms without explicitly considering neighbouring intersections while planning. However, such approaches are myopic and can lead to traffic build ups across the network. Third, we have methods based on multi-agent Reinforcement Learning (Bazzan 2005; Kuyer et al. 2008) and Deep Reinforcement Learning (Li, Lv, and Wang 2016; Mousavi, Schukat, and Howley 2017; Prabuchandran, AN, and Bhatnagar 2014; Prashanth and Bhatnagar 2010; Shabestary and Abdulhai 2018; Wei et al. 2018), which are decentralized and also coordinated with neighbouring intersections. Unfortunately, RL based approaches have multiple issues which have prevented deployment on real traffic lights: (1) Require elabo-

*This research was supported by the Singapore Ministry of Education Academic Research Fund (AcRF) Tier 2 grant MOE2016-T2-1-174.

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

rate simulators (typically present in games) and millions of simulations of traffic flows to learn good policies for a single intersection; (2) As highlighted in Van der Pol and Oliehoek (2016), Deep MARL approaches for traffic intersection control are not stable and may not converge to solutions. (3) The number of simulations required grows exponentially in number of agents. Papers in DeepRL have shown results on networks with at most four intersections (Van der Pol and Oliehoek 2016) and limited traffic.

Finally, we have SURTRAC (Scalable Urban Traffic Control) (Xie, Smith, and Barlow 2012; Xie et al. 2012), which is a real-time, distributed, schedule-driven approach and a leading approach for online traffic signal control. In this paper, we primarily focus on experimental comparison against SURTRAC since it provides a significant improvement over traditional traffic signal control methods; scales up to networks close to an order of magnitude larger than the related work discussed and has been deployed on real traffic signals in Pittsburgh. Despite its scalability and effectiveness, SURTRAC has two fundamental limitations: (1) Instead of minimizing expected delay over possible turn movement scenarios of vehicles, it minimizes delay for the expected scenario of vehicular turn movements. (2) In order to ensure scalable coordination among traffic signals, only the expected scenario of outgoing traffic is communicated to neighbouring traffic signals. While these approximations ensure real-time tractability, schedule quality with respect to expected delay degrades as probability of vehicles taking turns at intersections increases. Since our focus is on improving a fundamental approximation in SURTRAC, we choose its most basic version (Xie, Smith, and Barlow 2012; Xie et al. 2012) for comparison. Recent enhancements (Hu and Smith 2018; Goldstein and Smith 2018) to SURTRAC are complementary to our approach.

Contributions: To address these key limitations, we introduce TUSERACT (TUrn-SampLE-based Real-time trAffic signal ConTrol), a turn-sample-driven distributed scheduling approach to traffic signal control. TUSERACT (i) optimizes expected delay over a small set of vehicle movement scenarios and (ii) communicates samples of projected outgoing traffic to neighbouring traffic signals. This is achieved through a combination of two key contributions: (i) By exploiting the insight that vehicle turn movements are independent of traffic control schedule, we provide a scalable constraint programming formulation that computes an expected delay minimizing schedule over a few vehicle movement samples. (ii) A novel communication mechanism between traffic signals at neighbouring intersections that is based on samples of vehicle turn movements.

On multiple benchmark problems considered in the literature and on a real network from Pittsburgh, we demonstrate that TUSERACT is able to significantly outperform the leading approach for real-time traffic signal control, SURTRAC. We were able to consistently reduce the expected delay under varying traffic demand conditions.

Online Traffic Signal Control (OTSC)

Traffic signal control systems aim to give right-of-way to competing streams of oncoming traffic at intersections so as

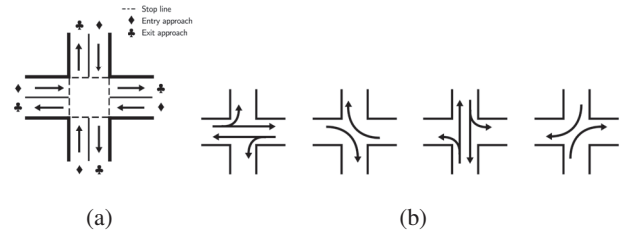


Figure 1: (a) An intersection with multiple entry and exit approaches. Arrows indicate the direction of traffic flow. (b) Phase design for a 4-phase intersection. Each arrow represents a turn movement, i.e., traffic flow from an entry road to an exit road at the intersection. Turn movements that can safely be given right-of-way simultaneously are grouped into a *phase*.

to minimize network-wide vehicular expected delays. Online traffic signal control ensures adaptability to constantly changing traffic patterns. In this section, we intuitively and formally describe the key aspects (local context and relevant non-local context) of the online traffic signal control problem at each traffic signal.

Intuitively, each traffic signal has to compute a schedule for different competing traffic streams (referred to as phases) based on its local context – operating constraints, observed traffic and initial conditions – and relevant traffic from neighbouring traffic signals. While there is observed traffic information, there is uncertainty about whether each vehicle will continue straight or take a turn at an intersection. This uncertainty results in uncertainty about traffic in each phase and consequently in uncertainty about delay along each phase. Therefore, the goal is to compute a schedule that minimizes expected delay in crossing the intersection given uncertainty over turn movements of vehicular traffic.

Local Context – Operating Constraints, λ_i : Intuitively, there are three parts to the *operating constraints* at an intersection i :

- First, we have the possible turn movements at a traffic signal or intersection.¹ Figure 1a provides an example of possible directions for traffic at an intersection. Formally, the set of possible turn movements is represented using Γ_i .
- Second, we have turn probabilities for different turns at an intersection. Formally, p_i^τ represents the probability of turn movement $\tau \in \Gamma_i$. p_i and p_{N_i} refers to the probability distributions for all possible turns at an intersection i and its neighbours N_i respectively.
- Finally, we have the set of phases, where each phase represents a set of non-conflicting turn movements as shown in Figure 1b. The ordering of phases, the lower and upper bounds on the green time for each phase, inter-green time between phases are all critical operating constraints.

Formally, ϕ_i is the *phase model* for intersection i . ϕ is an ordered set of phases that are cyclically given right-of-way at the intersection. Each phase ψ is further characterized by the tuple $\langle \mathcal{T}, G_{min}, G_{max}, Y \rangle$, where $\mathcal{T} \subset \Gamma$ is

¹We will refer to traffic signal and intersection synonymously.

the set of turns that have right-of-way during the phase, G_{min} and G_{max} are the lower and upper bounds on the time for which ψ has right-of-way (green time), and Y is the fixed inter-green time (informally, yellow time) that must be applied after ψ , during which no phase has right-of-way to ensure safety.

Local Context – Observed Traffic, δ_i : *Observed traffic* at intersection i along different phases is crucial for the decisions on how much green time to allocate to each of the phases. In this paper, δ is represented in terms of vehicle cluster sequences detected on the intersection’s different entry roads (Figure 1a).

Local Context – Initial Conditions, θ_i^0 : *Initial conditions* represent the initial conditions at the intersection at the decision point, i.e., the phase that has right-of-way (or green time) at the decision point and the time for which it has been green (current phase duration). The initial conditions determine the extension feasibility of the current phase.

Relevant Non-Local Traffic, δ_{N_i} : The *non-local traffic conditions*, primarily the traffic conditions at neighbouring intersections, have an impact on the delay-minimizing traffic signal plan computed at each intersection. δ_{-i} refers to the traffic conditions at all other intersections in the network. Formally, at a decision step, δ_{-i} is the non-local context of relevance to i and is communicated to i from one or more intersections in $V \setminus \{i\}$, where V is the set of all intersections in the network. Since we are interested in scalable control, we assume that the non-local traffic at a decision step is communicated to i solely from its immediate neighbouring intersections N_i .

OTSC Objective: At each intersection, given a planning horizon, local context and non-local traffic conditions at a decision point, the goal is to compute an expected delay-minimizing traffic signal timing plan. Formally, a *traffic signal timing plan* computed at intersection i specifies the *duration of each phase* $\psi_i^k \in \phi_i$ in every cycle r of phases and is represented as $\pi_i^{k,r}$.

Due to turn probabilities (i.e., p_i, p_{N_i}), there is uncertainty about traffic reaching the intersection on each phase (i.e., about δ_i^k) and consequently there is uncertainty about the delay, \mathcal{L} . We compute a traffic signal timing plan, π_i that minimizes the expected delay for the vehicles approaching the intersection:

$$\min_{\pi_i} \mathbb{E}_{p_i, p_{N_i}} \left[\mathcal{L} \left(\delta_i^0, \dots, \delta_i^{|\phi_i|-1} \mid \delta_i, \delta_{N_i}, \lambda_i, \pi_i \right) \right] \quad (1)$$

The computed traffic signal timing plan is then implemented up to the next decision point, after which it is recomputed to account for newly arrived traffic. Hence, it is an *online* traffic control mechanism.

Here, we consider systems that simply make a *termination* or *extension* decision for the current phase and then recompute the plan at the next decision step:

- **Termination:** The current phase is terminated and the next phase is given right-of-way for a minimum green time after an appropriate yellow time.
- **Extension:** The current phase is extended for a duration $\epsilon > 0$.

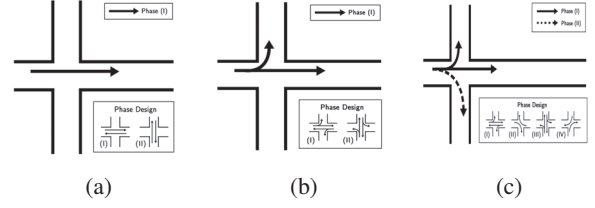


Figure 2: Manifestations of: (a) No uncertainty; (b) Intra-phase uncertainty; (c) Intra- and inter-phase; uncertainty.

Like in previous work (Hu and Smith 2018), we assume that minimizing expected delay over all traffic signals is equivalent to minimizing expected delay at each individual traffic signal subject to each individual traffic signal considering relevant traffic from neighbours.

Turn-Sample-based Real-time traffic signal Control (TUSERACT)

In this section, we describe our key contribution, TUSERACT that is an online approach for minimizing expected delay in OTSC problems. In order to appropriately situate our contribution, we first provide details of SURTRAC, a state-of-the-art OTSC method that has been successfully deployed to control traffic signals in Pittsburgh (Xie, Smith, and Barlow 2012; Xie et al. 2012). To ensure scalability and responsiveness (taking decisions within a few seconds), it computes a schedule that minimizes the delay for expected number of vehicles:

$$\min_{\pi_i} \mathcal{L} \left(\mathbb{E}_{p_i, p_{N_i}} [\delta_i^0], \dots, \mathbb{E}_{p_i, p_{N_i}} [\delta_i^{|\phi_i|-1}] \mid \delta_i, \delta_{N_i}, \lambda_i, \pi_i \right) \quad (2)$$

Note that this objective considers delay, \mathcal{L} for expected traffic in each of the phases, $E_{p_i, p_{N_i}} [\delta_i^k]$ and hence differs from the actual OTSC objective of minimizing expected delay (see Equation 1). The key cause of the difference between the two objectives is the uncertainty associated with vehicle turn movements. This uncertainty manifests itself as:

- **Intra-phase uncertainty:** This arises when vehicles on an entry edge can exit the intersection in the same phase, but on to different exit edges, as shown in Figure 2b. This impacts traffic outflows δ_{N_i} communicated by and to intersection i .
- **Intra- and inter-phase uncertainty:** This arises when vehicles on an entry edge can exit the intersection in multiple, competing phases, as shown in Figure 2c. This has a direct impact not only on δ_{N_i} , but also on per-phase traffic flow estimation at intersection i , i.e., δ_i^k .

TUSERACT is a distributed, schedule-driven approach to traffic signal control that minimizes expected delay over samples of the intra and inter-phase uncertainty.

The first key observation that we exploit in TUSERACT is that turn uncertainty is independent of the traffic signal timing plan. That is to say, whether a vehicle turns at an intersection is typically not dependent on how long the vehicle

had to wait at a traffic signal. Turns are typically decided beforehand.

Due to this independence of samples and traffic signal timing plan, we can employ a sample-based approach, where (i) samples of traffic turn movements are generated beforehand (based on the turn probabilities) and (ii) a traffic signal timing plan that minimizes expected delay over the generated samples can be computed through optimization.

Formally, each realization/sample $\xi \in \Xi_i$ is represented as

$$\xi = \{\delta_i^0(\xi), \dots, \delta_i^{|\phi_i|-1}(\xi)\}$$

where $\delta_i^k(\xi)$ is the traffic along phase k at intersection i according to realization ξ . Using *Sample Average Approximation* (SAA) (Kleywegt, Shapiro, and Homem-de Mello 2002), we can approximate the expectation of delay over intra- and inter-phase uncertainty by averaging delay across multiple realizations of traffic flow along different phases. This approximates the original stochastic optimization problem of Equation 1 as the following deterministic problem:

$$\min_{\pi_i} \frac{1}{|\Xi_i|} \sum_{\xi \in \Xi_i} \mathcal{L}(\xi \mid \delta_i, \delta_{N_i}, \lambda_i, \pi_i) \quad (3)$$

The second key observation that we exploit in TUSERACT is that this is primarily a scheduling problem with many time interval variables (corresponding to green times associated with the phases). To effectively exploit this observation, we employ a *Constraint Program (CP) solver*, instead of a *Mixed Integer Programming (MIP) solver*, to solve the above sample-based deterministic problem.

Overall, at each decision step, each intersection independently samples turn movements for observed vehicles, clusters them by proximity (Xie et al. 2012), computes a signal timing plan that minimizes the average delay across these clustered samples, terminates or extends the current phase according to the computed plan, and communicates projected traffic outflow samples to neighbouring intersections. The key changes with respect to SURTRAC are (i) a sampling-based objective function (ii) sampling-based communication (iii) cluster division (preemptive scheduling). We now describe the major components of our sample-based constraint programming approach.

Constrained Optimization

As we described above, our approach to the scheduling problem under uncertainty is based on sample average approximation, where we aim to compute a signal timing plan that minimizes the average delay across sampled traffic inflows. We formulate this deterministic, sample-based scheduling problem (Equation 3) as a constraint program, and solve it using the IBM ILOG CP Optimizer. Specifically, we compute a schedule for clusters of vehicles (as multiple vehicles can be close to each other) and not for individual vehicles.

Inputs: The key inputs to the constraint program at intersection i at a particular decision point are as follows:

Ξ_i : Each sample, $\xi \in \Xi_i$ describes the incoming cluster sequence $\delta_i^k(\xi)$ for each phase $\psi^k \in \phi$, ordered by arrival times at the intersection. Each cluster is represented as

$c_{i,q}^k(\xi) \in \delta_i^k(\xi)$, and points to the clustered group of vehicles at position q in the sequence on phase k at intersection i according to sample ξ . Each cluster has four attributes: (i) number of vehicles in the cluster ($|c_{i,q}^k(\xi)|$); (ii) arrival time at the intersection stop line denoted by $a_{i,q}^k(\xi)$ (with $a_{i,q}^k(\xi) \geq a_{i,q}^{k-1}(\xi)$); (iii) length of the cluster denoted by $l_{i,q}^k(\xi)$; and (iv) cluster composition (arrival times and sampled exit edges for each vehicle in the cluster) denoted by $\gamma_{i,q}^k(\xi)$.

H Planning horizon or number of cycles. A cycle represents one complete sequence of all phases.

$\langle \lambda_i, \delta_{N_i} \rangle$: Refer to Section for the inputs from the underlying Online Traffic Signal Control (OTSC) problem.

Decision Variables: We model our decision variables using the notion of *interval variables*, which allow us to intuitively model scheduling problems in terms of intervals of time. An interval variable ι is an interval of time $[start(\iota), end(\iota)]$. $length(\iota)$ is defined as $end(\iota) - start(\iota)$. An interval variable can be optional, i.e., it may or may not be present in the solution. The key decision variables are as follows:

- *Traffic signal timing plan:* This represents the amount of green time to be allocated for different phases. $\pi_i^{k,r}$, represents the interval corresponding to phase k in cycle r for intersection i .
- *Outgoing cluster intervals:* Each cluster of vehicles could potentially leave an intersection in multiple different phases and different cycles. This set of interval variables captures these outgoing fragments of a cluster. Specifically, $O_{i,q}^{k,r}(\xi)$, represents the fragment of cluster $c_{i,q}^k(\xi)$ scheduled in cycle r (if any). We assume that clusters are divisible and that fragments of a cluster may be scheduled in any cycle. $start(O_{i,q}^{k,r}(\xi))$ represents the time at which the cluster fragment starts leaving the intersection by crossing the stop line. The absence of $O_{i,q}^{k,r}(\xi)$ implies that no fragment of $c_{i,q}^k$ is scheduled in cycle r .

Constraints: We now describe intuitively and formally the main constraints that are required to compute an optimal traffic signal control plan. The computed signal timing plan π_i and outgoing cluster departures are constrained by the the initial conditions at the decision point, the observed traffic inflow, and the distributed traffic signal control inputs. The constraints are summarized formally in Table 1 and are intuitively described below:

Phase duration: The duration of each phase is constrained by a minimum green and a maximum green time predefined by the operating constraints of the traffic signal.

Phase order: In each cycle, the phases ϕ_i must be given right-of-way in a fixed order per the operating constraints of the traffic signal, and a fixed intergreen time must be applied between consecutive phases.

Cycle order: The cycles occur in a fixed order, and a fixed inter-green time is applied between consecutive cycles.

Cluster fragment length: We assume that clusters are divisible, and that fragments of a single cluster can be scheduled in different cycles. To ensure correct scheduling of the fragments we have two constraints. The first ensures

Table 1: Constraint Program, TUSERACT

Name	Constraint	
Phase duration	$length(\pi_i^{k,r}) \in [G_{min}^k, G_{max}^k]$	$\forall r, \psi^k \in \phi_i$
Phase order	$start(\pi_i^{k,r}) = end(\pi_i^{k-1,r}) + Y^{k-1}$	$\forall \psi^k \in \phi_i, k \neq 1, r$
Cycle order	$start(\pi_i^{1,r}) = end(\pi_i^{ \phi_i-1 ,r-1}) + Y^{ \phi_i-1 }$	$\forall r \neq 1$
Cluster fragment length	$length(O_{i,q}^{k,r}(\xi)) \in [1, l_{i,q}^k(\xi)]$	$\forall \xi \in \Xi_i, \psi^k \in \phi_i, r, q$
Cluster fragment length	$\sum_r length(O_{i,q}^{k,r}(\xi)) = l_{i,q}^k(\xi)$	$\forall \xi \in \Xi_i, \psi^k \in \phi_i, r, q$
Fragment departure	$start(O_{i,q}^{k,r}(\xi)) \geq a_{i,q}^k(\xi)$	$\forall \xi \in \Xi_i, k, r, q$
Fragment departure	$start(O_{i,q}^{k,r}(\xi)) \geq start(\pi_i^{k,r})$	$\forall \xi \in \Xi_i, \psi^k \in \phi_i, r, q$
Fragment departure	$end(O_{i,q}^{k,r}(\xi)) \leq end(\pi_i^{k,r})$	$\forall \xi \in \Xi_i, \psi^k \in \phi_i, r, q$
Cluster precedence	$presenceOf(O_{i,q}^{k,r}(\xi)) \rightarrow not(presenceOf(O_{i,q-1}^{k,r'}(\xi)))$ $k \neq 0, r, r' \in \{r+1, r+2, \dots\}, q$ (i.e. r' is a succeeding cycle)	$\forall \xi \in \Xi_i$
Cluster precedence	$end(O_{i,q-1}^{k,r}(\xi)) \leq start(O_{i,q}^{k,r}(\xi))$	$\forall \xi \in \Xi_i, \psi^k \in \phi_i, q \neq 0, r$
Initial conditions	$start(\pi_i^{0,1}) = t - g$; $end(\pi_i^{0,1}) \geq t$	

that length of each fragment of a cluster is constrained by the length of the cluster. The second ensures that each cluster completely leaves the intersection in all cycles.

Fragment departure: There are three constraints related to fragment departure. First, outgoing clusters can be scheduled to exit the intersection only after they arrive at the stop line: Second and third constraints ensure that if fragment $O_{i,q}^{k,r}(\xi)$ is scheduled, it is scheduled in the appropriate phase $\pi_i^{k,r}$ with respect to its start and end times.

Cluster precedence: A cluster can be scheduled only after the previous cluster in the same phase has completely exited the intersection. This is achieved through two constraints.

Initial conditions: The current phase ψ^0 at decision time t determines the feasibility of extension of the current signal timing plan. We formulate this by constraining the start and end times of ψ^0 in the current cycle $r = 1$.

Objective: We compute a single signal timing plan π_i that minimizes the cumulative waiting time across the vehicles in all the inflow samples using 4.

$$\min \sum_{\xi, k, q, r} \left[\begin{array}{l} \text{Delay incurred by cluster fragment } O_{i,q}^{k,r}(\xi) \\ \overbrace{\left(start(O_{i,q}^{k,r}(\xi)) - a_{i,q}^k(\xi) \right)} \\ |c_{i,q}^k(\xi)| \times \frac{\text{size}(O_{i,q}^{k,r}(\xi))}{\underbrace{l_{i,q}^k(\xi)}} \\ \text{Number of vehicles in fragment } O_{i,q}^{k,r}(\xi) \end{array} \right] \quad (4)$$

Sample-Based Communication

Our approach to signal control involves intersections independently receiving non-local traffic information to extend their observation horizons, computing signal timing plans for traffic in these extended horizons, and in turn communicating projected traffic outflow samples to neighbouring intersections to expand their observation horizons. Limit-

ing communication to immediate neighbours makes the approach scalable to large road networks. At each intersection i , our approach involves six key steps:

(i) Receive samples of non-local vehicle arrival times along each incoming edge; (ii) Append each sample of non-local vehicle arrival times to the estimated arrival distribution of locally observed vehicles to create *extended observations*; (iii) Sample exit edges for vehicles in each *extended observation* and cluster them by proximity to create samples of incoming clusters Ξ_i ; (iv) Compute a delay-minimizing signal timing plan π_i using Ξ_i and operating constraints, λ_i ; (v) Estimate vehicle departure times for each sample $\xi \in \Xi_i$ using π_i ; (vi) Communicate these departure time samples to the appropriate neighbours N_i along the sampled exit edges.

Experimental Setup

We implement and evaluate two versions of TUSERACT – an *uncoordinated* TUSERACT version *without* communication between neighbouring intersections (UTuS) and a *coordinated* TUSERACT version *with* communication between neighbouring intersections (CTuS). We use our implementations of two versions of SURTRAC as baselines – *uncoordinated* SURTRAC *without* communication among neighbours (USUR) (Xie et al. 2012) and *coordinated* SURTRAC *with* communication between neighbours (CSUR) (Xie, Smith, and Barlow 2012).

We evaluate our approach on a variety of synthetic road networks adapted from the literature (Xie et al. 2012; Xie, Smith, and Barlow 2012) in increasing order of network complexity: (1) isolated intersection; (2) arterial network; and (3) 5x5 grid network. We modify the original networks to allow vehicle turning. We vary road lengths (observation horizon) and phase designs across the synthetic networks in order to evaluate our approach in a variety of scenarios (Table 2). We also evaluate our approach by simulating traffic movements on a real road network in Pittsburgh adapted from (Smith et al. 2013). We import a free street-level map from OpenStreetMap and modify it to obtain the nine-intersection network used in the paper. We keep the road lengths (observation horizon) and phase design un-

Table 2: Configurations of traffic networks used for performance evaluation. Network maps (Figures 3a, 4a, 5a, 6a) indicate road lengths, allowed turns and phase designs.

	Isolated	1x5 grid	5x5 grid	Real Network
Phase designs	4-phase	3-phase, 2-phase	4-phase, 2-phase	4-phase
Road lengths (m)	300	250	25, 75, 150	Between 36.66 and 168.60 with an average of 82.37
Observation horizon (sec)	30	25	2.5, 7.5, 15	Between 3.66 and 16.86 with an average of 8.23
Turn probabilities (through, left,right)	{0.6-0.2-0.2}	{0.65-0.15-0.20, 1-0-0, 1-0-0, 0.8-0-0.2}	{0.65-0.15-0.20, 0.8-0.2-0}	{0-0.8-0.2, 0.2-0.8-0, 0.2-0.6-0.2, 1-0-0, 0-1-0}
Demand levels (vph)	{900, 1350, 1800}	{900, 1200, 1500}	{4000, 5000, 6000}	{5200, 7800, 9600}
Horizon extension (sec)	-	20	15	15

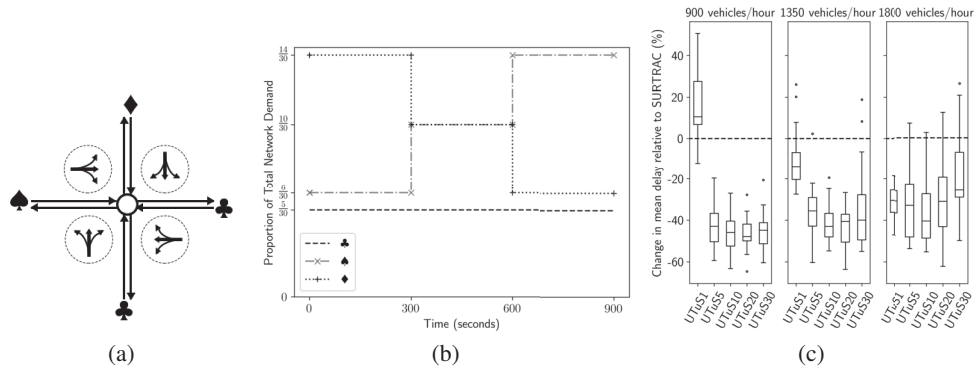


Figure 3: Isolated Intersection: (a) Setup; (b) Demand flow profile; and (c) Performance gain achieved by UTuS over USUR for various sample counts and demand levels

changed across the network to simulate real world conditions. We allow vehicle turns according to the routes used in the performance analysis in (Smith et al. 2013). Since our approach is decentralized and communication occurs only between immediate neighbours, adding intersections to the network without an increase in width (maximum number of neighbouring nodes) does affect the performance of our approach. We have considered the worst-case real-world scenario of a maximum of four neighbours in our 5x5 network.

We use the IBM ILOG CP Optimizer (single thread, 5s solver compute time limit) to solve the sample-based constraint program at every decision point. Mean vehicle delay or waiting time is used as an indicator of solution quality. Similar to previous work, all simulations are run on SUMO (Behrisch et al. 2011), an open source traffic simulation package. For each network, we define a traffic demand profile that specifies how the total traffic demand is distributed over the input edges in the network over time. For each network, we run simulations on low, medium, and high traffic demand levels (in vehicles/hour or vph, 20 test instances per demand level unless specified). Traffic is gen-

erated for 15 minutes and the problem horizon extended till all vehicles have cleared the network as is done by Guilliard et al. (2016b). We evaluate TUSERACT on each test instance with 5 independent and identically distributed sample sets. For each sample set, we run TUSERACT with {1, 5, 10, 20, 30} samples. This amounts to 100 runs of TUSERACT for each demand level and sample count. We generate samples offline to study the effect of adding additional samples to an existing sample set. In practice, vehicle turns are sampled online at each decision step.

Following Xie et al. (2012), we assume that vehicles travel at the constant speed of 10 m/s and queued vehicles are discharged after a startup lost time of 3.5s. All vehicles are 5m long. Vehicles arriving at the intersection within the same second are clustered together and a threshold of 3s is used to further aggregate clusters by proximity. For each phase, we set G^{min} to 5s, G^{max} to 55s and Y to 5s. The time resolution used by our implementation of SURTRAC is 0.5s while that used by TUSERACT is 1s. An optimization horizon of 3 cycles is used to limit schedules computed by TUSERACT. Our solver is limited to taking 5 seconds

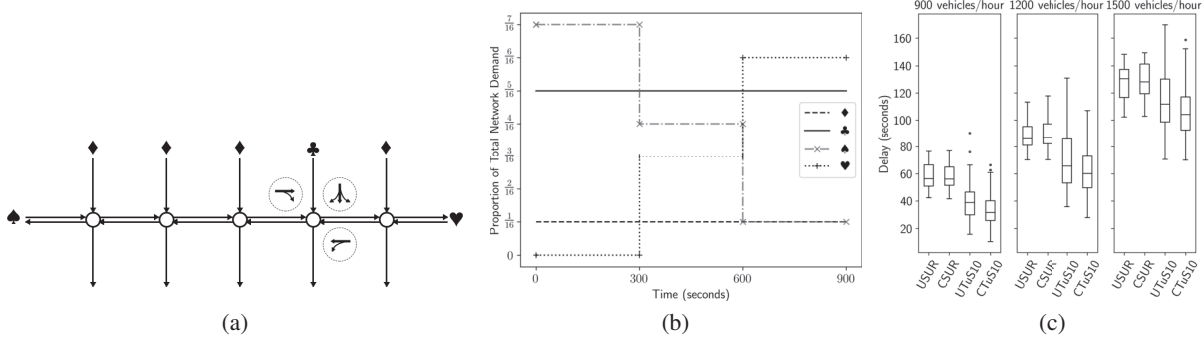


Figure 4: Arterial Network: (a) Setup; (b) Demand flow profile; and (c) Absolute delays for coordinated and uncoordinated SUR and TuS10 for various demand levels.

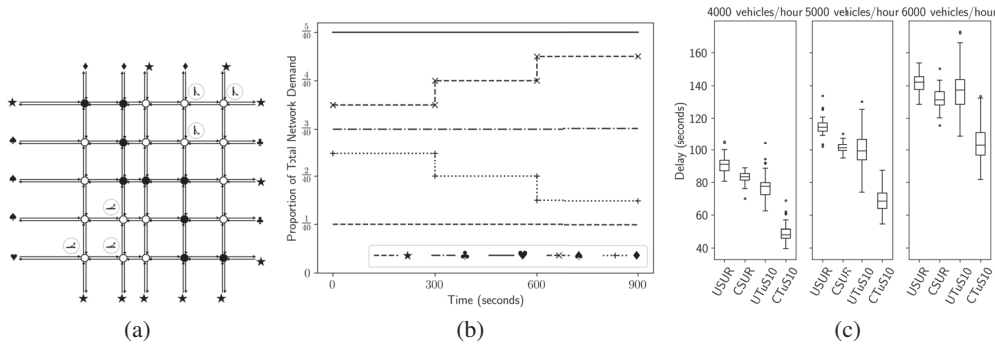


Figure 5: 5x5 Grid Network: (a) Setup: Shaded circles represent 4-phase intersections where through, left and right turns are permissible on all roads; other intersections are 2-phase intersections; and additional turn movements are shown alongside the respective roads; (b) Demand flow profile; and (c) Absolute delays for coordinated and uncoordinated SUR and TuS10 for various demand levels.

which is slightly more than SURTRAC. We can improve this by trading off samples for an increase in delay and by using commercial processors. For both approaches and all simulations:

- Vehicles are detected exactly along the full observation horizon without detection error.
- Turn proportions remain static and are known exactly by each intersection. In practice, these are estimated online as moving averages but this would not require any modification to our approach.
- Planning and communication times are counted outside the simulation. We make this simplifying assumption to focus solely on the solution quality of the two approaches. We acknowledge the trade-off between solution quality and real-time tractability in the context of both the approaches, but do not aim to address it in this paper. Here, we investigate whether a sampling-based approach can reduce waiting times when planning under turn uncertainty.

To ensure that non-sampling-related enhancements (cluster division, maximum green time constraint) do not provide a significant performance improvement by themselves, we ran

TUSERACT with a single expected traffic sample as used by SURTRAC for varying traffic conditions on an isolated intersection. Non-sampling enhancements did not provide a significant advantage (at best 7.77%) and in low-traffic scenarios gave worse results than SURTRAC.

Experimental Results

Isolated Intersection (Figure 3). Figure 3c shows the percentage change in mean waiting time of UTUS relative to USUR for our three demand levels of 900, 1350, and 1800 vehicles per hour (vph). For each demand level, the figure shows the improvements of UTUS with varying number of samples. We make the following observations:

- UTuS results in significantly lower (40% on average) mean vehicle waiting times with respect to USUR across all demand levels.
- The number of samples required to produce this substantial improvement is small. Using 5 or more samples consistently provides a 35-45% reduction in delay across all demand levels.
- Increasing the sample count beyond 10 results in a perfor-

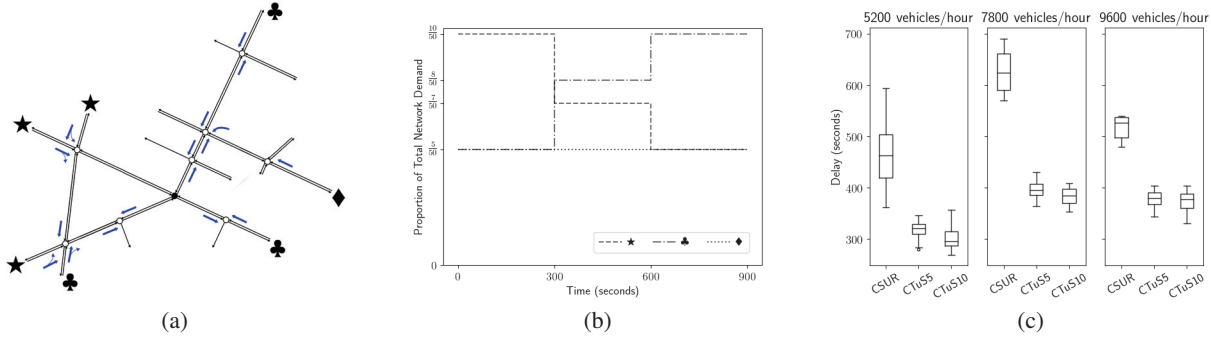


Figure 6: Real-world Network: (a) Road network with 9 intersections in the East Liberty neighbourhood of Pittsburgh, Pennsylvania. Setup: All intersections are 4-phase; for shaded intersection: through, left and right turns are permissible; for other intersections: allowed turn movements are shown alongside the respective roads; (b) Demand flow profile; and (c) Absolute delays for coordinated SUR, TuS5 and TuS10 for various demand levels. Each boxplot includes data from 10 independent test instances.

mance drop for higher demand levels due to the tougher computational challenge posed to UTuS. To verify that this degradation is due to insufficient compute time, we ran simulations with a solver time limit of 30 seconds. The increased compute time indeed results in higher performance gains ($\sim 45\%$) even for high demand (1800 vph) and high sample counts $\{20, 30\}$.

Arterial Network (Figure 4). This network tests our approach in a scenario with low uncertainties. To allow comparison between coordinated and uncoordinated versions of SURTRAC and TUSERACT we plot absolute delays. For brevity, we report results for TUSERACT with 10 samples only. We observe that TUSERACT is able to perform at par with SURTRAC and even reduce mean waiting times (by 10-40%) in scenarios with low uncertainty. However, for both approaches, coordinated approaches do not significantly outperform uncoordinated approaches. This could be due to the fact that: (i) The observation horizon is sufficiently long and non-local information does not provide significant advantage during planning, and (ii) the bottleneck intersection causes queue spillover that propagates to neighbouring intersections. Xie, Smith, and Barlow (2012) proposes a spillover prevention strategy, which we plan to explore as part of future work.

5x5 Grid Network (Figure 5). Since the intersections are closely placed (as in a typical urban network), this scenario represents a situation that could benefit from coordinated planning. At 10 samples, UTuS provides a 7-17% reduction in delay over USUR and CTuS provides a 19-40% delay over CSUR. We also observe that the performance gains can be significantly improved (up to a consistent 40-50%) by using a previously computed signal timing plan as a starting solution to guide CP search at the current decision step (additional details in the supplementary material). We note that the coordinated approaches significantly reduce delays relative to uncoordinated approaches as any observation beyond the short local observation horizon is advantageous. In experiments with non-local observation, we note that in-

creasing the non-local observation length from 5s to 30s for TUSERACT can reduce delays by as much as 20%.

Real-world Scenario (Figure 6). The road network is a triangular grid with nine intersections. The three major roads have bidirectional traffic flow along with incoming traffic from minor roads. Distances between the intersections range between 36.66 to 168.60 meters, with an average of 82.37 meters, requiring tight coordination between the intersections. The demand levels are selected to simulate non-congested, moderately-congested and highly-congested traffic conditions. At 5 and 10 samples we observe that CTuS reduces mean waiting times by 27-39% over CSUR, while UTuS provides 16-20% less delay over USUR. To allow comparison between CSUR and CTuS we plot the absolute delays. We focus on the coordinated versions of SURTRAC and TUSERACT, since the coordination between neighbouring intersection reduces delay.

Conclusions

In this paper, we propose a sampling-based approach to traffic signal control in the presence of vehicle turn-induced uncertainty. We show experimentally that our approach provides significant reductions in delay over SURTRAC, which makes an approximation to solve the turn-induced stochastic optimization problem in order to remain real-time tractable. This performance improvement can be achieved with very few samples (≤ 10) and in reasonable time (5s). We did a thorough set of experiments to evaluate the performance of TUSERACT on low, medium and high complexity networks over numerous parameter settings. Our results show that sampling is a promising approach to tackle uncertainty in this domain and we obtained up to 60% reduction in delay over the leading approach for online traffic signal control.

References

Bazzan, A. L. 2005. A distributed approach for coordination of traffic signal agents. *Autonomous Agents and Multi-Agent Systems* 10(2):131–164.

- Behrisch, M.; Bieker, L.; Erdmann, J.; and Krajzewicz, D. 2011. Sumo—simulation of urban mobility. In *The Third International Conference on Advances in System Simulation (SIMUL 2011), Barcelona, Spain*, volume 42.
- Cai, C.; Hengst, B.; Ye, G.; Huang, E.; Wang, Y.; Aydos, C.; and Geers, G. 2009. On the performance of adaptive traffic signal control. In *Proceedings of the Second International Workshop on Computational Transportation Science*, 37–42. ACM.
- Chin, S.-M.; Franzese, O.; Greene, D. L.; Hwang, H.-L.; Gibson, R.; et al. 2004. Temporary losses of highway capacity and impacts on performance: Phase 2. Technical report, United States. Dept. of Energy. Office of Scientific and Technical Information.
- Gartner, N. H.; Pooran, F. J.; and Andrews, C. M. 2002. Optimized policies for adaptive control strategy in real-time traffic adaptive control systems: Implementation and field testing. *Transportation Research Record* 1811(1):148–156.
- Goldstein, R., and Smith, S. F. 2018. Expressive real-time intersection scheduling. In *Thirty-second AAAI conference on artificial intelligence*.
- Guilliard, I.; Sanner, S.; Trevizan, F.; and Williams, B. 2016a. A non-homogenous time mixed integer lp formulation for traffic signal control. *Transport Research Record (TRR): Journal of the Transport Research Board* 2525. Recipient of the Kikuchi-Karlaftis Best Paper Award.
- Guilliard, I.; Sanner, S.; Trevizan, F. W.; and Williams, B. C. 2016b. A non-homogeneous time mixed integer lp formulation for traffic signal control. In *Presented at the 95th Annual Meeting of the Transportation Research Board, Washington, DC*.
- Hu, H.-C., and Smith, S. F. 2018. Bi-directional information exchange in decentralized schedule-driven traffic control. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 1962–1964. International Foundation for Autonomous Agents and Multiagent Systems.
- Kleywegt, A. J.; Shapiro, A.; and Homem-de Mello, T. 2002. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization* 12(2):479–502.
- Kuyer, L.; Whiteson, S.; Bakker, B.; and Vlassis, N. 2008. Multiagent reinforcement learning for urban traffic control using coordination graphs. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 656–671. Springer.
- Li, L.; Lv, Y.; and Wang, F.-Y. 2016. Traffic signal timing via deep reinforcement learning. *IEEE/CAA Journal of Automatica Sinica* 3(3):247–254.
- Luyanda, F.; Gettman, D.; Head, L.; Shelby, S.; Bullock, D.; and Mirchandani, P. 2003. Acs-lite algorithmic architecture: applying adaptive control system technology to closed-loop traffic signal control systems. *Transportation Research Record* 1856(1):175–184.
- Mousavi, S. S.; Schukat, M.; and Howley, E. 2017. Traffic light control using deep policy-gradient and value-function-based reinforcement learning. *IET Intelligent Transport Systems* 11(7):417–423.
- Prabuchandran, K.; AN, H. K.; and Bhatnagar, S. 2014. Multi-agent reinforcement learning for traffic signal control. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2529–2534. IEEE.
- Prashanth, L., and Bhatnagar, S. 2010. Reinforcement learning with function approximation for traffic signal control. *IEEE Transactions on Intelligent Transportation Systems* 12(2):412–421.
- Robertson, D. I., and Bretherton, R. D. 1991. Optimizing networks of traffic signals in real time—the scoot method. *IEEE Transactions on vehicular technology* 40(1):11–15.
- Sen, S., and Head, K. L. 1997. Controlled optimization of phases at an intersection. *Transportation science* 31(1):5–17.
- Shabestary, S. M. A., and Abdulhai, B. 2018. Deep learning vs. discrete reinforcement learning for adaptive traffic signal control. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 286–293. IEEE.
- Sims, A. G., and Dobinson, K. W. 1980. The sydney coordinated adaptive traffic (scat) system philosophy and benefits. *IEEE Transactions on vehicular technology* 29(2):130–137.
- Smith, S.; Barlow, G.; Xie, X.-F.; and Rubinstein, Z. 2013. Surtrac: Scalable urban traffic control. In *Proceedings of Transportation Research Board 92nd Annual Meeting Compendium of Papers*. Transportation Research Board.
- Van der Pol, E., and Oliehoek, F. A. 2016. Coordinated deep reinforcement learners for traffic light control. *Proceedings of Learning, Inference and Control of Multi-Agent Systems (at NIPS 2016)*.
- Wei, H.; Zheng, G.; Yao, H.; and Li, Z. 2018. Intellilight: A reinforcement learning approach for intelligent traffic light control. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2496–2505. ACM.
- Xie, X.-F.; Smith, S. F.; Lu, L.; and Barlow, G. J. 2012. Schedule-driven intersection control. *Transportation Research Part C: Emerging Technologies* 24:168–189.
- Xie, X.-F.; Smith, S. F.; Barlow, G. J.; and Chen, T.-W. 2014. Coping with real-world challenges in real-time urban traffic control. In *Compendium of Papers of the 93rd Annual Meeting of the Transportation Research Board*, 1–15.
- Xie, X.-F.; Smith, S. F.; and Barlow, G. J. 2012. Schedule-driven coordination for real-time traffic network control. In *ICAPS*.
- Yu, X.-H., and Recker, W. W. 2006. Stochastic adaptive control model for traffic signal systems. *Transportation Research Part C: Emerging Technologies* 14(4):263–282.