An Empirical Investigation of Adaptive Traffic Control Parameters

Jeffery Raphael[1] and **Elizabeth I. Sklar**[2] and **Simon Maskell**[3]

[1]Dept of Computer Science, University of Liverpool, UK
[2]Dept of Informatics, King's College London, UK
[3]Dept of Electrical Engineering and Electronics, University of Liverpool, UK
jeffery.raphael@liverpool.ac.uk,elizabeth.sklar@kcl.ac.uk,s.maskell@liverpool.ac.uk

Abstract

The goal of *adaptive traffic management* is to adjust the timing of traffic signals at intersections in order to dynamically adapt, in real time, to traffic conditions. The SCOOT system, a commercial product widely deployed around the world, focuses on adjusting three traffic signal control parameters: *split*, *cycle* and *offset*. By responding to data collected from sensors embedded in roadways, SCOOT can effectively adjust to expected fluctuations in traffic, such as those that occur regularly during commuting hours. However, SCOOT does not perform optimally when there are unexpected disruptions in traffic flow, such as after the occurrence of an accident or during events that cause traffic conditions to deviate from the norm. The work presented here outlines an empirical study of the three SCOOT parameters, comparing the adjustment algorithm employed by SCOOT to a number of different adaptive methodologies, including two novel schemes. Experimental results, analysed across a range of different traffic flows, demonstrate that the novel methods perform as well as SCOOT under normal conditions and better under disruptive conditions.

1 Introduction

The notion of *adaptive traffic management* has been considered in a range of fields, from traffic control engineering to intelligent systems science. The goal is to maximise the throughput of vehicles across networks of roadways: reducing travel times for individuals, minimising wait times at intersections and avoiding collisions. There are a number of desirable subgoals, such as reducing the amount of pollution created by decreasing travel times, lowering petrol costs by shortening idle times and diminishing stress on commuters.

Within the *multi-agent systems (*MAS*)* community, a popular approach is to represent each vehicle as an *autonomous agent* and employ mechanisms that require the vehicles to negotiate with each other [Carlino *et al.*, 2013; Dresner and Stone, 2004; Vasirani and Ossowski, 2012]. However, widespread deployment of *autonomous vehicles* in real-world environments is not a near-term reality. There are many challenges that remain before self-driving cars will be used by the masses. First, there is the development and deployment of the cars themselves. Google's self-driving cars are widely talked about, with a fleet of autonomous cars that have collectively covered over $700K$ miles [Gomes, 2014]. Yet, these cars navigate using special maps that have enhanced information, such as location of traffic signals and driveways. As well, they cannot avoid unmarked potholes and would not be able to obey commands from a traffic officer [Gomes, 2014]. Second, there is the current state of connectivity. The communication infrastructure necessary for broad vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) currently does not exist. In the USA, the National Highway Traffic Safety Administration (NHTSA) is currently pushing for the use of V2V technology nationwide, arguing that it could dramatically reduce accidents by warning of dangers ahead; but, to date, there is no nationwide agreement or timeline for implementation. It is estimated that self-driving cars will not completely supercede human-driven cars until at least the year 2040 [Litman, 2015; Shanker *et al.*, 2013].

Motivated by these practical contraints, our work does not rely on the presence of autonomous vehicles—instead, we focus on adaptive solutions to traffic problems that can be deployed within today's infrastructure. An *intersection* is a prominent feature of existing infrastructure, where roads cross each other and the need to coordinate access to the intersection is vital for preventing collisions. The task of *intersection management* is primarily achieved using traffic signals, familiar artifacts that are well integrated into road infrastructures world-wide[1]. Traditionally, intersection management by traffic signals is implemented as *fixed* periods of green, amber and red lights. In an effort to improve on the performance of fixed traffic signals, *adaptive Urban Traffic Controllers* (UTCs) have been developed and deployed in many cities around the world [Wang, 2005; Mladenovic and Abbas, 2013; Papageorgiou *et al.*, 2003]. Adaptive UTCs use information about current road conditions and determine, some in real-time, the best signal settings. These systems attempt to harmonise the interplay between all aspects of traffic (private cars, public transporta-

---

[1]In some countries, another common feature of the infrastructure is a *roundabout* (also called a *rotary* or *traffic circle*); but these are controlled through norms and driver behaviour, and do not fall into the category of technologies that are controlled through infrastructure external to the driver, which is what we consider here.
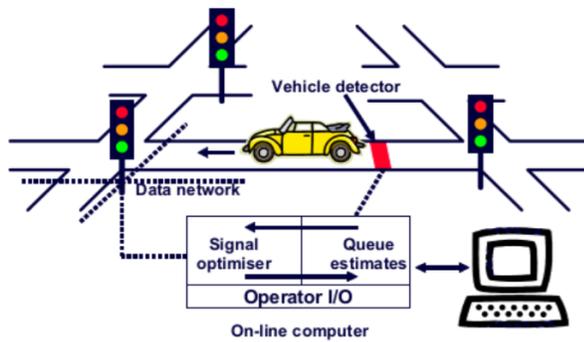
Figure 1: The SCOOT Model [Limited, 2016]. The red stripe across the road behind the yellow car in the figure illustrates an induction-loop road sensor.

tion, cyclists and pedestrians) in areas ranging in size from a few city blocks to entire cities. The majority of adaptive UTCs employ optimisation algorithms which are costly to develop, calibrate, maintain and expand [Wang, 2005]. Examples of deployed UTCs include: SCOOT[2] [Hunt *et al.*, 1981], RHODES [Mirchandani and Wang, 2005] and OPAC [Gartner *et al.*, 2001]. We focus on SCOOT because it is a popular system, it is deployed in our local city and we have access to data for modelling. The remainder of this paper is organised as follows. Section 2 describes how SCOOT works. Our approach is presented in Section 3 and experiment design in Section 4. Our results are presented in Section 5 and discussed in Section 6. Section 7 reviews other adaptive approaches to traffic control, and Section 8 closes with a summary and directions for future research.

2 SCOOT

SCOOT *(Split, Cycle and Offset Optimisation Technique)* is a centralised, real-time system that minimises delay and prevents congestion by coordinating small sets of traffic signals, called *regions*. The intersections within a region always form a linear path, i.e., signal timings are optimised to improve traffic flow in a single direction. SCOOT responds to data collected from *induction-loop sensors* embedded in roadways (Figure 1), which are simple counter devices that trigger when vehicles drive over them. Using the sensor data, SCOOT responds effectively to expected fluctuations in traffic.

Traffic can flow into an intersection from multiple directions, each of which is called a *link*. The *degree of saturation* of an intersection is a measure of its level of use, i.e., the amount of traffic demand compared to its maximum capacity. The traffic signal has a *phase* for each link which sequences through a period of *green time*, followed by a period *red time*[3]. SCOOT adjusts three traffic signal control parameters, as follows:

---

[2]http://www.scoot-utc.com

[3]Typically, red time is preceded by a short period of amber (yellow) time; in some countries, green time is preceded by a short period of joint amber and red time.

- **split**—The amount of green time allocated to each individual link is called *split*. Five seconds before a phase change, SCOOT considers the effect on the degree of saturation caused by *advancing* (terminating the phase), *retarding* (extending the phase) or *holding* (allowing the phase to continue to termination). SCOOT selects the option that reduces the degree of saturation the most. The split is adjusted in increments/decrements of 4 seconds.

- **cycle**—*Cycle length* is the total amount of time it takes for every link to receive its complement of green time. SCOOT optimises cycle length by examining the roadway with the highest degree of saturation. If that is greater than 90%, then the cycle length (for the entire region) is increased. SCOOT decreases the cycle length if every roadway entering the intersection has a degree of saturation greater than 90%. Cycle length changes are made in increments (or decrements) of 4, 8, 16, and 32 seconds (the shorter the cycle, the smaller the change) [Halkias, 1997].

- **offset**—A *green wave* is a phenomenon that occurs when a vehicle crosses many intersections in a row and all the traffic signals show green, so the vehicle does not have to stop at each intersection. In order for a green wave to occur, the traffic signals at adjacent intersections in a given path must be synchronised. The *offset* parameter represents the difference between the start of green time at two consecutive intersections. SCOOT checks the offset once at the end of every cycle and attempts to minimise the number stops required per vehicle by adjusting the offset in increments/decrements of 4 seconds.

Although SCOOT responds well to expected changes, such as regular increases in directional traffic flows during commuting times, SCOOT does not perform optimally when there are *unexpected disruptions* in traffic flow, such as when there are accidents or entertainment events that suddenly cause patterns to deviate from the norm. In our work, we have developed a set of traffic patterns that test the efficacy of SCOOT under different conditions. We use these patterns to compare several different parameter adjustment policies to the SCOOT benchmark, including two novel schemes that take a market-based approach. Experimental results, analysed across different traffic flows, demonstrate that our novel methods perform as well as SCOOT under normal conditions and better than SCOOT under disruptive conditions.

3 Our Approach

Our approach to traffic control parameter optimisation considers the three SCOOT parameters described above. In order to tune these parameters for real-time traffic control, we address a number of questions: *Which parameters should be adjusted? When should the parameters be adjusted? What data is used to inform an adjustment?* and *How should the parameters be adjusted?*

Our approach to traffic control revolves around the notion that traffic control is a *coordination* problem where intersections work together to minimise delay. Thus, we decompose the intersection into a *multi-agent system* and utilise an

auction-based approach to facilitate coordination amongst its agents. Our approach shares some similarities with SCOOT: it manages traffic flow using the same three parameters (cycle, split and offset), uses degree of saturation to measure road usage and uses transportation technology (vehicle detectors) that is currently available. However, our approach has a many significant differences. Adjustments to the traffic control parameters are made periodically and intersections are not clustered into fixed, pre-defined regions. Without these restrictions, our approach allows our mechanism to function on a much larger scale than SCOOT.

We first experimented with the idea of intersections as agents, informed in real-time by road sensors, in Raphael *et al.* [2015], where we presented our SAT mechanism. Here we expand upon that work in several ways. First, we present two new strategies for the behaviour of our traffic control agents. Second, we present experimental results that demonstrate the robustness of our approach in the face of unexpected disruptions in traffic flow. Finally, we compare our approach with a broad set of alternate strategies.

In both approaches, we use an *intersection agent* as an auction manager and *traffic signal agents* that represent the traffic signal phases. A phase represents multiple traffic streams. A single phase can service multiple vehicle manoeuvres. For example, the first phase of a traffic signal may allow through traffic and left turns. We use a two-phase signal plan: one light phase for north/south-bound traffic and the other phase for west/east-bound traffic. Thus, at every intersection, there is an intersection agent working in concert with two traffic signal agents. Our traffic signal control mechanism employs a first-price, single-item auction. As traffic flows through an intersection, auctions take place at fixed intervals[4]. The traffic signal agents bid against each other; the winner is the agent with the highest bid. The winning agent then makes a single adjustment to its traffic signal timing.

3.1 GRACE

Our initial investigation into traffic control mechanisms [Raphael *et al.*, 2015] was limited in its ability to react to changing traffic conditions because only green time was adjusted (in 5-second segments). Our new method presented here, **G**ene**R**al Purpose **A**uction-based Traffic **C**ontroll**E**r (**GRACE**), allows traffic signal agents to change all three variables. Adjustments are made in discrete steps, s (measured in seconds), defined as:

$$s = \langle \Delta green\_time, \Delta offset, \Delta cycle\_length \rangle$$

For example, if $s = \langle 3, -4, 10 \rangle$, then the green time would be increased by $3s$, the offset reduced by $4s$ and the cycle length increased by $10s$. A finite set of possible adjustment values is defined, specific to each mechanism (see below).

In [Raphael *et al.*, 2015], we measured the level of use of a roadway by calculating *saturation*, the ratio of the volume

---

[4]The optimal length of the fixed interval varies with each mechanism, and the values we use in our work were determined experimentally. Detailed discussion of these results is beyond the scope of this paper, but can be found in our technical reports.

of traffic (as measured by road sensors) to its estimated maximum capacity. However, this ratio does not quantify how a change to green time (or cycle length) effects the level of use in a lane(s), so GRACE uses *degree of saturation* [Lee *et al.*, 2002; Roess *et al.*, 2009], X, which is defined as:

$$X = \frac{v}{c} * \frac{L}{g} \qquad (1)$$

where: v is the volume of traffic read by the traffic signal agent; c is the maximum possible volume of traffic (in vehicles per hour); L is cycle length; and g is green time. Traffic signal agents in GRACE are characterised by their utility function and their bidding rule. Next, we present two different GRACE-based traffic signal agents: DCF and MMDOS.

3.2 DCF

In *Dynamic Coalition Formation (*DCF*)*, traffic signal agents find the best offset to reduce the number of vehicles that will have to stop for the red light and a green time that will minimise the maximum degree of saturation. At an intersection, each lane of traffic flow may have a different degree of saturation. DCF attempts to minimise the degree of saturation of the lane experiencing the highest level of use. The utility of adjustment s is given by:

$$U(s) = -[X + D(s)] \qquad (2)$$

where the values for the degree of saturation X_t and estimated number of stopped vehicles $D(s)$ reflect the adoption of adjustment s. The bidding rule for DCF is:

$$b = X \qquad (3)$$

The possible adjustment values for DCF are: $\Delta green\_time \in \{0 \dots 5\}$, $\Delta offset \in \{-4, 0, 4\}$, and $\Delta cycle\_length = 0$ (i.e., cycle length does not change).

3.3 MMDOS

In *Minimise Maximum Degree Of Saturation (*MMDOS*)*, traffic signal agents minimise the *degree of saturation* of the lane experiencing the highest level of use. The utility of adjustment s in MMDOS is given by:

$$U(s) = -[X] \qquad (4)$$

The biddings rule for MMDOS is:

$$b = X + u \qquad (5)$$

where u is the length of the queue of cars on the roadway associated with the phase under the agent's control. The possible adjustment values for MMDOS are: $\Delta green\_time \in \{1 \dots 5\}$, $\Delta offset = 0$, and $\Delta cycle\_length = 0$ (i.e., offset and cycle length do not change).

4 Experiments

We evaluated GRACE in a simulated 5×5 grid-based city plan (Figure 2). Our traffic control experiments were conducted on *Simulation of Urban MObility* (SUMO) [Krajzewicz *et al.*, 2012], an open source microscopic traffic simulator. All traffic signals used a two-phase signal plan: during one
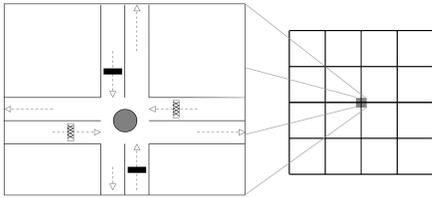
Figure 2: Grid-based city plan with intersection layout.

phase, north/south bound traffic passed through the intersection, while west/east bound traffic passed in the other phase. The signal plan did not include dedicated turning (right or left) phases, therefore left and right turns were given lower priority than through movements, i.e., vehicles turning left or right waited until it is safe to do so. All the roads were fitted with road sensors to collect traffic volume data. Also, the four corner traffic signals were disabled because there were no conflicting traffic movements at those intersections. Thus, in our experiments, GRACE adaptively controls twenty-one of the intersections.

4.1 Traffic Conditions

For the experiments described here, we utilised three different traffic scenarios to evaluate the performance of our market-based mechanism. The scenarios employed sudden increases in traffic volume (or *intensity*) to disrupt traffic flow. The final scenario replicated traffic conditions that may occur during a sporting event. The scenarios are:

- **Structured** is traffic that flows through the network with an identifiable (e.g., commuter) path with heavy flow;

- **Unstructured** is traffic flow with no identifiable path with heavy flow; and

- **Football** emulated traffic conditions before, during and after a football match. The traffic flow represented a worst-case scenario where there is a sudden sharp increase in traffic demand. There are two disruptions: first, fans enter the area of the arena (30 minutes after the simulation started); and second, fans exit the arena (approximately 90 minutes later).

We raised the intensity of traffic at the one-hour mark during Structured and Unstructured traffic conditions. Structured represents the traffic pattern that is ideal for an adaptive urban controller such as SCOOT. Each set of experimental conditions were repeated 30 times to attain suitable statistics.

We evaluated the performance of the traffic controllers using the metric *travel time*. Travel time is by far the most common way of measuring the effectiveness of traffic controllers. We examined *travel time* in several different forms. First, we looked at the average travel time of all the vehicles across the 30 simulations. Second, we collected data on the average travel time of vehicles as they finished their journey at each time step. We compare the performance of our market-based controller to SCOOT (described in Section 1), fixed-time traffic signals (Section 4.2) and an auction-based traffic controller that learns a bidding strategy (Section 4.3).

4.2 Fixed-time Signals

We also implemented a fixed-time traffic signal controller, FXM. The fixed-time traffic signal controllers represented traditional, non-adaptive, traffic signal devices. In the case of fixed-time traffic signal controllers, all three traffic control parameters remain constant. The traffic signals displayed the same light sequences for the same duration every cycle. We chose to use the initial traffic signal timing settings used by the adaptive mechanisms as the settings for the fixed-time traffic signals. Thus, any differences in performances can be attributed to the adaptive nature of the controller (and not initial signal timings). The fixed-time traffic signals have a cycle length of 80 seconds, and 87.5% of that is allotted to the split.

4.3 Learning to Bid

We implemented a version of the auction-based traffic control mechanism of Mashayekhi and List [2015] in our SUMO traffic controller evaluation testbed. Of the three parameters adjusted by SCOOT, Mashayekhi and List modify only one, the split (green time). Their auction determines the amount of green time in a phase as well as the order of the phases. Mashayekhi and List used *Reinforcement Learning* (RL) to learn a bidding strategy. The only major difference between their implementation and ours was the number of *movement managers*. In their work, each movement manager was associated with a single stream of traffic. In our version, there were fewer movement managers because our test network did not have dedicated turning lanes. Furthermore, Mashayekhi and List did not specify an action space. Therefore, we discretised the bidding space to values $[0 \ldots 10]$ as our action space. That is, whenever an agent bids, its bid amount is some value between 0 and 10.

5 Results

	Average Travel Time *(std.)*		
	Traffic Pattern		
Policy	*Structured*	*Unstructured*	*Football*
SAT	160.22 (8.22)	623.64 (42.31)	150.66 (9.10)
MMDOS	169.50 (7.31)	652.09 (48.57)	137.36 (5.35)
DCF	158.37 (4.98)	**609.22 (32.80)**	**135.15 (4.84)**
FXM	165.93 (1.38)	927.47 (107.39)	184.34 (7.13)
SCOOT	**143.66 (4.85)**	1931.35 (225.81)	233.42 (9.42)
RL	302.82 (17.70)	1038.09 (266.38)	200.89 (10.59)

Table 1: Average travel time of vehicles under different methods of traffic control.

We simulated our three scenarios using six different traffic control methods: our earlier mechanism (SAT, from [Raphael *et al.*, 2015]), two new GRACE mechanisms (MMDOS and DCF), and three baselines: a fixed-time traffic signal (FXM), SCOOT, and the RL controller. In this section, we describe our

results, primarily the difference in performance of the controllers with *patterned* traffic (e.g., Structured traffic) versus *non-patterned* traffic (Unstructured and Football traffic).

Average travel times reflect time saved (or incurred) at intersections due to adequate traffic flow. With Unstructured and Football traffic, our market-based approaches outperformed all the other traffic controllers (Table 1). The worst performing mechanism from our approaches did better than FXM. DCF had the best overall average travel time in both the Unstructured and Football traffic. In Unstructured traffic, DCF reduced average travel time by 34.3% and 68%, compared to FXM and SCOOT, respectively. For the simulated football event, DCF reduced average travel time by 26.7% and 42%, compared to FXM and SCOOT, respectively. SCOOT had the worst performance with the two non-patterned traffic scenarios. With Unstructured traffic, SCOOT increased average travel time by over 100% and with the football match traffic it increased travel time by 26% (this is compared to FXM). However, SCOOT had the best performance with Structured traffic (the second best time was achieved by DCF). RL performed slightly worse than FXM with Unstructured and Football traffic; it increased travel time by nearly 10% in both cases.

Figure 3 provides a more detailed picture of travel time under SCOOT control versus our DCF controller. At each time step, as vehicles completed their journey, we captured their average travel time. With Unstructured traffic, SCOOT's travel time begins to increase even before the occurrence of the disruption at the $3600th$ second (Figure 3b). Under SCOOT, there is a sharp increase in travel times during the Unstructured disruption and it never recovers until the very end of the simulation. During the half-hour influx of drivers beginning at the $1800th$ second (Figure 3c), cars under DCF experienced significantly less delay than vehicles controlled by SCOOT. Immediately after the disruption ends, the average travel time peaks for both DCF and SCOOT, but SCOOT had the highest increase in average travel times. Both methods return to normal day-to-day travel times soon after the influx ends. Again, for the second disruption, starting at the $9000th$ second, traffic under SCOOT experienced far more delays than DCF. Although SCOOT did better than DCF in overall performance with Structure traffic, we find that there was significant overlap (Figure 3a) in travel times between vehicles under SCOOT control and vehicles controlled by DCF. In other words, there were many vehicles under DCF control that experienced travel time as short as those found in SCOOT. In Figures 3b and 4b, the SCOOT and RL simulations required more time steps than the other traffic controllers. The difference in the simulation horizon is due to how SUMO (the traffic simulator) works. SUMO does not terminate a simulation until all the vehicles that have been spawned complete their assigned trip. In all our simulations, the same number of vehicles were spawned but delay caused by the traffic controllers (e.g., SCOOT) resulted in a significant increase in the simulation horizon.

We also collected cumulative averages as the simulations ran (Figures 4). With Unstructured and Football traffic (Figures 4b and 4c), we see how quickly SCOOT's performance diverges from the market-based approaches. Our market-based
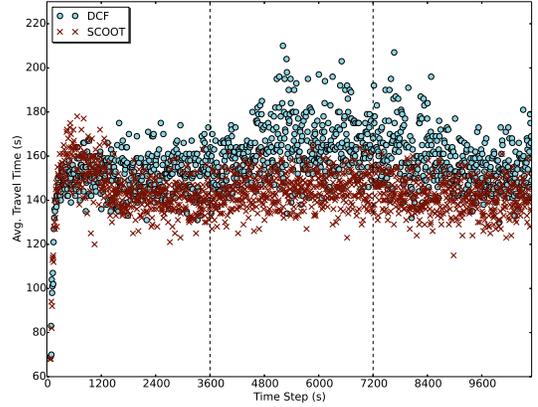
(a) Structured
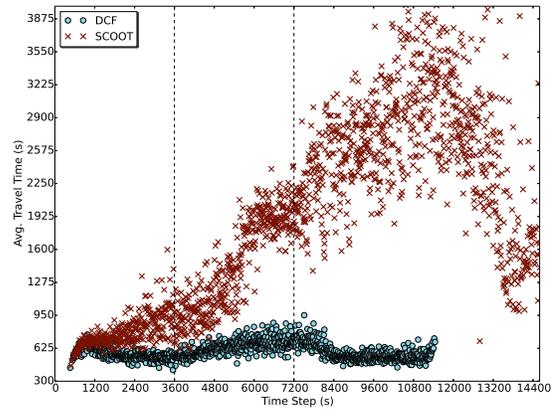
(b) Unstructured
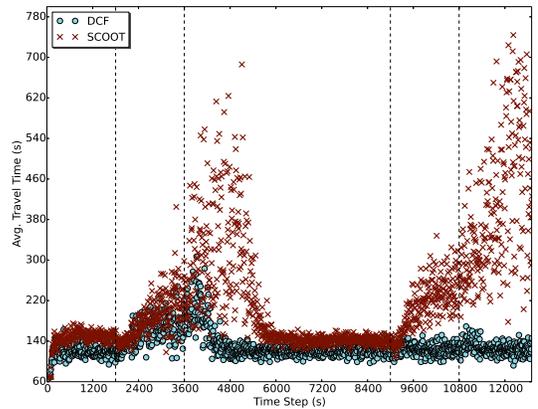
(c) Football

Figure 3: A comparison of average travel times of vehicles that have completed their journey at each time step. Beginning and end of disruptions are marked by dotted lines.
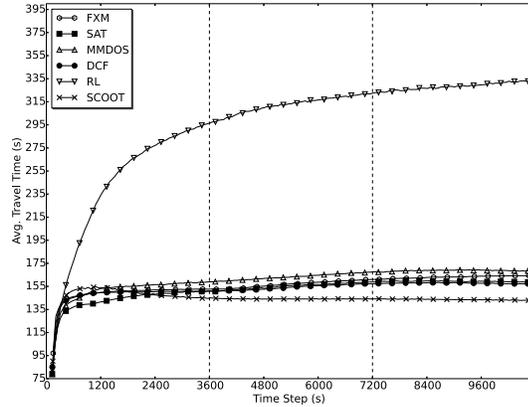
(a) Structured
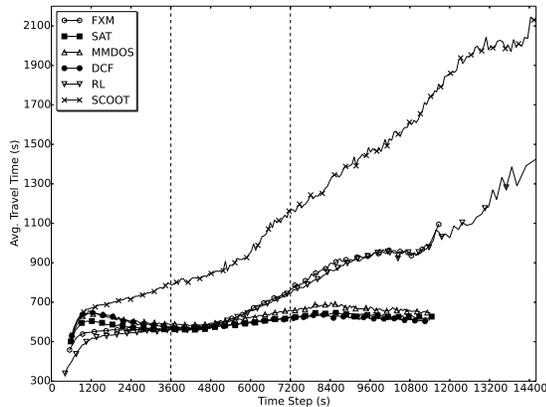
(b) Unstructured
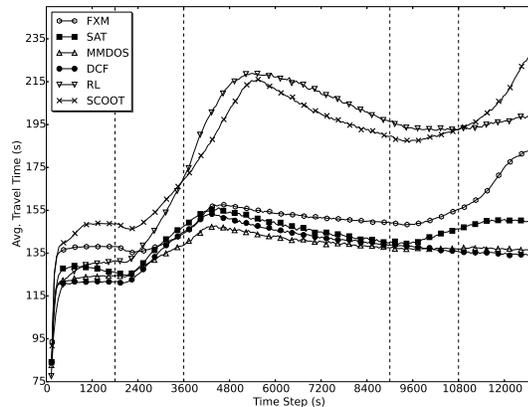
(c) Football

Figure 4: Cumulative average travel times. Beginning and end of disruptions are marked by dotted lines.

approaches did experience some increase in travel time during disruptions (e.g., the period from $1800th$ second to the

$3600th$ second in Figure 4c), but never peaked as high as SCOOT. With Structured traffic, the traffic scenario where SCOOT had the best performance, we find that our approach closely matched FXM (Figure 4a). RL had the worst performance under Structured traffic. In Figure 4a, we see that RL never showed any signs of adapting to the traffic demands. Also, in Unstructured traffic (Figure 4b) RL's performance closely mirrors FXM but in Football traffic (Figure 4c) it behaved more like SCOOT.

6 Discussion

Our results clearly demonstrate the dramatic effect traffic disruptions may have on the performance of SCOOT. Although our market-based approach utilises the same traffic parameters as SCOOT, we manipulate the split, offset and cycle time in a completely different manner. SCOOT is simply unable to satisfy the changing traffic demands and conflicting intersection manoeuvres (it is the latter that our approach excels at). SCOOT was designed to optimise the signal timing of small sets of traffic signals (that form a linear path). This severely restricts the ability of SCOOT to adapt to unexpected cross traffic. SCOOT performed well with traffic that had some established pattern of behaviour such as Structured, but could not cope with the Unstructured and Football scenarios. In Structured traffic (and the other scenarios like it) the scope of the control problem is more manageable than in other traffic scenarios.

RL did not perform as well as expected and our results did not resemble those found in [Mashayekhi and List, 2015]. There are a number of factors inherent to reinforcement-learning that could have contributed to its poor performance. For example, state space size (and representation) can affect learning, i.e., convergence to an optimal policy [Bakker *et al.*, 2010; Sutton and Barto, 1998].

Lastly, DCF and MMDOS represents our latest efforts to expand the capabilities of our market-based traffic controllers. One of the most important improvements to our approach is the new way in which it selects green time shifts. SAT can only make changes to green time in 5 second increments. DCF and MMDOS can make smaller adjustments, if necessary, to fine-tune green time allocations. Although DCF does attempt to form *green waves*, this ability does not always provide much of an advantage over SAT. DCF does use a constant cycle length and this may have negatively effected its performance. We will investigate this question in future work.

7 Related Work

Our approach is inspired by the work of Tumer and Agogino [2007], who applied MAS to the problem of air traffic control. Rather than modelling airplanes as autonomous agents, the authors made a counter-intuitive choice and defined *waypoints*—intermediate positions in an airplane's flight path—as the agents. These static waypoints negotiated for the "right" to accept a plane at a particular instance in time. We adopt a similar approach to traffic control and select geographically fixed agents whose behaviour is influenced by traffic conditions. This is very different from many other traffic control systems that view the vehicles—rather than the

intersections—as their focus. To address the parameter adjustment questions from Section 3, we employ auctions to expedite parameter adjustments and coordinate intersections.

The variety of approaches to auction-based traffic control demonstrates the versatility of auctions as a means of resource allocation. Dresner and Stone [2004] did away with traffic lights entirely; relying instead on a reservation system to work out when it is safe to enter an intersection. Auctions can be deployed as a tool to determine road pricing (or congestion charge) in order to optimise route selection [Iwanowski *et al.*, 2003; Markose *et al.*, 2007]. Auctions can also be used as complete, intersection-level, traffic controllers. Carlino *et al.* [2013] described a traffic control system where second-price sealed bid auctions were used at intersections to determine order of use. Vehicles have an embedded agent bidding on their behalf, which is referred to as the *wallet agent*. A *system agent* also bids in a manner that facilitates traffic flow beneficial to the entire transportation system—while the *wallet agent* is solely (selfishly) concerned with getting its vehicle to its destination in the least expensive and quickest way. The authors tested different modes and found that the typical fixed-length traffic signal performed the worst in terms of reducing trip times.

One of the more interesting properties of utilising an auction mechanism as a component of traffic control is that it allows the intersection to consider the needs of individual drivers. Schepperle *et al.* [2007] described an intersection controller called *Initial Time-Slot Auction (*ITSA*)* which is *valuation-aware*—a mechanism that takes into consideration the individual's cost of waiting at an intersection. In ITSA, vehicles approach and *register* with an intersection. An *intersection agent* executes a second-price sealed-bid auction for the most current time slot available. The authors also described two variants of ITSA: a mechanism is included to prevent *starvation*[5] where auctions are suspended if vehicle waiting time has reached some fixed limit; and ITSA+SUBSIDIES, which considers subsidies where vehicles that have not participated in an auction yet can influence the auction of the vehicles in front of them. The authors compared their traffic controller to the reservation-based system in Dresner and Stone [2004]. Both ITSA and ITSA+SUBSIDIES were able to reduce average travel time while minimising average weighted waiting time, as compared to the reservation-based system. ITSA+SUBSIDIES was better at reducing average weighted waiting time.

Vasirani *et al.* [2012] expanded on Dresner and Stone's [2004] work by examining the performance changes to a reservation-based system where time slots were allocated using a *combinatorial auction (*CA*)*. As drivers approached the intersection, reservations were awarded through the auction, instead of simply handed out in order of arrival (the Dresner and Stone approach). In this way, drivers express their true valuation for a contested reservation. In a network with a single intersection, the authors looked at the delay experienced by drivers based on the amount they were willing to "pay" to use the intersection. They found that initially having a willingness to pay does decrease delay, but eventually this levels off. However, CA was found to increase overall delay. As the intensity of traffic increased, CA experienced far more delays and rejected reservations than the first-come, first-served approach. Both reservation-based systems described in [Dresner and Stone, 2004; Vasirani and Ossowski, 2012] rely on vehicle agents having the capability to communicate with each other.

Other researchers have investigated approaches similar to our auction-based mechanism. Mashayekhi & List [Mashayekhi and List, 2015] designed a multi-agent auction-based traffic controller. The major difference between our approach and [Mashayekhi and List, 2015] is in the bidding strategy. We designed our bidding strategy from common traffic engineering practices while Mashayekhi & List used Reinforcement Learning to acquire a bidding strategy. Another significant difference is their traffic controller needs vehicle-to-infrastructure communication: as vehicles approach an intersection, they must report their presence to the movement managers via *tokens*. Our methods do not rely on such technologies.

8 Summary

We have presented our exploratory work on automated traffic control systems that do not require the existence of vehicle agents and can adjust dynamically as road conditions change. Moreover, our approach uses local traffic state information gathered from induction-loop vehicle detectors. As a result, our market-based traffic control methods are not constrained by the lack of transportation communication devices and protocols. Locally acting agents provide a robust traffic control system that maintains performance gains during and after traffic flow disruptions.

In patterned traffic, such as Structured, SCOOT performs well, but so do fixed-time signals. Thus, when recognised, these traffic patterns can be exploited; but this is not always the case in large cities where traffic disruptions (such as accidents or local events) can easily perturb the norm. Through a broad series of experiments, we have demonstrated the efficacy of our new approach, in comparison with our earlier work and several benchmarks (SCOOT, fixed-time signals and a reinforcement learning approach). The experimental results highlight the impact of including offset and fine-tuned green time adjustments in bidding, which produce improvements in travel time. Our next steps with this work involve incorporating elements in the bidding to improve green waves. We will also continue evaluating the traffic parameters discussed in this paper with the aim of developing a clearer picture of the impact that adjusting split, cycle and offset (and various combinations thereof) has on travel time.

References

[Bakker *et al.*, 2010] Bram Bakker, Shimon Whiteson, Leon J. H. M. Kester, and Frans C. A. Groen. Traffic Light Control by Multiagent Reinforcement Learning Systems. In Robert Babuska and Frans C. A. Groen, editors, *Interactive Collaborative Information Systems*, volume 281

---

[5] In this context, *starvation* refers to one traffic flow being given a green signal for (too) long periods, and the other (stopped) traffic flows are "starved" for green time.

of *Studies in Computational Intelligence*, pages 475–510. Springer, 2010.

[Carlino *et al.*, 2013] Dustin Carlino, Stephen D. Boyles, and Peter Stone. Auction-based autonomous intersection management. In *Proceedings of the 16th IEEE Intelligent Transportation Systems Conference (ITSC)*, 2013.

[Dresner and Stone, 2004] Kurt M. Dresner and Peter Stone. Multiagent Traffic Management: A Reservation-Based Intersection Control Mechanism. In *Proceedings of the Third International Joint Conference on AAMAS*, pages 530–537. IEEE Computer Society, 2004.

[Gartner *et al.*, 2001] Nathan H. Gartner, Farhad J. Pooran, and Christina M. Andrews. Implementation of the OPAC Adaptive Control Strategy in a Traffic Signal Network. In *IEEE Intelligent Transportation Systems*. IEEE, 2001.

[Gomes, 2014] Lee Gomes. *Hidden Obstacles For Google's Self-Driving Cars: Impressive Progress Hides Major Limitations Of Google's Quest For Automated Driving*. MIT Technological Review, (www.technologyreview.com), 2014.

[Halkias, 1997] John A. Halkias. Advanced transportation management technologies : participant reference guide : Demonstration Project No. 105. Tech Report FHWA-SA-97-058, United States. Federal Highway Administration, 1997.

[Hunt *et al.*, 1981] P. B. Hunt, D. I. Robertson, R. D. Bretherton, and R. I. Winton. SCOOT - A traffic responsive method of coordinating signals. Technical Report 1014, Transport and Road Research Laboratory, 1981.

[Iwanowski *et al.*, 2003] S. Iwanowski, W. Spering, and W.J. Coughlin. Road traffic coordination by electronic trading. *Transportation Research Part C: Emerging Technologies*, 11(5):405–422, 2003. cited By 0.

[Krajzewicz *et al.*, 2012] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker. Recent Development and Applications of SUMO - Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements*, 5(3&4):128–138, December 2012.

[Lee *et al.*, 2002] Sang Soo Lee, Seung Hwan Lee, Young Tae Oh, and Kee Choo Choi. Development of Degree of Saturation Estimation Models for Adaptive Signal Systems. *RSCE Journal of Civil Engineering*, 6(3):337–345, 2002.

[Limited, 2016] T. R. L. Limited. *SCOOT Advice Leaflet 1: The SCOOT urban traffic control system*. April 2016.

[Litman, 2015] Todd Litman. Autonomous Vehicle Implementation Predictions. Technical report, Victoria Transport Policy Institute, December 2015.

[Markose *et al.*, 2007] S. Markose, A. Alentorn, D. Koesrindartoto, P. Allen, P. Blythe, and S. Grosso. A smart market for passenger road transport (SMPRT) congestion: An application of computational mechanism design. *Journal of Economic Dynamics and Control*, 31(6):2001–2032, 2007. cited By 0.

[Mashayekhi and List, 2015] Mehdi Mashayekhi and George List. A Multi-agent Auction-based Approach for Modeling of Signalized Intersections. In *Second Workshop on Synergies Between Multiagent Systems, Machine Learning and Complex Systems (TRI 2015)*, Buenos Aires, Argentina, July 2015.

[Mirchandani and Wang, 2005] Pitu Mirchandani and Fei-Yue Wang. RHODES to Intelligent Transportation Systems. *IEEE Intelligent Systems*, 20(1):10–15, 2005.

[Mladenovic and Abbas, 2013] Milos Mladenovic and Montasir Abbas. A Survey of Experiences with Adaptive Traffic Control Systems in North America. *Journal of Road and Traffic Engineering*, 59(2):5–11, 2013.

[Papageorgiou *et al.*, 2003] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Yibing Wang. Review of road traffic control strategies. *Proceedings of the IEEE*, 91(12):2043–2067, December 2003.

[Raphael *et al.*, 2015] Jeffery Raphael, Simon Maskell, and Elizabeth Sklar. From Goods to Traffic: First Steps Toward an Auction-based Traffic Signal Controller. In *13th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS'15)*. Springer, 2015.

[Roess *et al.*, 2009] Roger P. Roess, Elena Prasas, and William R. McShane. *Traffic Engineering: International Edition, 4th Edition*. Pearson Education, Inc., 4th edition, 2009.

[Schepperle and Böhm, 2007] Heiko Schepperle and Klemens Böhm. Agent-Based Traffic Control Using Auctions. In Matthias Klusch, Koen V. Hindriks, Mike P. Papazoglou, and Leon Sterling, editors, *CIA*, volume 4676 of *Lecture Notes in Computer Science*, pages 119–133. Springer, 2007.

[Shanker *et al.*, 2013] Ravi Shanker, Adam Jonas, Scott Devitt, Katy Huberty, Simon Flannery, William Greene, Benjamin Swinburne, Gregory Locraft, Adam Wood, Keith Weiss, Joseph Moore, Andrew Schenker, Paresh Jain, Yejay Ying, Shinji Kakiuchi, Ryosuke Hoshino, and Andrew Humphrey. Autonomous Cars, Self-Driving the New Auto Industry Paradigm. Technical report, Morgan Stanley Research, November 2013.

[Sutton and Barto, 1998] Richard S. Sutton and Andrew G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition, 1998.

[Tumer and Agogino, 2007] K. Tumer and A. Agogino. Distributed Agent-Based Air Traffic Flow Management. In *Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 330–337, May 2007.

[Vasirani and Ossowski, 2012] Matteo Vasirani and Sascha Ossowski. A Market-Inspired Approach for Intersection Management in Urban Road Traffic Networks. *Journal Artificial Intelligence Research*, 43:621–659, 2012.

[Wang, 2005] Fei-Yue Wang. Agent-Based Control for Networked Traffic Management Systems. *IEEE Intelligent Systems*, 20(5):92–96, 2005.