Transportation Research Part C 17 (2009) 456–474

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Adaptive traffic signal control using approximate dynamic programming

Chen Cai *, Chi Kwong Wong, Benjamin G. Heydecker

Centre for Transport Studies, University College London, London WC1E 6BT, United Kingdom

ARTICLE INFO

Article history:
Received 6 May 2008
Received in revised form 31 January 2009
Accepted 14 April 2009

Keywords:
Traffic signal
Dynamic programming
Approximation
Adaptive
Reinforcement learning

ABSTRACT

This paper presents a study on an adaptive traffic signal controller for real-time operation. The controller aims for three operational objectives: dynamic allocation of green time, automatic adjustment to control parameters, and fast revision of signal plans. The control algorithm is built on approximate dynamic programming (ADP). This approach substantially reduces computational burden by using an approximation to the value function of the dynamic programming and reinforcement learning to update the approximation. We investigate temporal-difference learning and perturbation learning as specific learning techniques for the ADP approach. We find in computer simulation that the ADP controllers achieve substantial reduction in vehicle delays in comparison with optimised fixed-time plans. Our results show that substantial benefits can be gained by increasing the frequency at which the signal plans are revised, which can be achieved conveniently using the ADP approach.

1. Introduction

Operating traffic signals in urban areas requires proper timings, so that varying demands can be managed effectively. Conventional algorithms, which are optimised off-line, usually generate a library of signal timing plans each with fixed stage duration and sequence. Plans are retrieved from the library for implementation according to the time of day and the day of week. Such plans require manual maintenance and updating, otherwise the performance declines at a rate of about 3% per year (Bell and Bretherton, 1986). Most of the operating signal systems today are traffic responsive (or vehicle actuated). The responsiveness to traffic is that the allocation of green times is adjusted according to real-time traffic information. The real-time traffic data are usually detected by using inductive loops. While bringing substantial benefits in reducing vehicle delays and stops, responsive systems are usually constrained by preset control parameters, such as cycle length and stage sequence. Ideally, for optimisation over time, the controller should operate signals regardless of cycle and stage constraints. This requires a dynamic controller to act according to detected traffic and update its control parameters online without human intervention, thus being adaptive.

Dynamic programming (DP) developed by Bellman (1957) is so far the only exact solution for optimisation over time. It decomposes a control problem to a series of sub-problems which we denote as *step*, which corresponds to discrete segments of time in real-time control problem. Associated with each step is a set of *state* variables that give information on the controller and the traffic environment at that time. The DP recursively calculates *Bellman's equation* backwards step-by-step to find the optimal action that transfers the system from the current state to a new state. In doing this, the DP generates backwards in time a sequence of optimal actions that guarantee global optimality. The DP solutions to traffic signal control are studied in Robertson and Bretherton (1974) and Gartner (1983). The results show that using DP can reduce about 56% vehicle delays from the best fixed-time plans. Nevertheless, the DP's implication for real-time traffic signal control is limited. The

* Corresponding author. Tel.: +44 (0)20 76790467; fax: +44 (0)20 76791567.
E-mail addresses: c.cai@ucl.ac.uk (C. Cai), ckwong@transport.ucl.ac.uk (C.K. Wong), ben@transport.ucl.ac.uk (B.G. Heydecker).

0968-090X/$ - see front matter
doi:10.1016/j.trc.2009.04.005

computational demand in the recursive calculation of Bellman's equation is exponential to the size of the state space, the information space and the action space. This scenario is often described as the 'Three Curses of Dimensionality' (Powell, 2007). Furthermore, the DP requires a complete information on the time period in which the controller seeks optimisation. In real-time operation, however, traffic detectors may supply only 5–10 s data of future arriving vehicles.

To overcome the difficulties in applying DP and to preserve the fundamental features of dynamic control, a favourable option is approximation. An approximation to DP usually aims to reduce state space by replacing a look-up table of state values with aggregations or a continuous approximation function. Such an approach is frequently denoted as *approximate dynamic programming* (ADP). In this paper, we limit the study to continuous approximation function only. Since we may not know the appropriate values of functional parameters *a priori*, it is preferable that the controller acquires adaptive features that update the parameters according to both the changes in the prevailing traffic and the observation of the controller's interaction with the traffic environment. An approach that applies the fundamentals of dynamic programming to learn from interactions with the environment is *reinforcement learning*. This approach uses the dynamic programming formula to map the system state to action. The action changes the environment, and this change is communicated back to the controller through a scalar reinforcement signal. The functional parameters are updated by specific learning techniques upon receiving the reinforcement signal. In this study, we investigate two learning techniques, *temporal-difference* and *perturbation learning*.

In this paper, we show that an adaptive traffic signal controller using ADP and reinforcement learning is capable of reducing vehicle delays substantially from the best fixed-time control, while being computationally efficient. The numerical experiments presented here are limited to an isolated intersection with multiple signal stages, but the implication extends to distributed control in traffic networks.

This paper is organised as follows. In Section 2, we review the existing traffic signal control systems, from which we identify the scope for development and set objectives for the ADP controller. In Section 3, we introduce the fundamentals of the ADP, reinforcement learning and the specific learning techniques, based on which the control algorithms are formulated for traffic signals. Section 4 contains numerical experiments and results. Section 5, contains conclusion of this study and the scope for future research.

2. Traffic signal control

The operation of traffic signal settings can be broadly classified into off-line and online approaches. Off-line methods use historical traffic data as inputs for the calculation of signal plans. By contrast, online approaches use traffic information collected in real-time from on-street detectors to develop responsive signal control. Here, we limit the review in online control systems. Among online approaches, those that collect real-time traffic information from detectors and use it to calculate up-to-date signal settings for implementation pertain to *responsive control*; those that use real-time traffic information to select a preset signal plan according to the best match with the detected traffic pattern pertain to *plan selection*. Several well-known signal calculation packages are reviewed here; their characteristics are summarised in Table 1.

SCOOT (Hunt et al., 1982) and SCATS (Luk, 1984) are basically online variants of off-line optimisation strategies. Manual engineering work is required to update traffic data and feed them into an off-line optimiser, for example TRANSYT (Vincent et al., 1980), for the preparation of a library of plans that apply to different periods of a day and days of a week. The ultimate performance of such systems depends on the accuracy of the database and its conformity to software requirements. The online capability then enables the selection of the most appropriate plan from the library according to detected traffic, adjusts offsets between adjacent intersections to facilitate traffic flow, and makes small adjustments to the signal plan. SCOOT

Table 1
Summary of different design programs for traffic signal control.

Program	Traffic data	Decision on signal settings	Signal cycle	Signal coordination	Origin country	Objective for optimisation	Servo mechanism
OPAC	Online data from upstream detectors	Change of current signal settings Rolling forward	Acyclic	Through traffic profile	USA	Delay	Decentralized
UTOPIA	Online data from upstream detectors	Green start times, durations and offsets	Required	With offset optimisation	Italy	Stops and delay	Centralized
SCATS	Online data from stop line (downstream) detector	Pre-calculated signal plan selection	Required	With offset optimisation	Australia	Capacity*	Centralized
SCOOT	Online data from upstream detectors	Adjustment of whole signal plan	Required	With offset optimisation	UK	Stops, delay and congestion	Centralized
PRODYN	Online data from pair of upstream detectors	Change of current signal settings	Acyclic	Possible	France	Total delay	Decentralized
MOVA	Online data from a single upstream detector	Green extension or not	N/A	Nil	UK	Stops, delay and capacity	Decentralized
DYPIC	Off-line basis and perfect information	Complete signal settings	Acyclic	Nil	UK	Delay	Decentralized

* Vehicle discharge rates are monitored, and signal will be changed if it is found less than the saturation flow rate.