

Solving the Integrated Corridor Control Problem Using Simultaneous Perturbation Stochastic Approximation

Jingtao Ma

Ph.D Candidate
University of California, Davis
Dept. of Civil & Environmental Engineering
One Shields Avenue, Davis, CA 95616
Tel: 530-754-6429
Email: jtma@ucdavis.edu

Yu Nie

Ph. D.
University of California, Davis
Dept. of Civil & Environmental Engineering
One Shields Avenue, Davis, CA 95616
Tel: 530-754-6429
Email: ynie@ucdavis.edu

H. Michael Zhang

Professor
University of California, Davis
Dept. of Civil & Environmental Engineering
One Shields Avenue, Davis, CA 95616
Tel: 530-754-9203
CKP Professor
Tongji University,
School of Transportation Engineering
1239 Siping Road,
Shanghai, China 200092
Email: hmzhang@ucdavis.edu

Submitted to AHB25, TRB Committee of Traffic Signal Systems
Submission Date: July 31, 2006

Word Count: 5,820 + 1 table + 6 figures = 7,520

Revised on October 30, 2006

ABSTRACT

Integrating various control measures within a transportation corridor is believed to improve the overall operational performance of the entire corridor. In this study, we formulate a corridor control problem that considers two control actions: traffic signal timing and ramp metering, and propose a solution method for the formulated problem. In our formulation, traffic dynamics within a general corridor is modeled on a coherent platform based on the kinematic wave traffic flow model, and the traffic control actions of urban street signals and ramp meters are embedded in the platform. One solution algorithm based on the simultaneous perturbation stochastic approximation (SPSA) scheme is developed to optimize the integrated control. Numerical experiments show that the SPSA algorithm is computationally much more efficient than genetic algorithms (GA) and the hill-climbing algorithm, while its solutions are better than or comparable to the solutions obtained from the two other methods.

INTRODUCTION

A transportation corridor is operationally (rather than geographically or organizationally) defined as “a combination of discrete parallel surface transportation networks (e.g., freeway, arterial, transit networks) that link the same major origins and destinations” [1]. A corridor usually includes various types of facilities (e.g., freeway sections, ramps and urban streets), which are typically managed by different agencies and jurisdictions. In current practices, most corridors are operated separately with little consideration to the coordination of individual facilities [2-3], although it has long been recognized that integrating the control measures can improve the operational performance of the entire corridor (e.g., [4]).

Two components are fundamental to modeling an integrated corridor control system (e.g., [5]). The first is the traffic flow model that realistically represents traffic evolution, and the other is the optimization method to generate optimal control plans. Three major categories of traffic flow models have been developed and applied in traffic control studies: the point-queue (P-Q) or vertical queue model, the spatial queue (S-Q) or horizontal queue model and the Lighthill-Whitham-Richards (LWR) model. Most studies, including the classical ones such as Webster's [6] and later HCM methods, used the P-Q model. In this model the vehicles are assumed to travel at the design speed uniformly along the road section and arrive at the stop line at a constant rate. The vehicles behind the stop line take no physical space and will be discharged at the saturation flow rate during the effective green time. The platoon dispersion model in TRANSYT [7] uses an empirical formula to depict the cyclic flow profiles (CFP) on road sections and thus relaxes the constant arrival assumption, but vehicles are still queued at the stop line. TRANSYT version 8 [8] advanced to allow vehicles to join at the end of the stopped queue. Link storage capacity constraint is enforced and no traffic can enter a link if it is occupied by stopped vehicles. This S-Q model is seeing more applications recently in other traffic control studies [9-11].

Both the P-Q and S-Q models can provide good estimates of the *queue size*, i.e., the number of stopped vehicles, under low to medium traffic loads in particular. But when the traffic load is high and the intersection is near or over saturated, the traffic densities behind the stop line will be in frequent transitions because of the varying arrival rates and intermittent signal services [12]. Shock and acceleration waves, interfaces between two differing traffic states, will be generated in such a complicated way that neither the P-Q nor the S-Q model could capture the spatial extent of queue formations and dissipations. Consequently, *queue lengths* cannot be estimated accurately. In this study, queue length is stated as “the length of the roadway section behind the stop line where traffic conditions are in the congested region of the flow-density curve, i.e., they range from capacity to jammed.” [12]. While Michalopoulos and Stephanopoulos [13] further argued that under these circumstances the control action would be dictated by minimizing queue length instead of delay, Stephanopoulos et al [12] incorporated the more elaborate LWR model to analyze the complicated queuing phenomena at the signal intersections. In their analysis the linear speed-density relation [14] and the resulting parabola fundamental diagram (flow-density curve) was used to compute the maximum queue length analytically. Using the triangular fundamental diagram, Helbing [15] recently derived the formula for queue dynamics and travel time variations with respect to the arrival and departure flow rates. A self-organized control method was later developed based on these results [16].

Recently, researchers make use of a finite difference solution scheme to the LWR model, the so-called cell transmission model (CTM) [17-18], to perform traffic control studies. A linear transformation of the CTM model has been carefully designed to study the global optimal ramp metering strategies [19-20]. In an earlier work, Lo [21-22] also modified the original CTM to formulate the signal control problem into a mixed integer program. The program only considered the intersections without turns; and generally it is tedious to solve. He later applied a genetic algorithm (GA) based solution algorithm to optimize control plans for more general intersection layouts [23].

Optimization methods used to solve a traffic control problem are highly tied to the underlying traffic flow models. For instance, in [9-10], the researchers used the store-and-forward approach to depict the flow dynamics of urban streets, ramps and freeway mainline. This approach is essentially similar to the S-Q model, and the formulated integrated corridor control problem is a linear one with a sparse constraint set, for which highly efficient algorithms exist. But typically the store-and-forward approach requires the control updating time period to be no less than a signal's cycle length; this feature rules out the possibility of synchronizing the control actions and thus make the model only suitable as a strategic queue-management tool [10]. Later they adopted the high-order flow model in METANET [24] and studied integrated ramp metering and Variable Message Sign (VMS) controls. Conjugate gradient algorithms were deployed to solve the integrated control problem [25]. The same algorithm was applied in [5], where a forward time centered space method was used to model traffic evolution. The resulting system state equations are also twice-differentiable. However, both studies can only guarantee local optima, which can be sensitive to the initial guess of the solution [5].

To summarize, mathematical programming methods [e.g., 5, 9, 10, and 25] usually require the traffic flow models to be simplified so that the gradient information can be computed. Such simplification often compromises the underlying traffic flow models. On the other hand, heuristic optimization methods such as the genetic algorithm can search for a near-global optimal control plan while allowing more realistic representation of traffic flow (e.g., [23]). However, heuristic methods usually need a large number of evaluations of system performance and usually lead to high computational costs.

In this paper, we explore a stochastic approximation technique that can be viewed as a compromise of the above two types of approaches. The proposed simultaneous perturbation stochastic approximation (SPSA) has been used in other fields [26] and shown satisfying performances. In this study, an SPSA-based algorithm is developed to compute the time-of-day optimal corridor control plan, while the corridor operational performances under various control plans are evaluated on a CTM-based platform. The platform embeds signal control and ramp metering in a generic way and can thus model any general corridor network. Numerical examples are used to investigate the effectiveness of the method as compared to other heuristics methods. Practical guidelines of applying the SPSA method are also discussed.

MODELING DYNAMIC NETWORK FLOW

In this section, a cell transmission model (CTM) based network flow model is built, in which the control actions from traffic signals and ramp meters are modelled in a coherent way.

Flow Dynamics on a General Corridor Roadway Section

The well accepted LWR model states the following:

$$\frac{\partial q}{\partial x} + \frac{\partial \rho}{\partial t} = 0 \quad q = f(x, \rho, t) \quad (1)$$

where q is the flow rate on a road section; ρ is the density; x and t are the space and time variables, respectively. Daganzo [17] developed a stable numerical approximation scheme that approximates the LWR model. He shows that, if the relationship between traffic flow q and density ρ is in the form

$$q = \min\{v\rho, q_{\max}, w(\rho_j - \rho)\} \quad (2)$$

where v is the free flow speed, q_{\max} is the maximum flow rate, w is the backward shockwave speed and ρ_j is the jam density, then LWR model can be approximated by a set of difference equations. The model discretizes the entire time horizon T (assignment period) into small intervals t , the *loading interval*. Conforming to the loading interval, the model divides every road section of the network into homogeneous segments called cells, in a way that the cell length equals the distance traversed by one typical vehicle at free flow speed in one loading interval. The flows are updated by the following difference equations:

$$y_i(t) = \min\{n_{i-1}(t), q_{i,\max}, \delta(N_i - n_i(t))\} \quad (3)$$

and

$$n_i(t+1) = n_i(t) + y_i(t) - y_{i+1}(t) \quad (4)$$

where $y_i(t)$, $y_{i+1}(t)$ are the fluxes that entering cell i and $i+1$ at time t , respectively, $n_{i-1}(t), n_i(t), n_{i+1}(t)$ are the numbers of vehicles in the cell $(i-1)$, i and $i+1$ at time t , respectively, $q_{i,\max}$ is the capacity flow into i at t , $N_i - n_i$ is the space available in i , $\delta = w/v$.

Essentially equation (4) tells that the number of vehicles staying in cell i at loading interval $t+1$ is the number of vehicles from interval t plus the incoming vehicles and minus the outgoing vehicles. Daganzo [18] extended the model to a general network by carefully dividing various roadway junctions into basic merges and diverges. Since control actions take places at junctions, we will introduce the flow updating rules at general junctions including signalized intersections and metered ramps.

Flow Updating at Signalized Urban Intersections

In [21], Lo showed that CTM can be deployed to model the flow updates at urban intersections with a few changes. If the flow capacity q_{\max} in equation (2) is replaced by one that depends on the signal timing variable $g_i(t)$,

$$q_{\max}(t) = \begin{cases} q_{\max} & t \in \text{green} \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where it switches between q_{\max} (green) and zero (red), the end cell of an intersection approach will serve as a functioning signal, and the flow dynamics still approximates the LWR model. At a typical intersection, traffic is grouped into movements *or* streams. A generalized four-leg intersection with all vehicular movements can be illustrated in Figure 1.

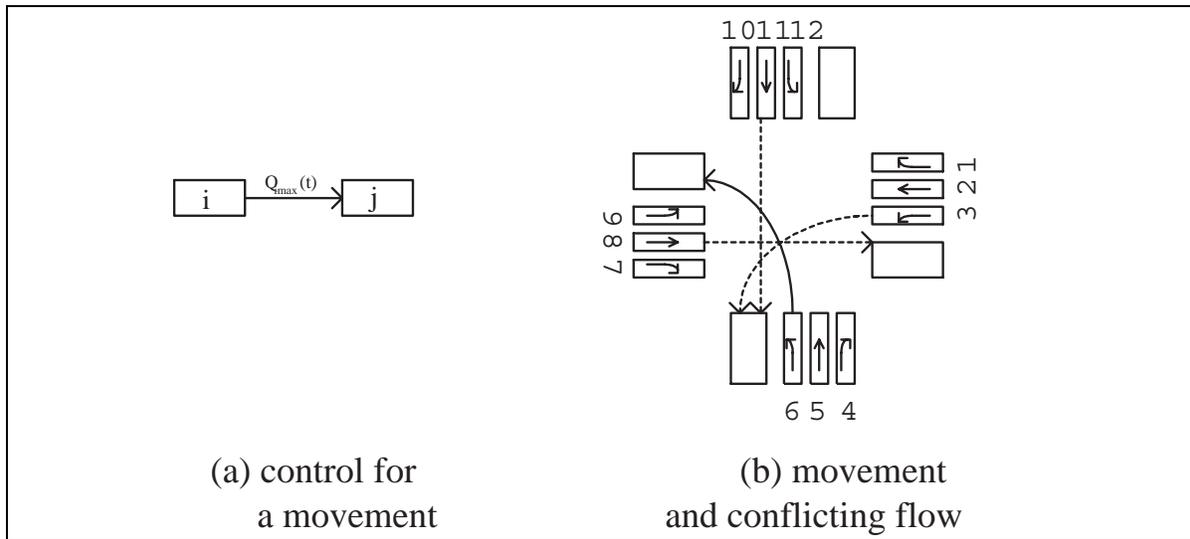


Figure 1 A general representation of cell-based intersection movements

Signalized Diverges

The diverging flows occur where the traffic stream on a single link splits into left turn, through and right turn movements. It is a common engineering practice to enlarge the road section behind the stop line to store the incoming vehicles. To accommodate this feature, the end cell C_s^j of a link j approaching a signalized intersection, the flow updating rule reads:

$$n_s(t+1) = \sum_{m=L,R,T} n_s^m(t) + \sum_{m=L,R,T} y_s^m(t) - \sum_{m=L,R,T} y_{s+1}^m(t) \quad (6)$$

The superscripts of L, R, T denote the left turn, right turn and through movement, respectively.

The cell C_{s-1}^j is the preceding cell of C_s^j . The fluxes into and out of cell s are stated as:

$$y_s^m(t+1) = \min\{n_{s-1}^m(t), q_{s, \max}^m(t), \delta(N_s^m - n_s^m(t))\}, m = L, R, T \quad (7)$$

$$y_{s+1}^m(t+1) = \min\{n_s^m(t), q_{s, \max}^m(t), \delta(N_{s+1}^m - n_{s+1}^m(t))\}, m = L, R, T \quad (8)$$

where the notation naming convention follows (3) and (4). Note that $N_s^m, m = L, R, T$ in equation (7) are the different storage capacities for various movements, ensuring that different sizes of turning bays can be modeled accurately.

Signalized Merges

In this study, the right turns are explicitly considered in the signal timing optimization. In this way, the flow updating at intersections is simplified to be the same as a set of coupled consecutive links, which then reads:

$$n_{i+1}(t+1) = n_{i+1}(t) + y_{i+1}(t) - y_{i+2}(t) \quad (9)$$

where $(i+1)$ is the start cell index for the downstream link, i.e., the first cell of the downstream link that receives the stream with cell index of i serviced by the signal. The incoming flux $y_{i+1}(t)$ is then determined by the signal timing plan but shares the same updating rules as in (3) with $q_{i,\max}$ replaced by $q_{i,\max}(t)$ in (5). One may notice that this simplified treatment has also been used in Lo's study [21-23].

The above defined flow dynamics model can conveniently accommodate all four types of signal control actions, namely cycle length C , phase sequencing, phase duration g and offset Δ between two adjacent signalized intersections. In this study, the offset is in reference with respect to the start of the analysis horizon; the numerical values of each variable are also calculated in the multiples of the loading interval t .

Metered Freeway Onramp

Modeling ramp meters has only one control variable to deal with, the metering rate at on-ramp j at time t . For notational simplification, the ramp subscript j is omitted in the following development. Modifying the demand-supply method for merges [18], we apply one generic flow updating rule to represent the flow dynamics at a freeway merge section [27]:

$$y_i(t) = \min\{n_i(t), q_{i,\max}(t), \delta(N_i - n_i)\} \quad (10)$$

$$D_R^t = \min(D_R^t, R^t) \quad (11)$$

$$D^t = D_M^t + D_R^t \quad (12)$$

$$S^t = \min(S_M^t, D^t) \quad (13)$$

$$f_M^t = \frac{D_M^t}{D^t} S^t \quad (14)$$

$$f_R^t = \frac{D_R^t}{D^t} S^t \quad (15)$$

where the ramp metering R^t is embedded, and other notations are:

- D_R^t : Ramp demand at time t ;
- D^t : Demand upon the beginning cell of the link downstream of the ramp;
- D_M^t : Competing demand on mainline;
- S_M^t : Supply of the beginning cell of the downstream link;

- S^t : Total service flow rate;
 f_R^t : Outflow from ramp;
 f_M^t : Outflow from upstream mainline.

The modification mainly lies in two aspects: (i) the ramp demand to the merge point is bounded not only by actual demand and the flow capacity, but also by the metering rate executed at that time step (Equation 11); (ii) in the overflow or congestion situation, the freeway mainline and ramp flows will be distributed proportionally to their relative demand (Equations 13-15) [27]. The ramp metering takes effect in the form of R^t .

Traffic Demand Input and Vehicle Routing

In the model, the traffic demand is given externally at any source node i :

$$Q_j(t) = \sum_r \sum_s D^{r,s}(t), \forall (r, s) \in \{(R, S)\} \quad (16)$$

where $Q_j(t)$ is the sum of the time-dependent demands entering the source node j .

Because our main focus is the development of optimal control plans, the discussion of users' route choice behavior is reduced to a minimum. In the later numerical examples, only one pre-determined shortest path is utilized for any path flow $D^{r,s}$. The network flow pattern and the resulting performance measure will only be determined by the specified control plan. Nevertheless, one needs to note that the proposed framework can be easily adapted to study the interaction between users' route choice and control strategies (e.g., [28-29]).

Minimizing Total Delay for Integrated Corridor Control

The performance of a control plan is often evaluated through delays and the number of stops, and many other criteria can be calculated or extracted from these two quantities. We thus select minimizing delay as the control objective. The fundamental diagram tells two regions that traffic flow status can fall into, the free flow region and the forced flow region. Once the flow falls in the forced flow region, the vehicles will not operate at the free flow speed any more, and delays are incurred to the vehicles. In the model, the total delay is the accumulation of the delay at the cell level, while the latter is conveniently expressed as the following:

$$d_i(t) = d(n_i(t) - y_i(t)) = (n_i(t) - y_i(t)) \bullet t \quad (17)$$

where $d_i(t)$ is the delay occurring at cell i during time interval t , and $n_i(t)$, $y_i(t)$ are the number of vehicles in i at t and the number of vehicles that can go out of i at t . When the loading interval t is a unit time one, the delay can simply be numerically represented by $n_i(t) - y_i(t)$. That is, the delay is the time that $n_i(t) - y_i(t)$ vehicles are forced to stay in cell i during time step t (because CTM dictates the movement of vehicles from only one cell into the next one at each time step). The objective function is then the summation over all cells and the overall time horizon:

$$\min D(C, \Delta, g, R) = \min \sum_t \sum_i d_i(t) \quad (18)$$

where $D(\bullet)$ is the total delay of the system, and $(\mathbf{C}, \Delta, \mathbf{g}, \mathbf{R})$ is the vector of the cycle, offset, phase duration of all street signals and \mathbf{R} is the vector of metering rates of all ramp meters in concern.

Practical Control Constraints

In practice, traffic controls usually enforce some physical constraints including the maximum and minimum duration of the cycle length and green duration for any phase, and the max/min metering rates as follows:

$$C_{i,\min} \leq C_i \leq C_{i,\max} \quad (19)$$

$$g_{i,\min} \leq g_i \leq g_{i,\max} \quad (20)$$

$$R_{i,\min} \leq R_i \leq R_{i,\max} \quad (21)$$

In this study, we are only concerned with the generation of time-of-day traffic control plans. Furthermore, it is assumed that cycle length and phasing sequences are fixed. The cycle length constraint for any intersection then reads:

$$\sum_{h=1}^N g_h^j = C^j - NL \quad (22)$$

It tells that the sum of the effective green duration of the phases $h = 1 \dots N$ at intersection j has to be equal to the available green time $C^j - NL$, i.e., the cycle length deducted by the loss time of all phases.

THE SPSA METHOD AND SPSA BASED INTEGRATED CONTROL

Stochastic optimization techniques apply in virtually all engineering areas where a closed-form solution to the problem is not available, or the input information into the model could be contaminated with noise. One of the techniques is the simultaneous perturbation stochastic approximation (SPSA) method that uses only the objective function information to compute approximated gradient information with respect to the decision variables. This method has been used in many areas such as industrial quality control, neural network training, sensor placement and configuration and so on [26]. In the formulated corridor control problem (1-20), the complexity of the traffic dynamics model precludes direct computation of the gradient information and heuristic method is thus considered more suitable. Because of its high computational efficiency proved in other studies, SPSA method is investigated to solve the integrated corridor control problem. To note the introduction of SPSA method here draws heavily on the theoretical results in [31-33].

Introduction of the SPSA Method

For a general SPSA procedure, the general objective function $L(\theta)$ as $D(\mathbf{C}, \Delta, \mathbf{g}, \mathbf{R})$ in (18) is a scalar-valued performance measure of the system, and θ is a continuous-valued p -dimensional

vector of parameters, i.e., $(\mathbf{C}, \Delta, \mathbf{g}, \mathbf{R})$ in the corridor control context. It could happen that noises ε occur when measuring the system performance measure $z(\theta)$:

$$z(\theta) = L(\theta) + \varepsilon \quad (23)$$

As a matter of fact, The SPSA method is mostly superior in the context of optimization with noisy measurements of the system of interest.

The SPSA method starts from an initial guess of θ (one feasible solution) and by a sequential “simultaneous perturbation” over the solution path, the approximation of the gradient $\varphi(\theta) \equiv \frac{\partial L(\theta)}{\partial \theta}$ will converge to zero, under several regularity conditions over the sequence.

Assume that $L(\theta)$ is differentiable over θ and the minimum is obtained at a zero point of the gradient, i.e.,

$$\varphi(\theta) = \left. \frac{\partial L(\theta)}{\partial \theta} \right|_{\theta=\theta^*} = 0 \quad (24)$$

The recursive updating of θ takes the standard form:

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k \hat{\varphi}(\hat{\theta}_k) \quad (25)$$

where the gain sequence $\{a_k\}$ must satisfy certain regularity conditions.

The perturbation is performed upon evaluating $\varphi(\hat{\theta}_k)$. First define a p -dimensional mutually independent mean-zero random variable vector $\Delta_k \in R^p \{\Delta_{k1}, \dots, \Delta_{kp}\}$ satisfying certain conditions the most important of which is that $E(|\Delta_{ki}^{-1}|)$ is bounded above by some constant α_1 , $E(|\Delta_{ki}^{-1}|) \leq \alpha_1$. An optimal distribution of Δ_k is symmetric Bernoulli [31], i.e., $P(\Delta_{ki} = \pm 1) = \frac{1}{2}$. Furthermore, $\{\Delta_k\}$ is a mutually independent sequence that is also independent of $\hat{\theta}_0, \hat{\theta}_1, \dots, \hat{\theta}_k$. Let

$$z_k^{(+)}(\theta_k) = L(\hat{\theta}_k + c_k \Delta_k) + \varepsilon_k^{(+)} \quad (26)$$

$$z_k^{(-)}(\theta_k) = L(\hat{\theta}_k - c_k \Delta_k) + \varepsilon_k^{(-)} \quad (27)$$

where c_k is a positive scalar satisfying the regularity conditions, and $z_k^{(+)}(\theta_k), z_k^{(-)}(\theta_k)$ are the measurements of the system under the perturbation $\hat{\theta}_k + c_k \Delta_k$ and $\hat{\theta}_k - c_k \Delta_k$, respectively.

The approximation of the gradient will then become:

$$\hat{\varphi}_k(\hat{\theta}_k) = \frac{z_k^{(+)} - z_k^{(-)}}{2c_k} \begin{bmatrix} \Delta_{k1}^{-1} \\ \vdots \\ \Delta_{kp}^{-1} \end{bmatrix} \quad (28)$$

Spall [30] showed that by recursively updating θ_k , the gradient will converge to a zero point. The basic recursive form (25) and gradient approximation (28) ensure that the approximation will settle down at a local minimum at least.

Regularity Conditions assuring Convergence

Five assumptions are made upon the gain sequence a_k to ensure θ_k to converge almost surely to at least a local optimum θ^* . We refer to [30] for the full development. A very brief description of the assumptions (called “regularity conditions”) is as below:

- A1:** $a_k, c_k > 0 \forall k; a_k \rightarrow 0, c_k \rightarrow 0$ as $k \rightarrow \infty; \sum_{k=0}^{\infty} a_k = \infty; \sum_{k=0}^{\infty} (\frac{a_k}{c_k})^2 = 0$;
- A2:** For some $\alpha_0, \alpha_1, \alpha_2 > 0$ and $\forall k, E\varepsilon^{(\pm)^2} \leq \alpha_0, EL(\hat{\theta} \pm \bar{\Delta}_k)^2 \leq \alpha_1, E\Delta_{kl}^{-2} \leq \alpha_2, l = 1, \dots, p$;
- A3:** $\|\hat{\theta}_k\| < \infty, \forall k$;
- A4:** θ^* is an asymptotically stable solution of the differential equation $dx(t)/dt = -\varphi(t)$.
- A5:** Let $D(\theta^*) = \{x_0 : \lim_{t \rightarrow \infty} x(t | t_0) = \theta^*\}$ where $x(t | x_0)$ denotes the solution to the differential equation of A4 based on initial conditions x_0 , there exists a compact set $S \subseteq D(\theta^*)$ such that $(\hat{\theta}_k) \in S$ infinitely often for almost all sample points.

The gain sequences of $\{a_k\}$ and $\{c_k\}$ generally take the power functions:

$$a_k = \frac{a}{(1 + A + k)^\alpha}, \quad c_k = \frac{1}{(1 + k)^\gamma} \quad (29)$$

where k is the iterator, and A is a constant introduced to stabilize the optimization process.

It is argued [33] that the asymptotically optimal values of α, γ are 1 and $\frac{1}{6}$, respectively. But Spall [26] pointed out that $\alpha < 1.0$ usually produces a better finite-sample performance. Hence another set of values of 0.602 and 0.101 that are the lowest allowable to satisfy the regularity conditions (**A1-A5**) were recommended.

It is observed that for most engineering problems these conditions are almost automatically satisfied with only **A3** being hard to verify for a general case [30]. In the corridor control problem, it physically implies that the transportation system leads to a complete gridlock. As this could be partly avoided by placing the practical constraints over the control (17-19), it does not impose difficulties in the solution as indicated in the numerical example.

Constrained Optimization via Stochastic Approximation

The SPSA procedure introduced above is suitable for solving unconstrained optimization problems. While most engineering problems are constrained by physical feasibility, the optimal corridor control problem is no exception. Sadegh [31] proposed a projection method to restrict $\theta_k \in \mathbf{R}^p$ at each iteration k to fall in the feasibility range of the control variables by simply replacing any violating $\hat{\theta}_k$ with the nearest point $\theta_k \in G(\theta)$ where $G(\theta)$ is the feasibility set of the control vector:

$$\hat{\theta}_{k+1} = P(\theta_k - a_k \hat{g}_k(\hat{\theta}_k)) \quad (30)$$

The perturbed vectors $\hat{\theta}_k - c_k \Delta_k$ and $\hat{\theta}_k + c_k \Delta_k$ in evaluating of the loss function (26-27) will also be projected such that these two perturbed vectors must lie in the feasibility range. By forcing another restriction (Assumption 1) over the constraints, SPSA can still converge to a Karash-Kuhn-Tucker point almost surely (*a.s.*) (Proposition 1 in [31]).

The set $G = \{\theta : f_i(\theta) \leq 0, i = 1, \dots, s\}$ is non-empty and bounded, and the functions $q_i(\theta), i = 1, \dots, s$, are continuously differentiable. At each $\theta \in \text{col}(G)$ where col denotes the boundary; the gradients of the active constraints are linearly independent. Furthermore, there exists an $\xi < 0$ such that the set $G^- = \{\theta : f_i(\theta) \leq r, i = 1, \dots, s\}$ is non-empty for $\xi \leq r < 0$ (i.e. the set G has a non-empty interior).

Because the parameter vector θ may have various numerical magnitudes, for example, the ramp metering rate \mathbf{R} of hundreds of vehicles per hour and the green duration g of seconds, they have to be synchronized during the decaying process. The following normalization process is then introduced:

$g_i^n = \frac{g_i - g_{i,\min}}{g_{i,\max} - g_{i,\min}}$	(31)
------------------------------------------------------------	------

where g_i^n can be any control variable with the physical boundary in (17-19). The following proposition examines whether the normalization process would affect the performance of SPSA.

Proposition *A normalized version of the projection method in constrained SPSA can assure a convergence to at least a local optimum a.s.*

Proof: It is trivial to verify the non-emptiness of the control feasibility set $G(\theta)$ since any points that fall in the box constraints (17-19) will fulfill the conditions. Since all constraints including the box constraints and summation constraint (20) are all linear, the following equation holds:

$$\frac{\partial f_i(\theta)}{\partial \theta_j} = 1 \text{ or } -1, i = 1, \dots, s, j = 1, \dots, q$$

As $g_{i,\max} > g_{i,\min}$ are constants, the linear transformation (31) does not change the above argument; then Assumption 1 for the control feasibility set after the linear transformation still holds.

With the assumptions A1-A5 and the above verification of Assumption 1, we conclude that after the linear transformation (31) as $k \rightarrow \infty$ with $\hat{g}_k(\hat{\theta}_k) = \hat{g}_k^{SP}(P_k(\hat{\theta}_k))$,

$$\hat{\theta}_k \rightarrow \hat{\theta}^* \text{ a.s.} \quad \blacksquare$$

One must note that the integrated corridor control problem formulated in (1-22) relies on the resolution of CTM. Therefore, the assumptions of continuity and differentiability could be violated numerically and the solution must be examined afterwards.

SOLUTION ALGORITHM

The iterative SPSA solution algorithm to solve the time-of-day corridor control has the following steps.

SPSA Algorithm for Integrated Network Traffic Control

Step 1: Initialization and Coefficient Selection.

- 1.0 Set iterator $k = 0$
- 1.1 Generate the control vector and normalize it via (31) as θ^N
- 1.2 Pick an initial feasible solution of θ_0^N
- 1.3 Select nonnegative coefficients a, c, A, α and γ

Step 2: Simultaneous Perturbation.

Generate a p -dimensional random perturbation vector Δ_k , where each component is mutually independent Bernoulli ± 1 distributed with probability of $1/2$ for each ± 1 outcome.

Step 3: Loss Function Evaluation by Dynamic Network Loading (DNL).

- 3.1 Perturb the normalized control vector with $\hat{\theta}_k \pm c_k \Delta_k$;
- 3.2 Project the perturbed control vectors onto $G(\theta)$ from (30);
- 3.3 Transform the projected control vector back to the real valued control variables;
- 3.4 Evaluate the system performances by loading the demand onto the network under both set of control variables and obtain (26- 27);

Step 4: Gradient Approximation.

Calculate the approximated gradient from (28).

Step 5: Control Update.

Update $\hat{\theta}_k$ with (25).

Step 6: convergence check.

If the convergence criteria are met, stop. Otherwise, set $k = k + 1$ and go to step 2.

As with any other heuristic method, the selection of appropriate parameters including the gain sequence a_k and c_k is crucial. A few selection guidelines are discussed after the numerical experiments.

NUMERICAL EXAMPLES

A Simple Network to Investigate the Convergence of SPSA

A typical diamond interchange is first constructed, where the freeway traffic travels from 1 to 2 and the surface street traffic 3-4 can go both ways (Figure 2). One ramp meter and two intersection signal groups control the traffic flow. Both ramp meter and signal controllers are assumed pre-timed, and the phasing diagram is shown in Figure 2. For illustrative purpose, only fifteen minutes of demand are set up and the hourly trip rates are also shown.

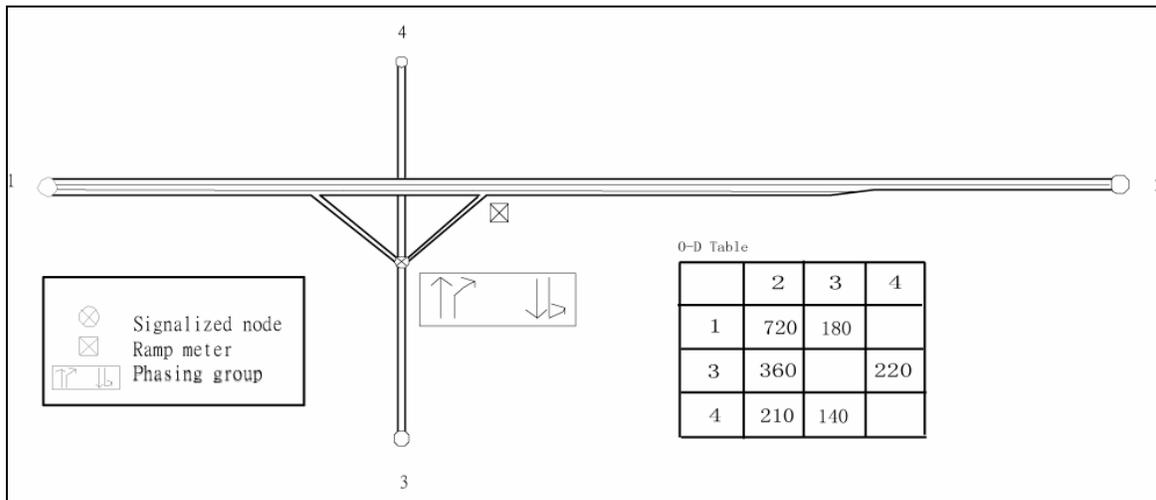


Figure 2 Geometric Layout and Demand of Sample network I

One may notice that only two independent control variables are present in the sample network, corresponding to the normalization procedure in (31), namely the green ratio g_1 for phase 1 (the green ratio for phase 2 will be $1 - g_1$ if we omit the loss time for the time being) and the metering rate R . The maximum/minimum green time is set to be 50 and 10 seconds respectively, and the range for the metering rates is set to be 300 - 1,500 vph. To locate the possible global optimal control plan, an exhaustive search through the feasibility solution space is performed. In this example, the network loading interval t is one second; thus the increment of the phase duration is set equal to t , while an increment of 20 vph is selected to scan the range of metering rates. Therefore, the exhaustive search goes through a total number of 2,400 (40×60) $g_i - R$ combinations, and the contour of the objective function (total travel time) under various combinations is plotted in Figure 3(a). The contour implies only one global optimal solution in the search space at $(g_1, R) = (0.61, 1500)$, with a total travel time of 312 veh-hr. In the example, the optimal metering rate is the upper bound of the feasible range, that is, allowing as many flows as possible into the freeway mainline during this 15-minute period.

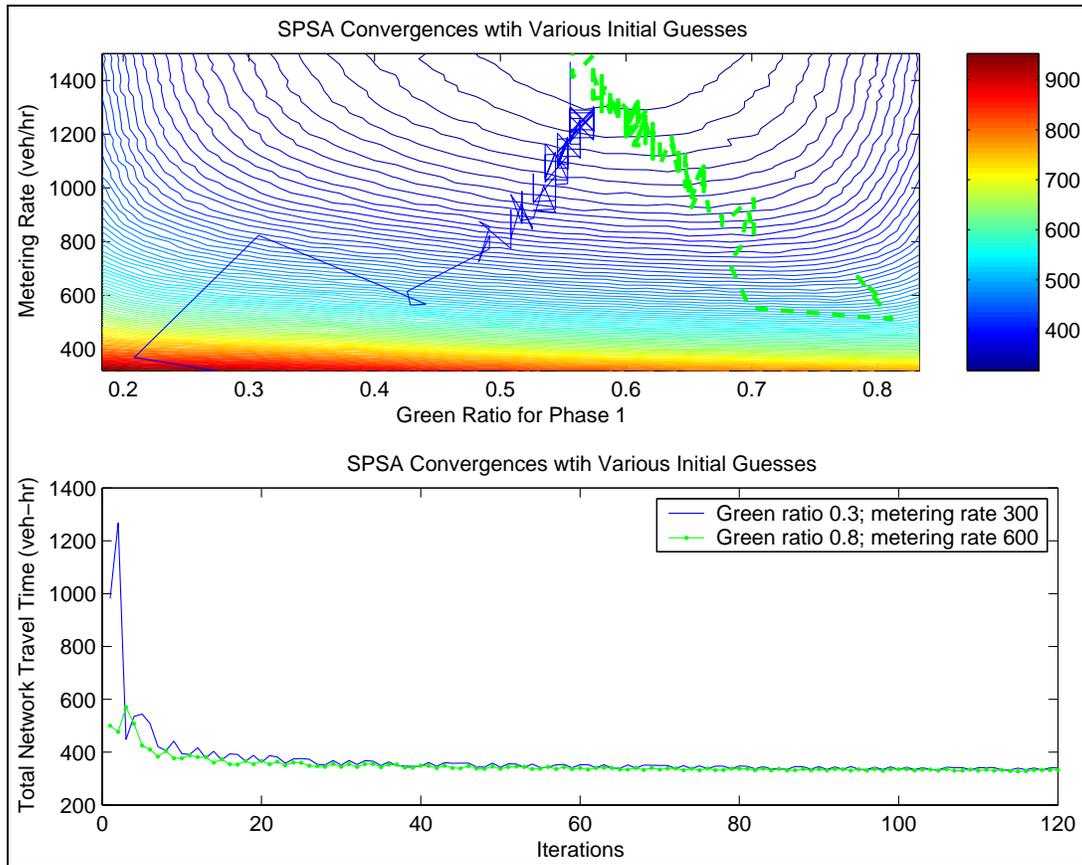


Figure 3 SPSA convergence process under various initial feasible solutions (network I)

Two SPSA processes with different initial feasible “guesses” (θ_0) are experimented and plotted in Figure 3. The first starts with $(g_1, R) = (0.30, 300)$ and stabilizes itself at $(g_1, R) = (0.58, 1473)$; the second starts with $(g_1, R) = (0.80, 600)$ and stabilizes itself at $(g_1, R) = (0.57, 1498)$. Generally both processes can reach the near-global optimal solution along a different path. Figure 3-b shows that the convergence process of SPSA has a feature of quick drop at the first few iterations; after less than forty iterations in the example, both processes reach near-global optima. This example confirms the ability of the constrained SPSA method can be applied to study the corridor control problem.

A Real Network to Investigate the Effectiveness of the SPSA Algorithm

Network Background and Preliminary Work

The other network is a real one of SR-81 corridor at Fort Worth, Texas. A DynaSmart-P network has been developed elsewhere [34] and is converted into the CTM representation. The geometric layout is illustrated in Figure 4.

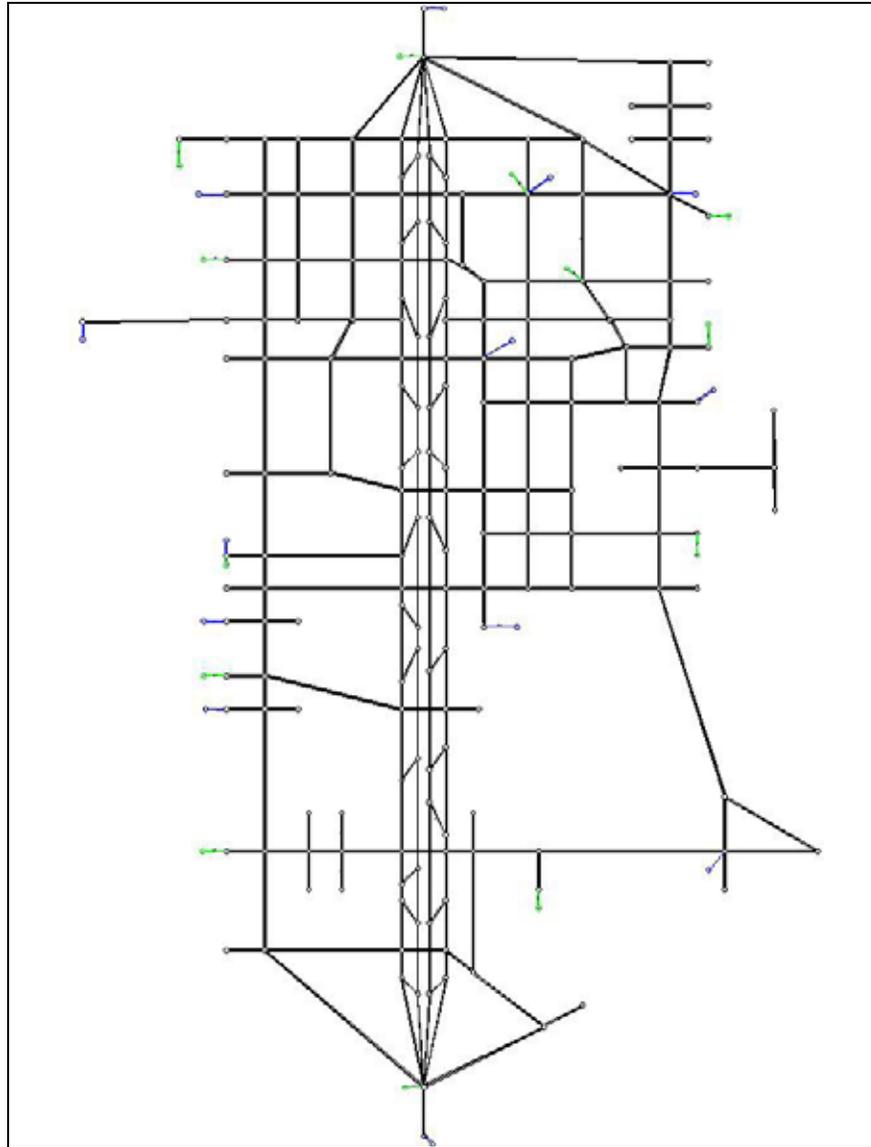


Figure 4 Geometric layout of Dallas Fort Worth network

Due to the differences in the network representation (e.g., the travel demand releasing mechanisms in DynaSmart-P and our DNL model are different) and lack of further data support, the network was slightly modified in the conversion. The most important modification is the controller type changes. In the original network, the signals are most vehicle-actuated controllers. Since herein only time-of-day corridor control is considered, all controllers are assumed pre-timed. However, the same phasing sequence and phase diagrams are inherited from the original settings.

Selecting an appropriate initial control setting, θ_0 in the SPSA context, is the first step to compute the optimal control plan. The preliminary experimentations indicate that the constructed network is heavily loaded and easily gridlocked if the controls are not properly set. An arbitrary control plan then cannot act as θ_0 because the performance index cannot be evaluated if a

gridlock happens under the control plan. A “good” control plan that at least allows the traffic flows smoothly through the network must be found before the SPSA optimization process could start.

The signal timing design procedure in HCM is followed to compute a feasible control plan. First the demand is loaded onto the network without any controls and the network flow pattern is obtained. Then the cycle length and green splits at each intersection is computed under the “equi-saturation” logic [32]. The resulting timing plans and an initial offset of zero for each intersection and a no-meter ($R = q_{\max}$ in equation 11) solution does not cause a gridlock. This HCM control plan is taken as θ_0 for the successive SPSA optimization process.

SPSA Application in the Dallas Network

A two-level procedure is taken to compute the optimal corridor control plan for the Fort Worth network. First the green splits and metering rates at each intersection or metered ramp are optimized; based on this and an adjusted common cycle length, the offsets are then computed. The first level of optimization has a total of 209 control variables encoded as the vector θ . For the second level, the offset for each intersection is all referenced to the start of the study period, and altogether 46 decision variables are encoded.

Two more documented heuristics methods are implemented to compare their relative computational capabilities. For the first level, the genetic algorithm in [23] is extended to accommodate ramp metering controls. The “hill-climbing” method in TRANSYT [7] is used to compute the offset of the second level of optimization.

The selection of parameters is important to the GA algorithm as well. The most important parameters in this algorithm include population size at each generation, and mutation rate. In this study, we apply a real-value gene-coding scheme instead of the commonly used binary-coding scheme (e.g., [33 and 23]). The real-value coding scheme is considered more efficient and accurate [35]. However, no empirical formula is available to estimate an appropriate population size as in the case of binary coding; a trial and error process then has to be used to come up with the following GA parameters: population size is 50; mutation rate 0.1; a predefined maximum generation number of 80. The convergence process with these two algorithms is shown in Figure 5.

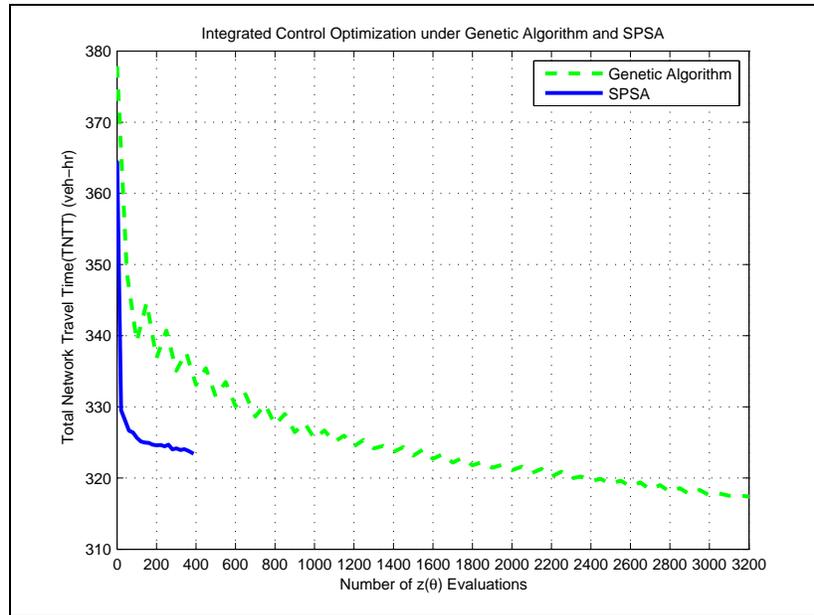


Figure 5 Computational performances of SPSA- vs. GA-based optimization of green splits/metering rates for network II

For optimizing the time-of-day control, the CPU time is mostly consumed by the evaluation of the system performance, that is, the dynamic network loading (DNL) as in Step 3 of the SPSA algorithm. For example, a single DNL process for the half-hour demand of Dallas Fort Worth network generally takes 7-10 seconds on an up-to-date PC (Pentium-4 3G CPU, 1G RAM). Then the total number of $z(\theta)$ evaluations determines the amount of computational resources when computing the optimal control plan. In Figure 5, the total network travel time (TNTT) for GA is averaged over each generation; while the TNTT in SPSA process are sampled every 20 DNL evaluations. It can be seen that SPSA only needs about 350 performance evaluations to reach a stable solution, while GA needs 3200 evaluations. It also illustrates that the objective function value can get a very sharp drop in early SPSA iterations, and this advantage can be utilized in other optimization applications to perform a quick search for a good starting solution. However, SPSA was slightly outperformed by GA in terms of the stable solutions they reached. SPSA does not jump out of an inferior “stable” solution (323.1 veh-hr) in the later process, while GA based optimization obtained a better stable solution (317.2 veh-hr) in terms of the TNTT.

The second level, offset optimization, is also conducted using both the SPSA algorithm and genetic algorithm based on the green splits obtained from their corresponding first level of optimization. The longest cycle length calculated from the critical intersection is used as the common cycle length. The cycle length and phase durations of the rest intersections are scaled accordingly. For the purpose of comparison, the classic “hill-climbing” algorithm is also implemented to compute the offset for each intersection.

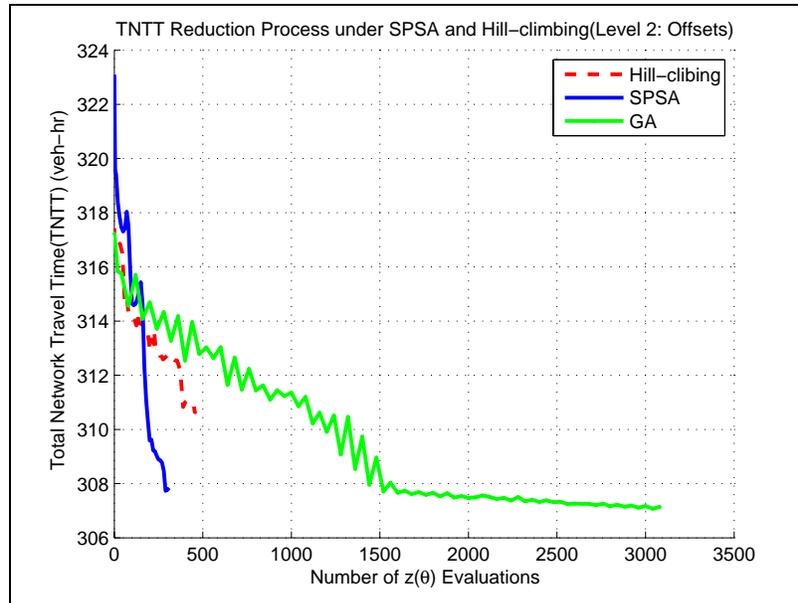


Figure 6 Computational performances of SPSA- vs. Hill-climbing based optimization of offset for network II

The hill-climbing method proceeds as a sequence of adjusting the offset at each intersection. First a step size is selected; the adjustments are then performed by a line search to find an improved global objective function that is also computed from network loading. The adjustments are incremental by the selected step size as long as the search improves the objective function. If the search degrades the objective function, the direction of adjustments will be reversed and continued at the same step size. In this way, a better offset is achieved for the intersection. Then the search proceeds to the remaining intersections. An optimization decision is made for each of several step sizes.

The optimization results of GA, SPSA and hill-climbing methods are shown in Figure 6. While all three methods can reduce the TNTT further by adjusting the offset for each intersection, hill-climbing method can only reach an inferior solution compared to the other two. Furthermore, the SPSA method outperforms both GA and hill-climbing methods using much less DNL evaluations. The solutions and performances at both levels are summarized in Table 1. It is interesting to note that the optimal solutions after the successive optimization processes from GA and SPSA are now very close (TNTT-GA 306.9 vs. TNTT-SPSA 307.5); it implies that various (local) optima could exist when searching for the optimal corridor control plan for a real network. Even though no method can guarantee a global optimal solution to the formulated corridor control problem, SPSA and GA can reach stable solutions that are comparable to each other.

Table 1 Performance Comparison of Various Optimization Methods

Method	Level 1				Level 2			
	# of $z(\theta)$ Evaluations	$z(\theta_0)$	$z(\theta^*)$	improvement	# of $z(\theta)$ Evaluations	$z(\theta_0)$	$z(\theta^*)$	improvement
SPSA	388	364.4	323.1	11%	327	323.1	307.5	4.8%
Genetic algorithm	3,200	377.9	317.2	16%	2800	317.2	306.9	3.2%
Hill-climbing	N/A	N/A	N/A		470	317.2	311.1	1.0%

GUIDELINES FOR SELECTING SPSA PARAMETERS

Selection of appropriate parameters for the gain sequence a_k and c_k is crucial to the performance of SPSA process. Spall [26] provided a few guidelines for the choice of the related parameters, i.e., α, γ, a, A and c .

With the Bernoulli ± 1 distribution for Δ_k , c can be set at a level approximately equal to the standard deviation of the measurement noise in $z(\theta)$ so that the magnitude of the approximated gradient $\hat{g}_k(\hat{\theta}_k)$ does not go excessively large. In our study, the system performance evaluation is deterministic from (1-22); in this case, c can be some small positive number. In our experimentations with various networks in the normalization scheme, it is found that 0.05 provides acceptable results, namely the change in each element of θ in the initial iterations is in the magnitude of around five percent.

It is also suggested that a “stability constant” A should be used for the sequence of a_k when large noises or variations of system performance measures are observed. A useful guideline for choosing A is to set to 10% (or less) of the maximum number of expected or allowed iterations. Meanwhile, Spall [26] also suggested to run a few preliminary replications of $\hat{g}_0(\hat{\theta}_0)$, and choosing a such that $\frac{a}{(A+1)^a}$ times the magnitudes of $\hat{g}_0(\hat{\theta}_0)$ should be approximately equal to the smallest change in θ . It is found that a larger a could lead to faster convergence to the optimal solution, but it may also run into the risk of reaching infeasible solutions (gridlock in our corridor control context). Following the above guidelines, we have found that the initial changes of 3-4 seconds in phase duration or offset values can generally provide smoother SPSA convergence.

CONCLUSIONS

An integrated corridor control problem is formulated and solved in this study. Based on the cell transmission model, the platform can capture the queuing phenomena within a general corridor network under all traffic conditions. Urban signal control and ramp metering are embedded in

the platform generically. A new heuristics solution algorithm is developed using the simultaneous perturbation stochastic approximation method. The algorithm can compute a near-global optimal control plan more efficiently compared to other heuristics methods, even if it may not guarantee global optima. Our results indicate that SPSA can be used to solve integrated corridor control problems for large-scale networks.

ACKNOWLEDGEMENT

We very much appreciate Professor James C. Spall at Applied Physics Laboratory at Johns-Hopkins University for his valuable comments in the early development of this work. Mr. Xuesong Zhou from University of Texas at Austin generously shared the Dallas Fort Worth network data used in this study. We would also like to thank all anonymous reviewers for their comments that lead to the final version. This work is partially funded by Caltrans under TO 5300 and TO 4136. The views are the authors alone.

REFERENCE

- [1] Federal Highway Administration(FHWA) 2005. Integrated Corridor Management System (ICMS) Work Plan. http://www.itsdocs.fhwa.dot.gov/icms/icms_workplan.htm. accessed 2005.
- [2] Wood, K. 1994. *Urban Traffic Control, Systems Review*. Project report 41, Transport Road Research Laboratory, Crowthorne, Berkshire, UK.
- [3] Zhang, H. M., et al 2001. *Evaluation of On-ramp Control Algorithms*. Research Report UCB-ITS-PRR-2001-36, University of California, Davis.
- [4] van Zuylen, H., H. Taale 2003. Urban networks with ring roads: a tri-level optimization. *Proceedings of 83th TRB annual meeting*, Washington D.C.
- [5] Chang, K., Y. J. Stephanedes 1993. Optimal control of freeway corridors. *ASCE Journal of Transportation Engineering*, 119(4):504–514.
- [6] Webster, F.V. 1958. *Traffic Signal Settings*. Technical Report 39, Transport Road Research Laboratory, Crowthorne, Berkshire, UK.
- [7] Robertson, D. I. 1969. *TRANSYT: a traffic network study tool*. Technical report 253, Transport Road Research Laboratory, UK.
- [8] Vincent, R.A., A. I. Mitchell, D. I. Robertson 1980. *User guide to TRANSYT version 8*. Technical Report TRRL Laboratory Report 888, TRRL Department of the Environment, Crowthorne, Berkshire, UK.
- [9] Diakaki, C., M. Papageorgiou, T. McLean 2000. Integrated traffic-responsive urban corridor control strategy in Glasgow, Scotland : Application and Evaluation. *Transportation Research Record*, Vol. 1727, pp.101:111.
- [10] Papageorgiou, Markos 1995. An integrated control approach for traffic corridors. *Transpn. Res. -C*, 3(1):19–30.
- [11] Shelby, Steve 2001. *Design and Evaluation of Real-time Adaptive Traffic Signal Control Algorithms*. Ph.D. thesis, University of Arizona, System and Industrial Engineering Department, December 2001.
- [12] Stephanopoulos, G., P. G. Michalopoulos, G. Stephanopoulos 1979. Modeling and analysis of traffic queue dynamics at signalized intersections. *Transpn. Res. -A*, 13(3):295–307.
- [13] Michalopoulos, P. G., G. Stephanopoulos 1977. Oversaturated signal systems with queue length constraints – I single intersection. *Transpn. Res.*, 11(6):413–421.
- [14] Greenshields, B.D. 1934. A study of traffic capacity. *Proceedings of Highway Research Board*, Vol. 14, pp. 448:477.
- [15] Helbing, D. 2003. A section-based queuing-theoretical traffic model for congestion and travel time analysis in networks. *Journal of Physics A: Mathematical and General*, Vol. 36:L593–L598.
- [16] Helbing, D. S. Lämmer, J.-P. Lebacque 2005. Self-organized control of irregular or perturbed network traffic in *Optimal Control and Dynamic Games*, Springer, Dortrecht, pp. 239–274.

- [17] Daganzo, C.F. 1994. The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transpn. Res. -B*, Vol. 28, pp.269:287.
- [18] Daganzo, C.F. 1995. The cell transmission model, part II: network traffic. *Transpn. Res. -B*, Vol. 29, pp.79:93.
- [19] Gomez, G., R. Horowitz 2004a. Globally optimal solutions to the onramp metering problem - part I. *IEEE on ITS'04*, Washington D.C. USA.
- [20] Gomez, G., R. Horowitz 2004a. Globally optimal solutions to the onramp metering problem - part II. *IEEE on ITS'04*, Washington D.C. USA.
- [21] Lo, H. K. 1999. A novel traffic signal control formulation. *Transportation Research, -A*, 33A:433-448.
- [22] Lo, H. K. 2001. A cell-based traffic signal formulation: strategies and benefits of dynamic timing plans. *Transportation Science*, 35(2):148-164.
- [23] Lo, H. K., E. Chang, and Y. C. Chan 2001. Dynamic network traffic control. *Transpn. Res. -B*, 35B:721-744.
- [24] Messmer, A., M. Papageorgiou 1990. METANET: a macroscopic simulation program for motorway networks. *Traffic Engineering and Control*, 31:466-473.
- [25] Kotsialos, A., M. Papageorgiou, M. Mangeals, and H. Haj-Salem 2002. Coordinated and integrated control of motorway networks via nonlinear optimal control. *Transpn. Res. C*, Vol.10:65-84.
- [26] Spall, J. C. 1998. Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3):817:823.
- [27] Zhang, H. M., W. Jin. On the distribution schemes for determining flows through a merge. *Transpn. Res. -B.*, 37(6):521-540.
- [28] Chen, O. J. 1998. *Integration of Dynamic Traffic Control and Assignment*. Ph.D Thesis, Massachusetts Institute of Technology, Department of Civil and environmental Engineering. Cambridge, Massachusetts, USA.
- [29] Yang, H. S. Yagar 1995. Traffic assignment and traffic control in saturated road networks. *Transpn. Res. -B.*, 29 (2):125-139.
- [30] Spall, J. C. 1992. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, 37(3):332:341.
- [31] Sadeh, P. 1997. Constrained optimization via stochastic approximation with a simultaneous perturbation gradient approximation. *Automatica*, 33:889-892.
- [32] Chin, D.C. 1994 Comparative study of stochastic algorithms for system optimization based on gradient approximations. *IEEE Transactions on Systems, Man, and Cybernetics*, part B Vol. 27, pp. 244:249.
- [33] Sadeh, P., J. C. Spall 1998. Optimal random perturbations for multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, 43(3):1480:1484.
- [34] Mahamassani, H., H. Sbayti, X. Zhou 2004. *DYNASMART-P: Intelligent Transportation Network Planning Tool, Version 1.0 User Guide*. Maryland Transportation Initiative, University of Maryland, College Park. MD 20742.
- [35] Wright, A. H. 1991. Genetic algorithms for real parameter optimization. *Foundations of Genetic Algorithms*. Morgan Kaufmann Publishers, San Mateo, California, USA.