□ 論 文 □

A Simulated Annealing Method for Solving Combined Traffic Assignment and Signal Control Problem

통행배정과 신호제어 결합문제를 풀기위한 새로운 해법 개발에 관한 연구

이 승 재
(서울시립대학교 도시공학과 전임강사)

목 차

요 약

본 논문은 통행배정과 교통신호제어의 결합문제를 풀기 위한 새로운 해법의 제시를 목적으로 한다. 통행배정과 신호제어 결합모형은 네트웍 디자인 문제(Network Design Problem)로 비선형 비분리 목적함수(Nonlinear and Nonseparable Objective Function)와 비선형제약 및 비컴백스 집합(Nonlinear and Non-Convex Set)형태로 인해 다수의 국지해(Multiple Local Optima)를 갖는 특징이 있다. 따라서 이렇게 복잡하고 난해한 문제를 푸는 해법은 많은 국지해중에 가장 최소한 값(Global Optima)을 찾을 수 있는 방법을 제공하여야 한다. 전체최적해(Global Optima)를 찾을 수 있는 기존의 방법들은 확률적최적화방법(Stochastic Optimization Methods)에 속한다. 본 연구에서는 이러한 방법중 금속공학에서 발견된 모의담금질법(Simulated Annealing Method)에 근거한 해법을 제시한다. 이 방법이 통행배정과 신호제어 결합문제에 적용되는지 검토하기 위해 이해법의 수렴성(Convergence)을 증명했으며, 또한 실제 프로그램된 모형을 작은 고안된 네트워크에 적용했다. 마지막으로는 개발된 해법의 실용성을 실험하기 위해 두 가지의 보다 큰 도로망에 적용 및 분석을 했다.

I. INTRODUCTION

The traditional equilibrium approach to representing interactions between supply and demand sides of traffic assignment has been used widely in the estimation of traffic flows on road networks. However, there is a considerable gap between the observed and modelled values of generalized travel cost and flow. This gap can be reduced by relaxing some of the restrictive assumptions behind the traditional separable model. For example, weakening traditional monotonicity, separability and symmetry assumptions on link performance functions is worthy of consideration.

Traffic assignment and signal control have played important roles in transportation planning and traffic engineering. Traffic assignment is used to represent users' behaviour in selecting routes so as to minimize their individual travel cost in the system. These problems have been tackled together in order to create models with improved realism. When the traffic assignment problem is solved in detail, signal control data are required in the cost functions. On the other hand, when signal control is solved, traffic flow is necessary to calculate signal control variables.

There are two main approaches to the computation of this combined problem. The first is the global optimization method which solves the amalgamated signal optimization and traffic assignment problem as a whole. The second is the iterative approach which solves signal control and traffic assignment alternately. Both approaches can be described using simplicity decomposition which represents and solves the combined problem by dividing it into two subproblems: a linear and a master subproblem which are solved alternately (for a detailed description, see Lee 1995).

The global approach is known as the network design problem in which the decision variables correspond to signal timings, and flows are constrained to be in equilibrium consistent with these timings. This method suffers from some limitations. First, it is restricted to networks of small size. Second, this method can only be used on somewhat unrealistic representation of road characteristics because it needs strong assumptions on cost functions and on network structure for good behaviour. Even though the iterative method overcomes the limitations of the global method, both methods can guarantee only local equilibria.

Traditional deterministic solution methods for solving this complicated problem cannot guarantee the global solution. However, stochastic solution methods including simulated annealing for this complex model can overcome this difficulty. The simulated annealing methods considered can be viewed as simple adaptations of existing solution methods, The main difference between them is how to calculate a master subproblem. In a simulated annealing method, a line search of master subproblem is determined using a randomly generated step length. This generated step length is tested against an acceptance probability criterion: a generated step length that improves the objective function is accepted unconditionally but a step size leading to a detrimental change in the objective function is accepted according to an auxiliary criterion. The latter part of the acceptance test allows the optimization algorithm to walk out of local optima. This direction of the master problem is searched in an incremental way in order to find a global solution.

The objective of this paper is to develop a simulated annealing method for solving the combined traffic assignment and signal control junction problem. The combined problem has a form of non-convexity so that it cannot be guaranteed that a global solution be obtained using existing deterministic optimization methods. However, the development of a simulated annealing method leads to a technique which can locate a global solution for the combined problem. The iterative approach is used as a general framework. A gap function is employed to provide an objective value, and a related probabilistic acceptance criterion is used for finding a global solution. In order to analyse the method, a simple artificial network is used to test whether the global solution is obtained using a simulated annealing method. Two more larger networks are used to see whether this method is useful in practical cases.

In section 2, formulation on the combined signal control and traffic assignment problem are presented. In section 3, stochastic solution methods are presented. Particularly, this section looks at the feasibility of applying simulated annealing (and related methods of stochastic optimization) to the traffic assignment problem. A simulated annealing algorithm is introduced in this section. In section 4, a global convergence of the simulated annealing method is tested using a small contrived network. Furthermore, empirical studies using two larger networks are presented to indicate whether or not this algorithm is practically viable for the combined signal control and traffic assignment problem. Final comments and conclusions appear in section 5.

II. FORMULATION OF COMBINED SIGNAL CONTROL AND TRAFFIC ASSIGNMENT PROBLEM

We will adopt the mathematical model proposed by Smith (1982) in order to formulate signal controlled junction modelling in road traffic assignment. The network consists of a set of n nodes and a set of L links. Each link has a start and a terminal node. We can group links according to their terminal nodes. Let junction n be

$$J_n = \{ l_{a1}, l_{a2}, \cdots, l_{am_n} \}$$

corresponding to the set of links terminating at node n_a, where m_n is the number of links in junction n. Then the set of links, L is

$$L = J_1 \cup J_2 \cup \cdots \cup J_n$$

In road networks, the journey time is spent at junctions and on links. We therefore represent that the link cost is a weighted combination of running time on links and junction delays. The cost incurred in using link a, $T_a(v)$, is a weighted summation of link cost, $c_a(v_a)$ and junction delay, $d_a(v)$:

$$T_a(v)=t_0+\gamma[c_a(v_a)-t_0]+(1-\gamma)d_a(v)$$

where γ represents a relative magnitude of two delays and a user's perception on travel cost.

In this study, link cost is calculated from the BPR (1965) function steady state formula. Junction delay, $d_a(v)$, depends on the junction type, and is calculated from the approximated Webster's formula.

In this cost function, the link cost, $c_a(v_a)$ is only a function of the flow on link a, whereas the junction delay, $d_a(v)$ is a function of several flows approaching junction a in order to reflect any interacting movements. This relaxation makes the cost function non-separable.

$$\frac{\partial T_a}{\partial v_b} \neq 0 \qquad \text{for some } a \neq b$$

This non-separability can be accommodated by Smith's variational inequality formulation of an equilibrium assignment. Let D and S be the demand and supply feasible sets respectively: These are convex subsets of R^m. Then $v \in S \cap D$ is a Wardrop equilibrium if and only if

$$u \in D$$
$$T(v)(v-u) \leq 0$$

Hearn(1982) established that any solution of the mathematical programme

$$\min_{v \in D} G(v)$$

$$\text{where } G(v) = \max_{u \in D} T(v)(v-u), \ \forall v \in D$$

is a solution of the variational inequality formulation above. It can be shown that the gap function is continuous and differentiable (assuming differentiable cost functions) at each flow v such that there is a unique minimum cost flow with respect to T(v); see Patriksson(1994) for further properties of G(v). Nevertheless, G(v) is not convex in general, and will sometimes have multiple local minima. The result is that standard methods of optimization may fail to converge to v in D. (There is an alternative gap function which is smooth and convex, see Lawphongpanich and Hearn, 1984, for example, but evaluation of this function is itself a non-convex problem.) Throughout the remainder of this paper we will concentrate on the combined signal control and traffic assignment problem as a (non-convex) optimization of G(v). However, the methodologies presented in later sections can equally well be applied to any other non-convex objective function which is minimized by v in D.

III. SIMULATED ANNEALING METHODS

In this section we consider some methods of optimization which incorporate a stochastic element. The basic set up is as follows. Suppose that we wish to find a global minimum of some function G(v) which may have multiple local minima, and is not differentiable everywhere. An algorithm which seeks a direction of descent at each iteration will tend to halt at a local minimum rather than the required global solution. For such an optimization problem it follows that steps in a direction of increased G(v) may be necessary from time to time, whiles still maintaining an overall trend of decreasing objective function. A class of stochastic optimization algorithms (based on simulated annealing and related methods) achieves such behaviour by always allowing steps which decrease G(v), but also accepting steps 'uphill' with non-zero probability. We describe such optimization techniques below, concentrating on simulated annealing and its generalizations. Application of these methods to the problem of calculating Wardrop equilibria is then discussed.

Simulated annealing in combinatorial optimization is based on a simulation of the physical annealing process, which heats up a solid until it melts, followed by cooling it down until it crystalizes into a state with a perfect lattice. During this process, the free energy of the solid is minimized and leads to have crystal perfection. In combinatorial optimization, an analogous process is defined. This process can be formulated as the problem of finding a solution with minimal cost among potentially large number of solutions. This method is able to find an optimum solution.

Kirkpatrick, Gelatt and Vecchi(1983) introduced the concepts of annealing in combinatorial optimization. These concepts are based on an analogy between the physical annealing process of solids and the problem of solving large combinatorial optimization problems. The process contains two steps: increase the heat temperature to a maximum at which the solid melts, and then decrease with care the heat temperature until the particles arrange themselves in the ground state of the solid. In a maximum temperature all particles of the solid arrange themselves randomly. On the other hand, in the ground state the particles are arranged in a highly structured lattice and the energy of the system is minimal.

Binder(1978) reviewed computer simulation methods for the physical annealing process. According to him, Metropolis, Rosenbluth, Rosenbluth, Teller and Teller(1953) introduced a simple algorithm for simulating the evolution of a solid in a heat bath to thermal equilibrium, which is based on Monte Carlo techniques and generates a sequence of states of solid in the following way. Given a current state I of the solid with energy Ei, then a subsequent state j is generated by applying a perturbation mechanism which transforms the current state into a next state by a small distortion by displacement of a particle. The energy of the next state is E_j. If the energy difference, E_j-E_i, is less than or equal to zero, the state j is accepted as the current state. If the energy difference is greater than zero, the state j is accepted with a certain probability as

$$\exp\left[\frac{E_i - E_j}{k_E T}\right]$$

Where T is the temperature of the heat bath and kB is a physical constant known as the Boltzmann constant.

The acceptance rule above is known as the Metropolis criterion and the algorithm that goes with it is Metropolis algorithm. The simulated annealing algorithm can be viewed as an iteration of Metropolis algorithms. The probability of accepting deterioration is implemented by the energy value with a random number generated from a uniform distribution on the interval (0,1). Furthermore, the speed of convergence of the algorithm is determined by the choice of the Boltzmann constant.

Simulated annealing can be viewed as a generalization of local search. As in local search algorithms we assume the existence of a neighbourhood structure and a generation mechanism. Generation mechanism corresponds to the perturbation mechanism in the Metropolis algorithm whereas the acceptance criterion corresponds to the Metropolis criterion.

Now we consider the problem of minimizing G(v) where v in D. In order to implement simulated annealing, we need to define the following:

1. a reflexive neighbourhood structure on D, so that if v in D(z), where D(z) is the neighbourhood of flow pattern z, then z in D(v).
2. a 'candidate' probability distribution $q_v(z)$ for z in D(v), such that $q_v(z)>0$ for all z in D(v)

Suppose that the search is at v. Simulated annealing generates the next flow pattern by first sampling from $q_v(\bullet)$, generating a candidate flow z (say), and then accepts this candidate as the next flow with probability

$$\min\{1, \exp(G(v)-G(z))/\tau\},$$

where τ is a control parameter. Sampling of candidates at v continues until a candidate is finally accepted, or some algorithm termination criterion is met. Assuming the symmetry condition that $q_v(z)=q_z(v)$, it can be shown that the randomly generated sequence of flows v, z, forms a Markov process (see Grimmett and Strizaker, 1982) whose stationary distribution is the Gibbs distribution with density

$$p(v) = Z_\tau^{-1} \exp(-G(v)/\tau)$$

where Z_τ is a normalizing constant. As $\tau \to 0$, the Gibbs distribution becomes concentrated entirely on those flows with minimum G(v); i.e. on D* as required. In practice, a large value of τ is used initially, so as to allow moves uphill with high probability and hence avoid capture in local minima, but τ is systematically reduced as the algorithm continues. Given a suitable sequence of control parameter values, convergence of the algorithm to a flow in D* can be guaranteed, although the rate of convergence may be extremely slow. Further details on simulated annealing can be found in the review by Bertsimas and Tsitsiklis (1993), whilst Bohachevsky et al. (1986) describe its application to general function optimization.

The main difficulty in applying the above methodology to the Wardrop traffic assignment problem is one of computational expense. Simulated annealing is generally a computationally time consuming method to implement, and as such is reserved for non-convex problems which do not admit solution by simpler optimization algorithms. Furthermore, for traffic assignment there is the added difficulty of defining a reflexive neighbourhood structure on the flow space D. One method of doing so would be to consider path flows as opposed to link flows, since the possible results of rerouting a single quantum of flow would then define a neighbourhood. However, generation and storage of all acyclic O/D paths tends to be prohibitively computer intensive (although a restriction to `reasonable' paths only can reduce the burden). On the other hand, if link flow patterns are used, then forming convex combinations in order to define a neighbourhood implies the calculation of many extremal flow patterns at each iteration of the simulated annealing algorithm, which is again computationally demanding. An example of the difficulties to solve the (asymmetric) Wardrop traffic assignment problem by simulated annealing are provided by Hazelton (1995), in which the equilibrium state of a 6 link network required 137600 iterations to locate.

Generalized simulated annealing algorithms can ensure calculation of Wardrop traffic equilibrium, but only at huge computational expense. In one sense this should be of limited concern, since if a model of traffic assignment is sufficiently good then the time required for computation is well spent. However, Wardrop equilibrium is a rather crude model of traffic assignment. Indeed, its attraction over more complicated methods such as stochastic user equilibrium is its simplicity. The use of a highly complicated and time consuming optimization routine in order to compute Wardrop equilibrium would therefore seem somewhat perverse. A more attractive approach is to consider heuristic algorithms which tend to perform well in practice, despite lacking the theoretical justification of more sophisticated methods. Below we describe an adaptation of the generalized simulated annealing which falls into the category of heuristic algorithms.

Consider the following crude model of the dynamic progression of the flow pattern from some initial state to its equilibrium. Suppose that at the present epoch of time, some travellers are aware of routes which are less costly than their own. An idealized behavioural assumption is that a proportion of these travellers will switch to these cheaper alternatives in the next epoch (even though in the act of changing route they may actually encounter increased cost). Furthermore, the number so doing will be a random variable dependent on some measure of improvement in network performance and the volatility of traveller behaviour. If (and only if) the proportion of travellers changing journey path decreases as the relative attractiveness of alterative routes drops, then there is reason to believe that the transport system may settle down to a steady state. Our heuristic algorithm mimics this process. In other words, an iteration of this dynamic adjustment process for solving this combined signal controlled junction modelling and traffic assignment is the combination of the calculation of green times for the given feasible traffic flows, and the assignment of travel demand using these green times. The dynamic adjustment process continues until a convergence criterion is satisfies.

Simulated annealing algorithm for the combined signal control and traffic assignment

Step 0 : (Initialization)

Iteration n = 0

Identify a feasible point, $v^n \in D$

This is most easily done by defining v^0 to be the minimum cost flow pattern with respect to cost vector at zero flow, $T(v^0)$.

Step 1 : (Linear subproblem)

1-1. Calculate green times based on the Webster's min-max policy :

$$\lambda_a = \frac{(C - L)\, v_a / s_a}{C \sum_k v_k / s_k}$$

1-2. Calculate the total link cost

$$T_a(V^n, \lambda_a^n) = t_0 + \gamma\,[c_a(v_a^n) - t_0] + (1-\gamma)\,\overline{d_a}(v^n, \lambda_a^n)$$

where $c_a(v_a^n)$ is the BPR function and $\overline{d_a}(v^n, \lambda_a^n)$ is the approximated Webster's formula

1-3. Perform all-or-nothing assignment based on the total link cost

$$u^n = \arg\min_u \{T(v^n, \lambda^n)\, u \,;\, u \in D\}$$

1-4. Convergence test

Calculate the relative gap $G(v^n) / T(v^n)^t v^n$, and terminate algorithm if this is less than a prescribed ε (typically 10^{-2} or 10^{-3}).

Otherwise go to step 2

Step 2 : (Master subproblem)

2-1. Randomly select α in (0,1), and form the candidate new flow,

$z^n = (1 - \alpha)v^n + \alpha u^n$.

(This is a feasible pattern being a convex combination of feasible flows; it is the result of a proportion alpha of travellers changing to the cheapest routes.)

2-2. Accept the candidate flow with probability

$\min\{ 1, \exp\{(G(v^n)-G(z^n)) / \tau\} \}$.

If flow zn is rejected, go to 2-1.

(Here, the change in gap function, $G(v^n)-G(z^n)$, is being used as an assessment of aggregate improvement in network performance, whilst the control parameter τ is a measure of travellers' inertia to change.)

2-3. Put n = n+1. Define $v^n = z^{n-1}$. Go to step 1.

This algorithm is similar to the generalized simulated annealing algorithm discussed previously, but differs in the manner in which candidate flow patterns are generated. Whilst generalized simulated annealing allows the possibility of movement to any neighbouring state, the developed algorithm only allows steps in the direction of the minimum cost flow to be taken. The result is that no reflexive neighbourhood structure is even implicitly defined. An important result is that convergence of the algorithm is not guaranteed under the most general conditions. However, convergence is ensured if there is any realization of the dynamic rerouting process (described above) which (a) leads to an equilibrium flow, and (b) is robust to arbitrarily small perturbations to the proportions a, in the sense that equilibrium will still eventually be attained given such small changes. These ideas will be formalized below. This sufficient condition for convergence is very weak; certainly no more restrictive than the conditions required for convergence of other algorithms such as column generation (see Smith,1983a, for example) and parallel decomposition (see Mahmassani and Mouskos, 1988, for example). In particular, monotonicity of cost-flow functions is not essential. Indeed, it is worth considering whether or not we should be interested in calculating the equilibria of models which do not satisfy the convergence requirement for the developed algorithm. For such models it is difficult to see how Wardrop equilibrium could have developed, given that (almost) no sequence of traveller switches to cheaper routes can lead to equilibrium.

A formal mathematical expression of the sufficient condition for attainment of equilibrium is as follows.

Assumption Given any initial flow pattern v^0, there exists a sequence of intervals A1, A2,... of non-zero Lebesgue measure such that the sequence of flows, v^i, defined by $v^{i+1} = (1 - a_i)\, v^i + a_i\, u^i$ converges to Wardrop equilibrium for any set of $\{a_i\}$ such that $a_i \in A_i$ all i.

Theorem Under assumption, and given that in step 2-1 of the developed algorithm is drawn from a probability density which is non-zero almost everywhere on $a \in (0,1)$, then the developed algorithm converges to Wardrop equilibrium with probability 1.

Proof Let $\varepsilon > 0$ be given. Now, for each $v \in D$, define $p_{n(v),v} = P(\ |\ V^{n(v)} - v^*\ | < \varepsilon)$, where $V^{n(v)}$ is the random variable denoting the flow pattern generated by the developed algorithm after n iterations, having started at v. By the definition of the probability distribution of , the probability $P(a_1 \in A_1,\ a_2 \in A_2, \cdots, a_n \in A_n)$ is non-zero. By the assumption, for each $v \in D$ there exists $n = n_{(v)}$ such that for $a_1 \in A_1, a_2 \in A_2, \cdots, a_n \in A_{n(v)}$, we have $|\ V^{n(v)} - v^*\ | < \varepsilon$. But the probability of this event is non-zero. Hence, $p = \min\{p_{\max\{n(v)\},v}\} > 0$. It follows that the probability of not having attained Wardrop equilibrium after N max $\{n_{(v)}\}$ iterations of the algorithm is not greater than $(1-p)^N \rightarrow 0$ as $N \rightarrow \infty$, which completes the proof

Whilst this theorem does ensure convergence to Wardrop equilibrium under very general conditions, it is in fact only demonstrating that the algorithm retains traces of exhaustive search over an arbitrarily fine discretization of D. Of course, such an exhaustive search procedure is not a practical proposition for calculating Wardrop equilibrium. The crucial question is whether or not the developed algorithm performs well in practice. We address this point by analyzing the efficacy of the algorithm on some numerical examples in the next section.

Before moving on to empirical results on algorithm performance, two issues need to be addressed. First, the manner in which candidate $a \in (0,1)$ are generated, and secondly, the choice and progression of the control parameter, τ. Whilst a small value of may delay the algorithm by taking several steps where larger one would suffice, too large a value will result in large number of candidate states with huge gaps being generated and summarily rejected. Empirical evidence has suggested that generating alpha adaptively, as realizations of beta random variables with mean $G(v)/10T(v)^t v$ is sensible. Shrinking the choice of alpha will be required should a very high proportion of candidates be rejected. A suitable choice for is $T(v)^t v/100$, although careful fine tuning for the particular problem in hand can be very beneficial. As a general pointer, the more complicated the large scale the problem, the smaller the value of α (and τ) that is appropriate.

IV. NUMERICAL EXAMPLES

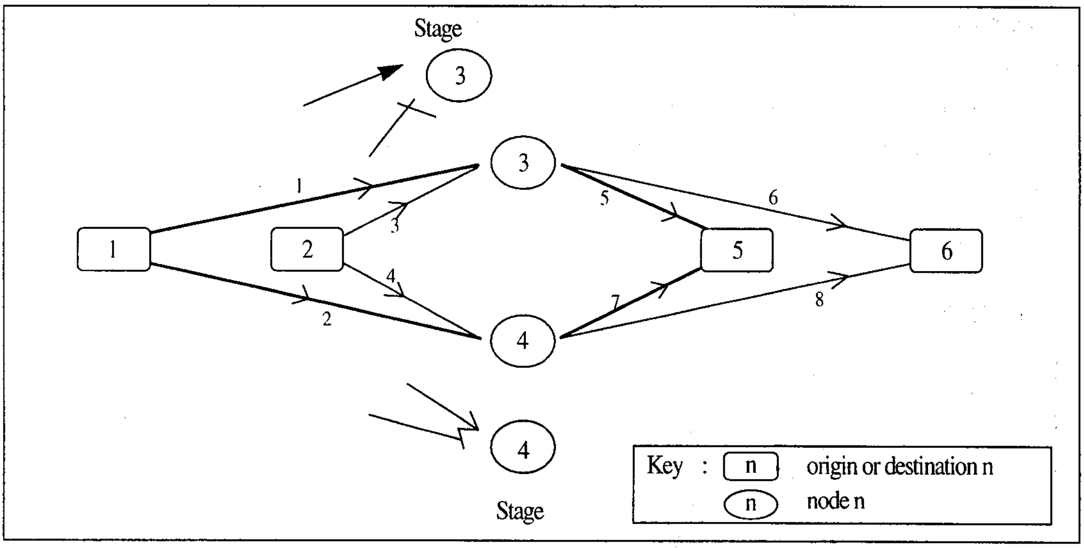
1. Small example network

We now use a simple symmetric network (see Figure 1) which has 2 origins and 2 destinations, 6 nodes and 8 links. The demand from origin 1 is entirely to destination 5 and from origin 2 to destination 6. Each of nodes 3 and 4 represents a signal controlled junction. The capacity of each link is 2000 vehicles / hour. We assume that signal is operated in two stages. The cycle time is taken as fixed at 60 seconds with a lost time of 4 seconds following each stage. Minimum green for each stage is 7 seconds and maximum green time for each stage is therefore 45 seconds. We vary the demand for each origin-destination pair in order to test if the developed algorithm obtains the global solution and is robust in all the demands. In this example, an equilibrium solution of green time and traffic flows is clearly provided by the symmetric values:

$$v^* = (v^*1, v^*2, v^*3, v^*4) = 1/2(T_{15}, T_{15}, T_{26}, T_{26})$$
$$g^*(v^*) = (g^*1(v^*), g^*2(v^*), g^*3(v^*), g^*4(v^*))$$
$$= (26, 26, 26, 26)$$

Tables 1 and 2 show the symmetric equilibrium traffic flows and green times respectively which are calculated at various levels of demands T_{15} and T_{26}. The lower-left parts of the tables are empty because the values of the upper-right parts are symmetric to those of the lower-left parts due to the symmetric example network.

In order to test if the developed algorithm obtains the global solution (here symmetric equilibrium solutions), calculated values by the simulated annealing algorithm are compared with the symmetric equilibrium value in the



〈Figure 1〉 Example network

〈Table 1〉 The symmetric equilibrium traffic flows

O/D	T_{26}					
	v*1/v*3	200	800	1400	2000	2600
T_{15}	200	100/100	100/400	100/700	100/1000	100/1300
	800	400/100	400/400	400/700	400/1000	400/1300
	1400	700/100	700/400	700/700	700/1000	700/1300
	2000	1000/100	1000/400	1000/700	1000/1000	1000/1300
	2600	1300/100	1300/400	1300/700	1300/1000	1300/1300

⟨Table 2⟩ The symmetric equilibrium green times

O/D	T_{26}					
	v^*1/v^*3	200	800	1400	2000	2600
T_{15}	200	26/26	10/42	7/45	7/45	7/45
	800	42/10	26/26	19/33	15/37	12/40
	1400	45/7	33/19	26/26	21/31	18/34
	2000	45/7	37/15	31/21	26/26	23/29
	2600	45/7	40/12	34/18	29/23	26/26

small example network derived above. We can see whether or not the calculated values correspond to the symmetric equilibrium value. In this test, the value $\gamma = 0$ (maximum junction effect) is used.

We obtain the calculated green time on the link by the developed algorithm shown in Table 3. We compare the calculated traffic flows from Table 3 with those symmetric values in Table 2 to test the global solution. In this case, the symmetric assignments were obtained. This test shows that this combined signal control and traffic assignment model using the developed algorithm appears to have global symmetric equilibrium solutions in the various demands. The maximum iteration was 50 and all the run took the maximum number of iteration.

⟨Table 3⟩ The calculated green times by the developed algorithm

O/D	T_{26}					
	g^n1/g^n3	200	800	1400	2000	2600
T_{15}	200	26/26	10/42	7/45	7/45	7/45
	800	42/10	26/26	19/33	15/37	12/40
	1400	45/7	33/19	26/26	21/31	18/34
	2000	45/7	37/15	31/21	26/26	23/29
	2600	45/7	40/12	34/18	29/23	26/26

2. Two larger example networks

In this section, the developed solution method is applied to two example networks each of which includes several signal controlled junctions. As the first example, the network and travel demand by Charlesworth (1977) are used here: this has 5 origins, 5 destinations, 46 nodes and 78 links (for the figures and data of the network, see Lee, 1995). The Sioux Falls network (see Lee, 1995) is used as the second example: this has 12 origins, 12 destinations, 23 nodes and 72 links. In each case, Webster' s method is used to obtain green times. As stopping criteria for each algorithm, either a required level of the relative gap value, $(G(v^n)/T(v^n)v^n<10^{-3})$ or a maximum number of iterations (50) are used.

The diagonalisation of the Frank-Wolfe (see, Lee 1995) and the developed algorithm have been tested to compare their performances. In this test, the weighting factor γ, which is used to combine junction and link delays, is varied from 0.0 to 1.0 in increments of 0.25. Table 4 shows that the final values for each algorithm are compared in terms of the iteration number, CPU time on SUN Sparc station and the relative gap value as the weighting factor γ is varied on the Charlesworth network. In the same way, Table 5 shows the final values of each algorithm on the Sioux Falls network.

In these examples, the developed algorithm runs quickly and reach a good level of the convergence. Nonetheless, in some cases, the algorithm cannot reach their zero gap values. However, the algorithm terminates with a gap value that is smaller than that achieved by the Frank-Wolfe algorithm, although the Frank-Wolfe algorithm performs well in these cases.

〈Table 4〉 Performance of algorithms in Charlesworth's network

γ (Junction effect)	Algorithms	Iteration	CPU(Second)	Gap value
0.00	Frank-Wolfe	50	18.41	7.18
	Simulated Annealing	50	13.97	4.86
0.25	Frank-Wolfe	50	18.20	5.5
	Simulated Annealing	50	13.82	3.5
0.50	Frank-Wolfe	50	18.19	3.77
	Simulated Annealing	50	13.84	2.35
0.75	Frank-Wolfe	50	18.16	1.92
	Simulated Annealing	50	13.85	1.15
1.00	Frank-Wolfe	50	17.73	0.0
	Simulated Annealing	50	13.78	0.0

〈Table 5〉 Performance of algorithms in Sioux Falls network

γ (Junction effect)	Algorithms	Iteration	CPU(Second)	Gap value
0.00	Frank-Wolfe	50	8.90	0.03
	Simulated Annealing	50	4.27	0.00
0.25	Frank-Wolfe	50	8.89	0.02
	Simulated Annealing	50	4.26	0.00
0.50	Frank-Wolfe	50	8.86	0.01
	Simulated Annealing	50	4.28	0.00
0.75	Frank-Wolfe	50	8.78	0.00
	Simulated Annealing	50	4.33	0.00
1.00	Frank-Wolfe	50	8.66	0.00
	Simulated Annealing	50	4.26	0.00

The convergence pattern of the developed algorithm fluctuates as the iteration proceeds whilst that of the diagonalisation of the F-W algorithm is smooth. The pattern fluctuates more as the weighting value, γ, approaches zero, which gives the maximum junction effect. Because the weighting factor in the cost function represents the degree of junction effects, it is indicative of the degree to which junction interactions are influential. As the junction effect increases, the non-separability increases so that the final gap value is higher. On the other hand, when there is no junction effect, the cost function becomes separable and the final gap value is closer to zero. In particular, when the weighting value is unity which is the case of link cost only, the gap value of each new algorithm converges to zero.

The pattern of convergence in the Charlesworth network shows more fluctuations than in the Sioux Falls one. This can be explained according to the characteristics and structure of each network. Charlesworth' s network has two main roads that provide alternative routes: choice between them can then cause gap values to fluctuate. On the other hand, the Sioux Fall network has a grid form so that many route choices are possible and thus the demand is spread more widely and the fluctuation reduces.

V. DISCUSSION

Computation of Wardrop equilibrium traffic flows has been the subject of much research. Efficient algorithms are available given certain assumptions, but none can guarantee convergence to an equilibrium flow pattern when the cost functions are non-monotone and have an asymmetric structure. In this paper, a method of stochastic optimization is applied to the combined signal control and traffic assignment problem. The developed algorithm which is certain to converge to the desired solution under general conditions is derived, and its performance is studied on some numerical examples and is compared with the diagonalisation of the Frank-Wolfe algorithm. The results of the numerical analyses show that the performance of the developed algorithm is better than that of the F-W algorithm in terms of CPU time and the gap values. Furthermore, in the example of the small contrived network, the developed algorithm obtains the global symmetric equilibrium values.

The developed algorithm seems capable of providing solutions to asymmetric, non-monotone traffic assignment problems which are notoriously complex and difficult to solve. Indeed, a stochastic approach to this type of assignment (combined signal control and traffic assignment) is quite natural, since such methods work well in cases (like the present one)where relatively little can be usefully said about the structure of the problem.

REFERENCES

Aarts, E. and Korst, J. (1989). Simulated Annealing and Boltzman Machines. John Wiley & Sons.

Bertsimas, D. and Tsitsiklis, J. (1993). Simulated annealing. Statistical Science, 8, pp.10~15.

Binder, K. (1978). Monte Carlo Methods in Statistical Physics, Springer-Verlag, New York.

Bohachevsky, I. O., Johnson, M. E. and Stein, M. L. (1986). Generalized simulated annealing for function optimization. Technometrics, 28, pp.209~217.

Frank, M. and Wolfe, P. (1956). An algorithm for quadratic programming, N. R. L. Q., 3, pp.95~110.

Grimmett, G. and Stirzaker, D. (1982). Probability and Random Processes. Oxford : Clarendon Press.

Hazelton, M.L. (1995). The feasibility of Wardrop traffic assignment by stochastic optimisation. Unpublished, Dept of Statistical Science, University College London.

Hazelton, M.L., S. Lee and J. Polak (1996). Stationary states in stochastic process models of traffic assignment: A Markov Chain Monte Carlo Approach. Proc. of the 13th Int. Sym. on Tramp. and Traffic Theory.

Hearn, D. W. (1982). The gap function of a convex program, Operations Research Letters, 1, pp.61~71.

Kirkpatrick, S., Gelatt, J. and Vecchi, M. P. (1983). Optimization by simulated annealing, Science, 220, pp.671~680.

Lawphongpanich S. and Hearn D. W. (1984). Simplical decomposition of the asymmetric traffic assignment problem, Transportation Research B, 18, pp.123~133.

Lee, S. (1995). Mathematical programming algorithms for equilibrium road traffic assignment. PhD thesis, University of London.

Mahmassani, H. S. and Mouskos, K. C. (1988). Some numerical results on the diagonalization algorithm for network assignment with assymetric interactions between cars and trucks, Transportation Research B, 22, pp.275~290.

Marcotte, P. (1983). Network optimisation with continuous control parameters. Transp. Sci. 17, pp.181~197.

Metropolis, N., A. Rosenbluth, M. Rosenbluth, A. Teller and E. Teller (1953). Equation of state calculations by fast computing machines. J of Chemical Physics 21, pp.1087~1092.

Smith, M. J. (1982). Junction interactions and monotonicity in traffic assignment. Transportation Research 16B, 1, pp.1~3.

Wardrop, J. G. (1952). Some theoretical aspects of road traffic research. Proceedings of the Institute of Civil Engineering Part Ⅰ, 1, pp.325~378.