第 59 回土木計画学研究発表会・講演集

Improving The Applicability of Sensitivity Analysis for Stochastic User Equilibrium Traffic Assignment based on Depth First Search

Tien Thiem BUI[1], Shoichiro NAKAYAMA[2], Hiromichi YAMAGUCHI[3]

[1]Student member of JSCE, PhD student, Graduate school of Natural Science and Technology, Kanazawa University
(Kanazawa University, Kakuma-machi, Kanazawa 920-1192, Japan)
E-mail:buitienthiem@stu.kanazawa-u.ac.jp
[2]Member of JSCE, Professor, Faculty of Geosciences and Civil Engineering, Kanazawa University
(Kanazawa University, Kakuma-machi, Kanazawa 920-1192, Japan)
E-mail:nakayama@staff.kanazawa-u.ac.jp
[3]Member of JSCE, Assistant Professor, Faculty of Geosciences and Civil Engineering, Kanazawa University
(Kanazawa University, Kakuma-machi, Kanazawa 920-1192, Japan)
E-mail: hyamaguchi@se.kanazawa-u.ac.jp

Stochastic user equilibrium (SUE) traffic assignment plays an important role in the system of traffic assignment model. However, repeatedly SUE traffic assignment for large-scale traffic networks requires a significant computational cost. With the order of reducing calculation costs while ensuring accuracy of traffic assignment results, here this paper proposes a method for sensitivity analysis for SUE traffic assignment with link-based approach. Our method is developed from STOCH3 algorithm with Method of Successive Averages (MSA), Depth First Search algorithm (DFS) and Ying and Miyagi results. By approaching from the derivative of the implicit function, the derivative of the link flow variable with respect to some parameters has been calculated effectively both in terms of computational cost and accuracy. Then, we verify effectiveness of the calculation method with illustrative example and case study in Kanazawa City road network.

Key Words : *Traffic assignment, Stochastic user equilibrium, Sensitivity analysis, Depth First Search, STOCH3 algorithm.*

1. INTRODUCTION

Traffic assignment, following trip generation, trip distribution and mode choice in the conventional urban transportation modelling, deals with the choice of paths between origins and destinations in transport network. The traffic assignment problem is a central part of transportation analysis. In general, traffic assignment for large-scale road networks requires a huge amount of computational cost. In addition, the mathematical problem with equilibrium constraints (MPEC) (e.g., an optimal toll decision[1,2,3]) call for repeated traffic assignment. Thus, it is very important to reduce the computational cost while ensuring the accuracy of traffic assignment results. Besides, since there are numerous uncertainty parameters in functions (e.g. link performance functions) providing a mathematical model for traffic network equilibrium analysis, it is also important to justify the strength of the model to such uncertainties and to recognize those parameters to which the equilibrium solution is most sensitive. To solve two problems, sensitivity analysis for traffic network equilibrium problems is taken into consideration.

To our knowledge, the research to date on computational methods and applications of sensitivity analysis has been mainly for the deterministic user equilibrium (DUE) traffic assignment problems. DUE assumes that each traveller has perfect information about actual travel conditions and seeks to minimize cost associated with their path choice (Wardrop[4], 1952; Sheffi[5], 1985). A long-familiar method for sensitivity analysis for DUE traffic assignment was proposed by Tobin and Friesz[6] (1988). Kyparisis[7,8] (1988, 1990) and Qui and Magnanti[9] (1989), among many others, developed sensitivity analysis for general variational inequalities. Based on the method of Tobin and Friesz[6], Yang[10] (1995) introduced sensitivity analysis for queuing equilibrium network flow and the derivatives of equilibrium link flows and equilibrium queuing times with respect to traffic control parameters was presented. Sensitivity analysis for others expansion of the DUE traffic assignment has also been proposed and applied for solving network design (Kim and Suh[11], 1990), bi-level traffic control (Yang et al.[12] 1994; Yang and Yagar[13], 1994), congestion pricing problems (Yang and Lam[14], 1996; Yang[15] 1997; Yang and Bell[16] 1997).

Daganzo and Sheffi[17] (1977) proposed the principle of stochastic user equilibrium (SUE) in which no traveler can improve his or her stochastically "perceived" travel cost by unilaterally changing paths. SUE model could be divided into two particular interest: the logit model (Dial[18], 1971; Bell[19], 1995) and the probit model with Monte-Carlo procedure (Burrel[20], 1968; Daganzo and Sheffi[17]). The logit model is more practical with both a remarkably efficient fixed time assignment procedure preventing path enumeration (STOCH algorithm), as well as a convex minimization formulation with a closed-form objective function (Fisk[21], 1980). The result of the SUE is more reasonable, realistic and general than DUE (Sheffi[5]). Sensitivity analysis for the SUE model is also an important problem. While Clark and Watling[22] (2000) proposed the mathematical programming method for sensitivity analysis of Probit-based SUE model, Ying and Miyagi[23] (2001) presented the research on sensitivity analysis of logit-based SUE model with dual approach. Moreover, sensitivity analysis of the link-capacitated SUE model was conducted by Kui Ji et al. [24] (2017).

The advantages of logit model and sensitivity analysis method (Ying and Miyagi[23]) have stimulated our research. The logit-based stochastic network loading algorithms obviating path enumeration (Dial[18]) with MSA algorithm have been widely used. Dial[18] proposed STOCH and STOCH2 procedure to assign flows to "reasonable" paths connecting each OD pair. While STOCH algorithm (double-pass algorithm) define a path is efficient (reasonable) if it includes only links that take the traveller further away from the origin and closer to the destination, STOCH2 algorithm (single-pass algorithm) considers a reasonable path as a path includes only links that their initial node closer to the origin than are their final node. However, these definitions are consistent only with respect to fixed travel times (Leurent[25], 1997) and cannot be used when travel time changes from one iteration to the next. The use of Dial's algorithm causes the problem when the stochastic loading is performed as part of a SUE algorithm in which link costs are flow-dependent. Link costs vary through the iterative process and the efficient path set for each origin may vary. As a consequence, the convergence of the iterative process cannot be guaranteed. Leurent[25] proposed the method to curb the problem in which a modified definition of efficient path Based on fixed reference travel cost, he introduced STOCH3 algorithm and the definition of STOCH3-efficient path. A path is called 'STOCH3-efficient' (or reasonable, or available) if it does not include the same node more than once, if every link has its initial node closer to the origin than its final node, and if every link is "reasonable enough" compared to a reference shortest path. Adopting the STOCH3-efficient path, Leurent[25] performed the effectiveness of the STOCH3 logit model using the MSA and concluded that path identification lead to better behavioural models and computationally useful.

Sensitivity analysis for stochastic user equilibrium traffic assignment was proposed by Ying and Miyagi[23]. Based on dual formulation (Daganzo[26], 1982), Ying et al. used the different notations of derivative and "apparent" derivative of link travel time and link flow with respect to arguments and calculated them. The computation procedure with Dial's traffic assignment and MSA algorithm was also presented. Nevertheless, the weakness of Dial's algorithm with the instability of the reasonable path definition was shown by Leurent[25]. The set of reasonable paths at equilibrium states will be different depending on the number of iterations and convergence criteria that makes the equilibrium states are significantly different when the parameters of the model change. Therefore, it is difficult to keep computational accuracy of the sensitivity analysis method when applying it into real network. In addition, the mathematical program formulations of this sensitivity analysis method confused practice users with the definitions of derivative and "apparent" derivative and made difficult with the calculation of some formulations.

In the process of applying STOCH3 algorithm, we discovered that Depth First Search algorithm (DFS) can be used to list all implicit paths. DFS is an algorithm for traversing tree data structures. When the traffic network is represented as adjacent links instead of matrices and lists types, the algorithm starts at the origin node, explores along each link and ends at the destination node before backtracking. The algorithm is used with recursive technique. From that discovery, we can think of expressing the logit-based stochastic network loading obviating path enumeration with DFS algorithm and using this algorithm to solve the sensitivity analysis problems with high efficiency.

In this paper, we present two developments which make the logit-based stochastic network loading obviating path enumeration and sensitivity analysis of logit-based SUE model more effective, easier to understand and apply into real network. First, we propose DFS algorithm with MSA to solve SUE model. Second, we design the mathematical formulations of the sensitivity analysis with derivatives of implicit function approach.

The remainder of this paper is organized as follows. In the next section, some notations and assumptions are demonstrated. While our DFS algorithm development for solving the logit-based stochastic user equilibrium problem is discussed in the third section, the sensitivity analysis method with link-based approach is formulated in the fourth section. In order to exhibit the correctness and applicability of our methods, section 5 provides numerical examples including the case study in Kanazawa road network. Finally, some remarks are summarized in the sixth section.

2. NOTATIONS AND ASSUMPTIONS

Notations and assumptions used in this paper are as follows:

rs: An origin-destination (OD) pair

$H = \{rs, \ldots\}$: The set of OD pairs

$K_{rs} = \{k, p, \ldots\}$: Set of paths connecting an origin and destination.

q^{rs}: The travel demand of an OD pair

p_k^{rs}: The probability of choosing the path k

f_k^{rs}: The flow of the path k

θ: The positive parameter of logit model.

c_k^{rs}: The travel time of the path k

$N = \{i, j, \ldots\}$: The set of nodes

$A = \{ij, gh, \ldots\}$: The set of links

$\delta_{ij,k}^{rs}$: The link-path incidence variable. If the path k includes link ij then $\delta_{ij,k}^{rs} = 1$, otherwise $\delta_{ij,k}^{rs} = 0$.

t_{ij}: The travel time of link ij. It is assumed that t_{ij} is continuous and non-decreasing function of x_{ij} and uncertainty parameter ε_{ij}. By using a conventional traffic model based on BPR (Bureau of Public Roads) curves, we have

$$t_{ij} = (t_{ij}^0 + \varepsilon_{ij})\left[1 + \alpha\left(\frac{x_{ij}}{Cap_{ij}}\right)^{\beta}\right] \tag{1}$$

t_{ij}^0: The free-flow travel time

α, β: The BPR function parameter

x_{ij}: The travel flow of link ij

Cap_{ij}: The traffic capacity of link ij

$\nabla_x \mathbf{t}$ and $\nabla_{\boldsymbol{\varepsilon}} \mathbf{t}$: The diagonal matrix of partial derivatives of t_{ij} as an explicit function with respect to x_{ij} and ε_{ij}, respectively.

$$\nabla_x \mathbf{t} = \begin{pmatrix} \frac{dt_{ij}}{dx_{ij}} & & 0 \\ & \ddots & \\ 0 & & \frac{dt_{|A|}}{dx_{|A|}} \end{pmatrix} \tag{2}$$

$$\nabla_\varepsilon \mathbf{t} = \begin{pmatrix} \frac{dt_{ij}}{d\varepsilon_{ij}} & & 0 \\ & \ddots & \\ 0 & & \frac{dt_{|A|}}{d\varepsilon_{|A|}} \end{pmatrix} \tag{3}$$

x_{ij}^{rs}: The link flow of an OD pair.

x_{ij-gh}^{rs} : The number of travellers from r to s who choose some paths which contain both links ij and gh in such a way that link ij is used prior to link gh.

$x_{ij,gh}^{rs}$: The number of travellers from r to s who choose some paths which contain both links ij and gh without consideration of priority.

$$x_{ij,gh}^{h} = x_{gh,ij}^{h} = \begin{cases} x_{ij-gh}^{rs} + x_{gh-ij}^{rs}, & ij \neq gh, \\ x_{ij}^{rs}, & ij = gh. \end{cases} \tag{4}$$

p_{gh}^{js}: The fraction of the number of travellers from j to s who trace link gh.

C_{rn}^0: The reference shortest generalized travel time from origin r to node n, based on the link travel time $\{t_{ij}^0\}$.

$q^{rs}(\gamma^{rs})$: The travel demand of an OD pair that depends on a perturbation γ^{rs}. With q_0^{rs} being a constant, assuming that

$$q^{rs}(\gamma^{rs}) = q_0^{rs} + \gamma^{rs} \tag{5}$$

$\nabla_{\boldsymbol{\gamma}} \mathbf{q}$: The diagonal matrix of partial derivatives of q^{rs} as an explicit function with respect to γ^{rs}. Clearly, with $\mathbf{I}$ denotes unit matrix, we have $\nabla_{\boldsymbol{\gamma}} \mathbf{q} = \mathbf{I}$

C_{rn}^0: The reference shortest generalized travel time from origin r to node n, based on the link travel time $\{t_{ij}^0\}$

h_{ij}^r: STOCH3 parameter

Ω_{ij}^{rs}: Indicator variable ($\Omega_{ij}^{rs} \coloneqq 1$ if link ij is reasonable from r to s and $\Omega_{ij}^{rs} := 0$ otherwise)

a_{ij}: Impedance of link ij

WL_{ij}: Link weight that presents the importance of link ij in contributing to a reasonable path

WN_n: Node weight

SWN: Summary of node weight at destination node.

3. DEVELOPMENT FOR SOLVING THE LOGIT-BASED STOCHASTIC USER EQUILIBRIUM PROBLEM

Logit model is one of the discrete choice models that based on the random utility theory to choose a single choice among several choices. In logit-based traffic assignment model, path choice is based on the logit model. A logit-based path choice model can be seen as

$$f_k^{rs} = q^{rs} p_k^{rs} = q^{rs} \frac{\exp(-\theta c_k^{rs})}{\sum_{p \epsilon K_{rs}} \exp\left(-\theta c_p^{rs}\right)} \quad (6)$$

where θ is a positive parameter that depicts the perceived travel time. If $\theta \rightarrow 0$, the perception error is large and the SUE assignment is totally random, regardless of path travel time; while if $\theta \rightarrow \infty$ the probability of choosing the shortest path of a driver is high and the SUE assignment converges to deterministic user equilibrium assignment (Sheffi[5]).

And, the travel time of path k is the total travel of all links that belong to it.

$$c_k^{rs} = \sum_{ij \in A} t_{ij} \delta_{ij,k}^{rs} \quad (7)$$

The link flow is given as

$$x_{ij} = \sum_{rs \in H} \sum_{k \in K_{rs}} f_k^{rs} \delta_{ij,k}^{rs} \quad (8)$$

When link time-flow functions such as the well-known BPR functions are used, the solution to the SUE problem is iterative in nature. The standard MSA is therefore directly applied as the solution method to avoid the evaluation of the objective function. With the link flows $x_{ij}^{(l)}$ at iteration l, giving rise to a set of link travel times $t_{ij}(x_{ij}^{(l)})$ which are used in a stochastic loading to produce an auxiliary flow pattern $y_{ij}^{(l)}$. The flow pattern is the updated by the formulation

$$x_{ij}^{(l+1)} = x_{ij}^{(l)} + \alpha^{(l)}(y_{ij}^{(l)} - x_{ij}^{(l)}) \quad (9)$$

In which $\alpha^{(l)}$ is the step length taken along the search direction $(y_{ij}^{(l)} - x_{ij}^{(l)})$. The most frequently predetermined step length $\alpha^{(l)} = 1/(l+1)$ is used. At SUE state, the auxiliary and the current solutions become identical.

As was mentioned in the first section, in order to avoid complete path enumeration in a logit stochastic loading, Dial[18] proposed the use of an efficiency algorithm which only use node-based and link-based variables. By sorting the shortest path in order of increasing access time from particular origin, while "forward" technique calculated variables in the order of increasing access time, "backward" technique used the order of decreasing access time to work out the auxiliary link flows. Leurent[25] also used these techniques even though fixed path choice set with reference link travel time (such as free-flow travel time pattern). The use of this techniques allow the logit stochastic loading to run on each origin, which reduces the calculation time in case there are many OD pairs having the same origin. However, in many cases, the logit stochastic loading must to run on each OD pair, calculation time is a problem when using these techniques. For example, when the definition of efficient path needs to mention the destination (STOCH efficient path) or formulations of sensitivity analysis method are calculated for each OD pair (Ying and Miyagi[23]). In addition, we propose the definition of STOCH4 efficient path that is the extension of STOCH3 efficient path. A path is called "STOCH4-efficient'' (or reasonable, or available) if it does not include the same node more than once, if it includes only links that take the traveler further away from the origin and closer to the destination, and if every link is "reasonable enough" compared to a reference shortest path. This definition is a combination of STOCH and STOCH3 efficient path definitions. Compared to the definition of STOCH3 efficient path, the definition of STOCH4 efficient path adds the condition of every link on efficient path has its final node closer to the destination than its initial node.

To solve this problem, we will propose a new algorithm to present stochastic network flow loadings based on DFS algorithm. Instead of relying on STOCH loading, we will use the DFS algorithm. We will call it DFS loading procedure. MSA will also be used to achieve equilibrium in SUE model with this algorithm.

The DFS loading procedure with MSA method is showed as follows

Step 0: Preliminaries

(a) Set iteration counter $l := 0$ and maximum value of the change (ε) in link flow from the previous iteration $l-1$ to the current one l

(b) Based on the free flow travel time $\{t_{ij}^0\}$, for each OD pair rs, calculate the minimum travel time to all other nodes.

(c) For each link ij, set $a_{ij} := \exp(-\theta\, t_{ij}^0)$ and $\Omega_{ij}^{rs} := 1$ if $(1 + h_{ij}^r)(\mathrm{C}_{rj}^0 - \mathrm{C}_{ri}^0) \geq t_{ij}^0$, other wise $\Omega_{ij}^{rs} := 0$ and set $t_{ij} = t_{ij}^0$

(d) Set all WL_{ij} and WN_n to 0. Set WN_r:=1 and $SWN = 0$. Running DFS algorithm from origin node r to destination node s with recursive technique. After each link and node with nonzero value of Ω_{ij}^{rs} is queried we update $WN_j = a_{ij}WN_i$. When reaching destination node s, we add WN_s to WL_{ij} if link ij is queried from r to s and add WN_s to SWN.

$$WN_s = \prod_{ij=1}^{|A|} \exp(-\theta\, t_{ij}^{0})\delta_{ij,k}^{rs} = \exp(-\theta c_k^{rs}) \quad (10)$$

With recursive technique, we can reach destination node s by $|K_{rs}|$ times. And,

$$SWN = \sum_{k=1}^{|K_{rs}|} WN_s^{(k)} = \sum_{k=1}^{|K_{rs}|} \exp(-\theta c_k^{rs}) \quad (11)$$

$$WL_{ij} = \sum_{k=1}^{|K_{rs}|} WN_s^{(k)}\delta_{ij,k}^{rs} = \sum_{k=1}^{|K_{rs}|} \exp(-\theta c_k^{rs})\delta_{ij,k}^{rs} \quad (12)$$

It is clear that we can compute probability of choosing link ij through WL_{ij} and SWN variables. And, path choice probability is represented to be equivalent to the logit-based model.

(e) Contribution to total link-flows

$$x_{ij}^{(l)} := x_{ij}^{(l)} + q^{rs}\frac{WL_{ij}}{SWN} \quad (13)$$

Step 1: Link travel time and link impedances update

(a) Set $l := l + 1$

(b) Set $t_{ij}^{(l)} \coloneqq t_{ij}(x_{ij}^{(l)})$ and $a_{ij} \coloneqq \exp(-\theta\, t_{ij}^{(l)})$

Step 2: Direction finding

Set all WL_{ij} and WN_n to 0. Set WN_r :=1 and $SWN = 0$. Based on link travel time $\{t_{ij}^{(l)}\}$ and a_{ij}, running DFS algorithm from origin node r to destination node s with recursive technique. After each link and node with nonzero value of Ω_{ij}^{rs} is queried we update $WN_j = a_{ij}WN_i$. When reaching destination node s, we add WN_s to WL_{ij} if link ij is queried from r to s and add WN_s to SWN. When the DFS algorithm stop, the auxiliary link flow is yielded as following

$$y_{ij}^{(l)} := y_{ij}^{(l)} + q^{rs}\frac{WL_{ij}}{SWN} \quad (14)$$

Step 3: Link flow update

(a) Choose a sequence of real numbers such that $(0 < \alpha^{(l)} < 1)$. The most frequently predetermined $\alpha^{(l)} = 1/(l+1)$ is used.

(b) Let $g_{ij}^{(l)} = y_{ij}^{(l)} - x_{ij}^{(l)}$

(c) Set $x_{ij}^{(l+1)} = x_{ij}^{(l)} + \alpha^{(l)} g_{ij}^{(l)}$

Step 4: Stopping test

If $max_{ij}\left\{\left|g_{ij}^{(l)}\right|\right\} \le \varepsilon$, stop. The solution is $\{x_{ij}^{(l)}\}$. Otherwise, go to step 2.

In the proposed new algorithm, instead of using well-known "forward" and "backward" techniques with requirement of sorting $\{C_{rn}^{0}\}$ in order of increasing access time from r, we only use DFS algorithm with link-based and node-based variables. While retaining the advantage of preventing the path enumeration that reduce computation storage, the new algorithm will also reduce the computation time.

4. SENSITIVITY ANALYSIS FOR STOCHASTIC USER EQUILIBRIUM TRAFFIC ASSIGNMENT WITH LINK-BASED APPROACH

We directly start from the equation of logit-based traffic assignment, we have

$$x_{ij} = \sum_{rs\in H}\left(q^{rs}\frac{\sum_{k\in K_{rs}}\exp(-\theta c_k^{rs})\delta_{ij,k}^{rs}}{\sum_{p\in K_{rs}}\exp\left(-\theta c_p^{rs}\right)}\right) \quad (15)$$

Let u_{ij} denotes the right-hand side of equation (15). Assuming u_{ij} is a function of arguments $q^{rs}(\gamma^{rs})$ and $c_k^{rs}(\, t_{ij}(x_{ij}, \varepsilon_{ij}))$.

When we have an result of SUE, the problem of sensitivity analysis is the calculation of the changes of link flow x_{ij} caused by small changes of γ^{rs} and ε_{ij}, *i.e.* computing the partial derivatives. Without loss of generality, since we compute the derivative of link flow with respect to one uncertainty parameter, another is assumed equal 0.

The first problem is the calculation of $\mathbf{\nabla_{\varepsilon} x}$ with $\boldsymbol{\gamma} \equiv 0$ and fixed travel demand $\{q^{rs}\}$.

By using Eq. (15), we can define the following function with vector form

$$\mathbf{d}(\mathbf{x}, \boldsymbol{\varepsilon}) = \mathbf{x} - \mathbf{u}(\mathbf{t}(\mathbf{x}, \boldsymbol{\varepsilon})) \quad (16)$$

This equation is assumed as functions with arguments **x** and perturbation **ε**. The gap at both sides of Eq. (16) should be zero; that is, **d(x, ε)** = **0**, under network equilibrium state.

From the general formula for derivative of implicit function. It is clear that

$$\nabla_{\varepsilon}\mathbf{x} = -\nabla_{\mathbf{x}}\mathbf{d}^{-1}\nabla_{\varepsilon}\mathbf{d} \tag{17}$$

In addition, by the chain rule of differentiation, we have $\nabla_{\mathbf{x}}\mathbf{u} = \nabla_{\mathbf{t}}\mathbf{u}\nabla_{\mathbf{x}}\mathbf{t}$ and $\nabla_{\varepsilon}\mathbf{u} = \nabla_{\mathbf{t}}\mathbf{u}\nabla_{\varepsilon}\mathbf{t}$

Therefore, we obtain the following equations with $\mathbf{I}$ as a unit matrix

$$\nabla_{\mathbf{x}}\mathbf{d} = \mathbf{I} - \nabla_{\mathbf{t}}\mathbf{u}\nabla_{\mathbf{x}}\mathbf{t} \tag{18}$$

And,

$$\nabla_{\varepsilon}\mathbf{d} = -\nabla_{\mathbf{t}}\mathbf{u}\nabla_{\varepsilon}\mathbf{t} \tag{19}$$

Eq. (17) become

$$\nabla_{\varepsilon}\mathbf{x} = -(\mathbf{I} - \nabla_{\mathbf{t}}\mathbf{u}\nabla_{\mathbf{x}}\mathbf{t})^{-1}\nabla_{\mathbf{t}}\mathbf{u}\nabla_{\varepsilon}\mathbf{t} \tag{20}$$

With the same approach, the second problem of calculating $\nabla_{\gamma}\mathbf{x}$ with $\boldsymbol{\varepsilon} \equiv 0$ is resolved by

$$\nabla_{\gamma}\mathbf{x} = -(\mathbf{I} - \nabla_{\mathbf{t}}\mathbf{u}\nabla_{\mathbf{x}}\mathbf{t})^{-1}\nabla_{\mathbf{q}}\mathbf{u}\nabla_{\gamma}\mathbf{q} \tag{21}$$

According Eq. (5), we have $\nabla_{\gamma}\mathbf{q} = \mathbf{I}$**,** thus

$$\nabla_{\gamma}\mathbf{x} = -(\mathbf{I} - \nabla_{\mathbf{t}}\mathbf{u}\nabla_{\mathbf{x}}\mathbf{t})^{-1}\nabla_{\mathbf{q}}\mathbf{u} \tag{22}$$

Since $\nabla_{\mathbf{x}}\mathbf{t}$ and $\nabla_{\varepsilon}\mathbf{t}$ are easily computed from implicit BPR function according Eq. (1), the difficulties in here are the calculations of $\nabla_{\mathbf{t}}\mathbf{u}$ and $\nabla_{\mathbf{q}}\mathbf{u}$. We have

$$\nabla_{\mathbf{t}}\mathbf{u} = \begin{pmatrix} \frac{\partial u_{ij}}{\partial t_{ij}} & \cdots & \frac{\partial u_{ij}}{\partial t_{|A|}} \\ \vdots & \ddots & \vdots \\ \frac{\partial u_{|A|}}{\partial t_{ij}} & \cdots & \frac{\partial u_{|A|}}{\partial t_{|A|}} \end{pmatrix} \tag{23}$$

And,

$$\nabla_{\mathbf{q}}\mathbf{u} = \begin{pmatrix} \frac{\partial u_{ij}}{\partial q^{rs}} & \cdots & \frac{\partial u_{ij}}{\partial q^{|H|}} \\ \vdots & \ddots & \vdots \\ \frac{\partial u_{|A|}}{\partial q^{rs}} & \cdots & \frac{\partial u_{|A|}}{\partial q^{|H|}} \end{pmatrix} \tag{24}$$

From Eq. (15), it is clear that

$$\frac{\partial u_{ij}}{\partial t_{gh}} = \sum_{rs\in H}\left(q^{rs}\left[\frac{-\theta\sum_{k\epsilon K_{rs}}\exp(-\theta c_k^{rs})\,\delta_{ij,k}^{rs}\delta_{gh,k}^{rs}}{\sum_{p\epsilon K_{rs}}\exp(-\theta c_p^{rs})} - \frac{-\theta\left(\sum_{m\epsilon K_{rs}}\exp(-\theta c_m^{rs})\delta_{ij,m}^{rs}\right)\left(\sum_{l\epsilon K_{rs}}\exp(-\theta c_l^{rs})\delta_{gh,l}^{rs}\right)}{\left(\sum_{p\epsilon K_{rs}}\exp(-\theta c_{h,p})\right)^2}\right]\right)$$
$$= \sum_{rs\in H}\left(q^{rs}\left[-\theta\frac{x_{ij,gh}^{rs}}{q^{rs}} + \theta\frac{x_{ij}^{rs}}{q^{rs}}\frac{x_{gh}^{rs}}{q^{rs}}\right]\right) = \sum_{rs\in H}\left(\theta\left[-x_{ij,gh}^{rs} + \frac{x_{ij}^{rs}x_{gh}^{rs}}{q^{rs}}\right]\right) \tag{25}$$

And,

$$\frac{\partial u_{ij}}{\partial q^{rs}} = \sum_{rs\in H}\left(\frac{\sum_{k\epsilon K_{rs}}\exp(-\theta c_k^{rs})\delta_{ij,k}^{rs}}{\sum_{p\epsilon K_{rs}}\exp(-\theta c_p^{rs})}\right) = \sum_{rs\in H}\left(\frac{x_{ij}^{rs}}{q^{rs}}\right) \tag{26}$$

In what follows, let us expand and explain how they can be efficiently computed by DFS method. Eqs. (25) and (26) are repeat computed for each OD pair, so we only need variables including link-based variables and node-based variables that reduce storage cost. At SUE state, when the travel time of all links are fixed, x_{ij}^{rs} can be calculated by using DFS algorithm once because in DFS algorithm, x_{ij} is computed for each OD pair and $x_{ij} = \sum_{rs\in H} x_{ij}^{rs}$. After x_{ij}^{rs} is easily computed with link-based approach, we need to compute $x_{ij,gh}$. In real large networks, if the number of OD pairs is too large, storage price will be very large for saving both x_{ij-gh}^{rs} and $x_{ij,gh}^{rs}$. Furthermore, the fact that either x_{ij-gh}^{rs} or x_{gh-ij}^{rs} is zero implies that $x_{ij,gh}^{rs} = \max\{x_{ij-gh}^{rs}, x_{gh-ij}^{rs}\}$ (J.Q.Ying and T.Miyagi[23]). And, Ying and Miyagi[23] computed x_{ij-gh}^{rs} through x_{ij}^{rs} and p_{gh}^{js}. In return, it need to run STOCH algorithm for each destination s and store p_{gh}^{js}. The former may cause computing time problem and the latter required large storage cost. In addition, if we used STOCH algorithm to computed p_{gh}^{js} that is used to compute x_{ij-gh}^{rs}, it will be no problem with simple traffic networks however many problems will occurred in complex road networks, including inaccuracies.

For example, the following small virtual network consists of 6 nodes, 8 links and two OD pair from node 1 to node 6 (OD1) and node 2 to node 6 (OD2) will show that we cannot use this technique. The link capacity and free-flow travel time are written in the form of [free-flow travel time (min), capacity (pcu)]. The parameters of the BPR-type performance function are $\alpha = 1.0$ and $\beta = 2.0$. The parameter in the logit model, θ, is 1.0. The OD demands are $q^{16} = 30$ and $q^{26} = 30$. After using STOCH algorithm with MSA method, the results of link travel time and link travel flow are shown as table 1.

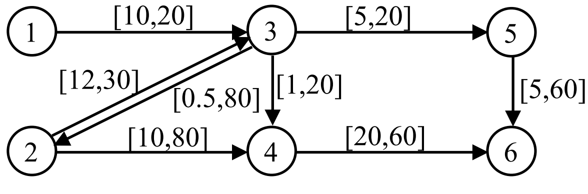

Fig. 1 A small network with 6 nodes and 8 links

Table 1 The results of SUE state of small network

No.	Link	Link travel time	Link travel flow		
			OD1	OD2	Total
1.	13	32.500	30.00	0	30.00
2.	23	12.608	0	6.75	6.75
3.	24	10.845	0.00	23.25	23.25
4.	32	0.500	0.00	0	0.00
5.	34	1.064	5.08	0	5.08
6.	35	17.539	24.92	6.75	31.67
7.	46	24.458	5.08	23.25	28.33
8.	56	6.393	24.92	6.75	31.67

At SUE state, if we compute p_{gh}^{js} for all links by running STOCH algorithm for assigning a virtual unit OD demand on link gh and compute $x_{ij-gh}^{rs} = x_{ij}^{rs} p_{gh}^{js}$, the results would be inaccurate. In constant, $p_{34}^{36} = 0.169$, $x_{23-34}^{26} = x_{23}^{26} p_{34}^{36} = 1.14$. However, this result contradicts the nature of the STOCH algorithm ($x_{23-34}^{26} = 0$). In fact, there is no traveller from 2 to 6 use link 34 (because this link is not included on the "efficient" path, see Table 1). Another unreasonable result is $x_{23-32}^{26} \neq 0, x_{32-23}^{16} \neq 0$ although small. The above inconsistencies can make the use of p_{gh}^{js} variable produce significant deviations. When this variable cannot be used, despite ignoring large storage space, we are forced to run the STOCH algorithm repeatedly with each OD pair and each non-zero x_{ij}^{rs} to get the result of $\{x_{ij-gh}^{rs}\}$ at the price of a large computational time. Moreover, the application of a mechanical STOCH algorithm to the actual complex traffic network without paying attention to the tree structure of the data at SUE state can cause many unforeseen problems.

Consequently, we will proposed DFS method instead of STOCH algorithm to compute x_{ij-gh}^{rs} solving the problem of storage space, accuracy as well as calculation time. We consider that $\delta_{ij-gh,k}^{rs}$ denotes the double link-path incidence variable. If the path k includes both links ij and gh in such a way that link ij is used prior to link gh then $\delta_{ij-gh,k}^{rs} = 1$, otherwise $\delta_{ij-gh,k}^{rs} = 0$. From the definition of x_{ij-gh}^{rs}, we have

$$x_{ij-gh}^{rs} = q^{rs} \frac{\sum_{k \in K_{rs}} \exp(-\theta c_k^{rs}) \delta_{ij-gh,k}^{rs}}{\sum_{p \in K_{rs}} \exp\left(-\theta c_p^{rs}\right)} \quad (27)$$

As highlighted above, by running DFS algorithm from origin node *r* to destination node *s* with recursive technique, each time we reach the destination *s*, we will have the information of all the links that are queried, *i.e.* all links belong to a path *k*. From this information, we can compute numerical part of Eq.27. With recursive technique, we can reach destination node *s* by $|K_{rs}|$ times, and Eq.27 is totally computed. This calculation process is combined with x_{ij}^{rs} variable calculation to save computing time.

After calculation x_{ij-gh} , we easily calculate $x_{ij,gh}$. We will calculate with each OD pair, and we can reuse variables to save memory. For simplicity, we can ignore the symbol ^{rs}. And, we only need two $link\ x\ link$ matrices to store x_{ij-gh} and $x_{ij,gh}$.

The calculation process of sensitivity analysis for stochastic user equilibrium traffic assignment with DFS algorithm is as follows.

Step 1: At SUE state, based on the free flow travel time $\{t_{ij}^0\}$, for each OD pair rs, calculate the minimum travel time to all other nodes. For each link ij, set $a_{ij} \coloneqq \exp(-\theta\, t_{ij})$

Step 2: This step is run for each OD pair

(a) Set $\Omega_{ij}^{rs} \coloneqq 1$ if $C_{is}^0 \geq C_{js}^0$ and $(1 + h_{ij}^r)(C_{rj}^0 - C_{ri}^0) \geq t_{ij}^0$, other wise $\Omega_{ij}^{rs} \coloneqq 0$

(b) Set all WL_{ij} and WN_n to 0. Set WN_r:=1 and $SWN = 0$. By running DFS algorithm from origin node *r* to destination node *s* with recursive technique. After each link and node with nonzero value of Ω_{ij}^{rs} is queried, we update $WN_j = a_{ij} WN_i$. When reaching destination node *s*, we add WN_s to WL_{ij} if link *ij* is queried from *r* to *s* and add WN_s to SWN. At the same time, with the information of the links that belong to a path *k*, we compute numerical part of Eq.27 .With recursive technique, we can reach destination node s by $|K_{rs}|$ times and x_{ij}^{rs}, x_{ij-gh}^{rs}, x_{ijgh}^{rs} variables are calculated.

(c) Compute Eqs. (23) and (24).

Step 3: Calculate $\nabla_{\boldsymbol{\varepsilon}}\mathbf{x}$, $\nabla_{\boldsymbol{\gamma}}\mathbf{x}$ according Eqs. (20) and (22).

After the derivatives of link traffic flow with respect to perturbations $\boldsymbol{\varepsilon}$ and $\boldsymbol{\gamma}$ are calculated, we can compute the approximate link flow with the changes of $\boldsymbol{\varepsilon}$ and $\boldsymbol{\gamma}$, according to first order Taylor expansion.

5. COMPUTATIONAL EVIDENCE

In this section, a simple application and Kanazawa city urban application are adopted to demonstrate the accuracy and effectiveness of the proposed method.

(1) Simple example

To assess the accuracy of the proposed method (SADFS), performance indicators are defined. In these indicators, $\tilde{\mathbf{x}}$ denotes the estimated perturbation link flows, and $\mathbf{x}$ denotes the actual perturbation link

Table 2 The comparison between two methods

Link	Unperturbed link flows	Perturbed link flows with ε = 1					
		Actual	Actual change	SASTOCH method		SADFS method	
				Estimated	Estimation Error	Estimated	Estimation Error
13	30.00	30.00	0.00	30.00	0.00	30.00	0.00
23	6.75	4.67	-2.08	4.35	0.32	4.50	-0.17
24	23.25	25.33	2.08	25.72	-0.40	25.50	0.17
32	0.00	0.00	0.00	0.00	0.00	0.00	0.00
34	5.08	4.87	-0.21	4.65	0.22	4.95	0.08
35	31.67	29.80	-1.87	29.63	0.17	29.55	-0.25
46	28.33	30.20	1.87	30.37	-0.17	30.45	0.25
56	31.67	29.80	-1.87	29.63	0.17	29.55	-0.25
RMSE				**0.223**		**0.175**	
%RMS				**1.16%**		**0.91%**	

flows. The $RMSE$ and $\%RMS$ indicate the error of comparing between estimated flows and actual flows:

$$RMSE = \sqrt{\frac{1}{|A|}\sum_{ij=1}^{|A|}\left(\tilde{x}_{ij} - x_{ij}\right)^2} \qquad (28)$$

$$\mu = \frac{1}{|A|}\sum_{ij=1}^{|A|} x_{ij} \qquad (29)$$

$$\%RMS = \frac{RMSE}{\mu}100(\%) \qquad (30)$$

We will compare the results of two sensitivity analysis methods. The difference lies in how $x_{ij,gh}^{rs}$ is calculated. The first is the proposed methods with DFS algorithm (SADFS); The second is the method by running STOCH3 algorithm repeatedly with non-zero x_{ij}^{rs} (SASTOCH). The results of comparison are expressed through the table 2.

We can see that the proposed method has a higher accuracy with a smaller error than the old method. The $RMSE$ and $\%RMS$ indicators of proposed model are all lower than the old method. Although the difference is quite small because the applied model is simple with only 2 OD pairs, 7 nodes and 8 links. The errors caused by using STOCH algorithm will increase when applied to the actual network with the complexity of the network and large OD pairs, nodes and links.

(2) Application to Kanazawa road network

Consequently, the performance of the proposed method will be shown in the application to the Kanazawa road network. Figure.2 Shows the Kanazawa road network in Ishikawa prefecture, Japan, including 272 nodes and 964 links. The OD demand data with 383 OD pairs of the morning peak from 6:00 to 7:00 AM that is derived from a previously personal trip survey is used. The BPR function of the travel time of the car uses $\alpha = 1.0, \beta = 2.0$ and the assumed parameter is $\theta = 1$. Elongation ratio h_{ij}^{r} is set to 1.5 for every links of network. The programmes were coded by fortran.90 programming language, and they are run on a personal computer with Intel Core 2 Quad CPU of 2.24Ghz, 4GB RAM, and Windows 10 Education.

Figure 3 shows the comparison between new algorithm and STOCH3 algorithm. The algorithms are coded for each OD pair. After 11547 iterations, the same results of traffic flow are output and the calculation time of new algorithm is significantly reduced when compared to the STOCH3 algorithm. In this case, the new algorithm has reduced the calculation time by 79%.

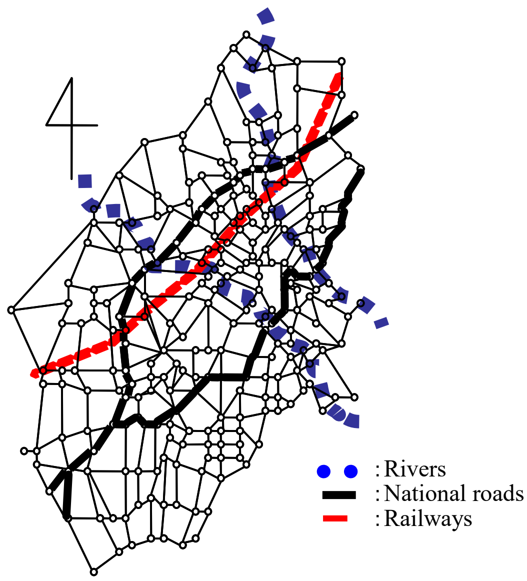

Fig. 2 Kanazawa road network

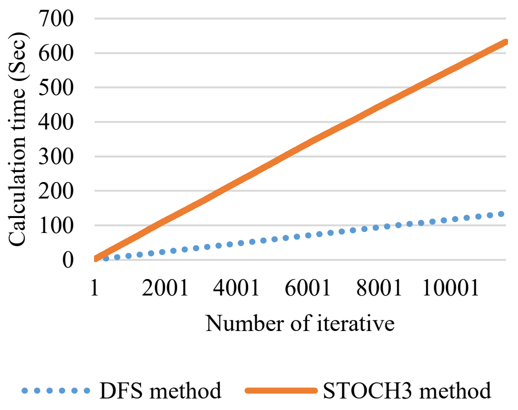

Fig. 3 Calculation time of SUE traffic assignment

Table 3 The calculation time of two methods

	SADFS	SASTOCH
Total calculation time (Sec)	96.7	152.22
Calculation time of Eqs. 23 and 24 (Sec)	8.64	66.31

Table 3 shows the calculation time of two sensitivity analysis methods. The computational time of the proposed method is only about half the SASTOCH method. If not mention the inverse matrix calculation time, the difference is greater when only mentioning the time for calculating Eqs. (23) and (24). The proposed method will save 91% of the time calculated Eqs. (23) and (24). This demonstrate the time-efficient calculation of the proposed method. In addition, the computation time for reaching convergence of STOCH3 algorithm with MSA is about 11 min. Therefore, using the proposed sensitivity analysis method will reduce the computational time by 84.8%. Especially, when this effect will increase as the number of iterations of the SUE model increases. However, this method needs to ensure accuracy, we will consider it through the table 4. The difference in accuracy is indicated at table 4.

Table 4 Accuracy comparison between two methods

Parameters	Methods	Indicators	
		RMSE	*%RMS*
$\boldsymbol{\varepsilon} = 0.1$	SADFS	0.42	0.38%
	SASTOCH	5.11	4.58%
$\boldsymbol{\gamma} = 10$	SADFS	1.70	0.11%
	SASTOCH	5.59	3.61%
$\boldsymbol{\varepsilon} = 0.1$ and $\boldsymbol{\gamma} = 10$	SADFS	2.58	1.68%
	SASTOCH	9.79	6.37%

While the error of comparing between estimated flows and actual flows of SASTOCH method are quite large, it of proposed method shows high accuracy. The percentage error (*%RMS*) of proposed method does not exceed 0.5% in both results when changing $\boldsymbol{\varepsilon}$ and $\boldsymbol{\gamma}$**.** When changing both parameters at the same time, the estimated results are also quite accurate. The results suggest that the proposed method is accurate method to estimate SUE STOCH3 model, robust to different parameter settings and highly applicable to complex networks.

6. CONCLUSIONS

This study has presented an effective sensitivity analysis for the logit-based stochastic traffic assignment problems. DFS algorithm have been used to achive SUE model and provide the information of derivative of link flows with respect to some parameters. Such derivative information could be utilized for the transport design and management policies. Since an equilibrium solution has been calculated, the proposed method allows estimate acceptable approximate solutions for any combination of changes in the parameters. Especially, the proposed method can keep the computational accuracy and reduce the computational cost. The results of the numerical and real traffic networks were presented the efficiency of the proposed solution method. In the future, it would be interesting to extend the results in this paper to other methods of stochastic assignment except MSA method.

REFERENCES

1) OKAMOTO, Y., Nakayama, S. & Takayama, J. Network path aggregation with sensitivity analysis for expressway pricing. *The Transportation Research Board (TRB) 90th Annual Meeting* (2011).
2) OKAMOTO, Y., Nakayama, S. & Takayama, J. A Sensitivity Analysis Method of Aggregating path Flows under Stochastic User Equilibrium. *The 16th international conference of Hong Kong society for transportation studies* (2011).
3) OKAMOTO, Y., NAKAYAMA, S. & TAKAYAMA, J. Efficient calculation of stochastic user equilibrium with path aggregation using sensitivity analysis. *Infrastructure Planning and Management,* 28, 481–489 (2011).
4) Wardrop, J. G. Some theoretical aspects of road traffic research. *Proceedings of the Institution of Civil Engineers*, 1, 325–362 (1952).
5) Sheffi, Y. Urban transportation networks: Equilibrium Analysis with Mathematical Programming Methods. *PRENTICE-HALL, INC., Englewood Cliffs, New Jersey* (1985).
6) Tobin, R. L. & Friesz, T. L. Sensitivity Analysis for Equilibrium Flow. *Transportation Science,* 22, 242–250 (1988).
7) Kyparisis, J. Perturbed solutions of variational inequality

problems over polyhedral sets. *Journal of Optimization Theory and Applications,* 57, 295–305 (1988).
8) Kyparisis, J. Sensitivity analysis for variaional inequalities and nonlinear complementarity problems. *Annals of Operations Research,* 5, 143–173 (1990).
9) Qiu, Y. & Magnanti, T. L. Sensitivity analysis for variational inequalities defined on polyhedral sets. *Mathematics of Operations research*, 14, 410–432 (1989).
10) Yang, H. Sensitivity analysis for queuing equilibrium network flow and its application to traffic control. *Mathematical and Computer Modelling*, 22, 247–258 (1995).
11) Kim, T. J. & Suh, S. Advanced Transport and Spatial Systems Models: Applications to Korea. (*Springer-Verlag, NewYork,* 1990).
12) Yang, H., Yagar, S., Iida, Y. & Asakura, Y. An algorithm for the inflow control problem on urban freeway networks with user-optimal flows. *Transportation Research Part B*, 28B, 123–139 (1994).
13) Yang, H. & Yagar, S. Traffic assignment and traffic control in general freeway-arterial corridor systems. *Transportation Research Part B,* 28B, 463–486 (1994).
14) Yang, H. & Lam, W. H. K. Optimal Road Tolls Under Conditions and Congestion. *Transportation Research Part A,* 30, 319–332 (1996).
15) Yang, H. Sensitivity analysis for the elastic-demand network equilibrium problem with applications. *Transportation Research Part B,* 31, 55–70 (1997).
16) Yang, H. & Bell, M. G. . Traffic restraint, road pricing and network equilibrium. *Transportation Research Part B,* 31, 303–314 (1997).
17) Daganzo, C. F. & Sheffi, Y. On Stochastic Models of Traffic Assignment. *Transportation Science,* 11, 253–274 (1977).
18) Dial, R. B. A probabilistic multipath traffic assignment model which obviates path enumeration. *Transportation Research,* 5, 83–111 (1971).
19) Bell, M. G. H. Alternatives to Dial logit assignment algorithm. *Transportation Research Part B,* 29B, 287–2 (1995).
20) Burrell, J. E. Multiple path assignment and its application to capacity restraint. *Proceedings of Fourth International Symposium on the Theory of Traffic Flow*, (1968).
21) Fisk, C. Some developments in equilibrium traffic assignment. *Transportation Research Part B,* 14B, 243–255 (1980).
22) Clark, S. & Watling, D. Probit-Based Sensitivity Analysis for General Traffic Networks. *Transportation Research Record,* 1733, 88–95 (2000).
23) Ying, J. Q. & Miyagi, T. Sensitivity Analysis for Stochastic User Equilibrium Network Flows — A Dual Approach. *Transportation Science*, 35, 124–133 (2001).
24) Ji, K., Ma, J. & Tang, W. Sensitivity analysis for stochastic user equilibrium traffic assignment with constraints. *Advances in Mechanical Engineering*, 9, 1–7 (2017).
25) Leurent, F. M. Curbing the Computational Difficulty of the Logit Equilibrium Assignment Model. *Transportation Research Part B,* 31, 315–326 (1997).
26) Daganzo, C. F. Unconstrained Extremal Formulation of Some Transportation Equilibrium Problems. *Transportation Science,* 16, 332–360 (1982).

(Received ?)