https://dx.doi.org/10.1016/j.ejor.2020.08.058

A Faster Path-based Algorithm with Barzilai-Borwein Step Size for Solving Stochastic Traffic Equilibrium Models

Muqing Du [a], Heqing Tan [a], Anthony Chen [b,*]

[a] *College of Civil and Transportation Engineering, Hohai University, 1 Xikang Rd, Nanjing, Jiangsu, China*

[b] *Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China*

[*] Corresponding author.

E-mail address: anthony.chen@polyu.edu.hk

ABSTRACT

Step size determination (also known as line search) is an important component in effective algorithmic development for solving the traffic assignment problem. In this paper, we explore a novel step size determination scheme, the Barzilai-Borwein (BB) step size, and adapt it for solving the stochastic user equilibrium (SUE) problem. The BB step size is a special step size determination scheme incorporated into the gradient method to enhance its computational efficiency. It is motivated by the Newton-type methods, but it does not need to explicitly compute the second-order derivative. We apply the BB step size in a path-based traffic assignment algorithm to solve two well-known SUE models: the multinomial logit (MNL) and cross-nested logit (CNL) SUE models. Numerical experiments are conducted on two real transportation networks to demonstrate the computational efficiency and robustness of the BB step size. The results show that the BB step size outperforms the current step size strategies, i.e., the Armijo rule and the self-regulated averaging scheme.

Keywords: transportation; stochastic user equilibrium; Barzilai-Borwein step size; path-based traffic assignment algorithm; cross-nested logit

1. Introduction

Traffic assignment (TA) problem plays a central role in urban transportation network analysis (Sheffi, 1985). It has been widely used for various purposes, for example, transportation system assessment (Kitthamkesorn and Chen, 2017), impact prediction of projects and policies (Huang and Li, 2007; Cantarella et al., 2015), traffic restraints with respect to congestion or emission (Chen et al., 2011), time-cost tradeoff in traffic assignment (Leurent, 1993; Chen et al., 2010, 2012; Perederieieva et al., 2018), traffic management strategy (Yang and Bell, 1997; Tan et al., 2019), signal control optimization (Chiou, 2007), road network design (Yin et al., 2009; Szeto et al., 2015), paradox analysis (Yang and Bell, 1998; Yang and Chen, 2009; Yao and Chen, 2014; Yao et al., 2019), etc. The stochastic user equilibrium (SUE) problem is one of the important models of the TA problem. Substantial efforts have also been made on the design of efficient algorithms for solving the SUE problems (Chen and Alfa, 1991; Damberg et al., 1996; Maher, 1998; Bekhor & Toledo, 2005; Liu et al., 2009; Xu et al., 2012; Zhou et al., 2012; Chen et al., 2013; Yu et al., 2014; Zhou et al., 2014; Xu et al., 2019). However, the existing solution algorithms often need to either frequently evaluate the objective function (and/or its derivative) or use inflexible step size determination rules (e.g., monotonically decreasing the step size sequence), which impede the efficiency on both speed and precision of the algorithmic convergence. Recently, a novel step size determination scheme, called the BB step size (Barzilai and Borwein, 1988), has been reported to show great potential for solving the travel demand forecasting models (Gibb, 2016). The BB step size originates from the Newton-type method (second-order approach), but it involves nearly no extra cost over the standard gradient method (first-order approach) for solving various optimization problems. In this paper, we extend the BB step size scheme to solve the SUE problem. This scheme tries to incorporate the step size with the derivative information by only utilizing the mapping function value of the last two consecutive iterations. It does not need to compute the derivatives of the mapping function or extra evaluations of the mapping function. Our numerical results show the BB step size significantly outperforms the existing step size determination schemes.

TA aims to allocate the origin-destination (O-D) travel demand to the transportation network based on the given path choice criterion (Sheffi, 1985). The most widely used criterion is the user equilibrium (UE) principle (Wardrop, 1952). However, the UE principle is recognized to be unrealistic, because it assumes that all travelers have accurate perceptions on the condition of the transportation network (Prashker and Bekhor, 2004). Daganzo and Sheffi (1977) suggested the stochastic user equilibrium (SUE) principle to capture travelers' perception errors on travel time. Further, in order to reflect the perception errors, the SUE principle incorporates an additional random error term into the travel cost. In practical applications, the random error terms are usually assumed to follow the Gumbel distribution (Dial, 1971), the Normal distribution (Daganzo and Sheffi, 1977), or the Weibull distribution (Castillo, 2008), which correspond to the logit-based, probit-based and weibit-based SUE models, respectively. Among these models, the logit-based SUE model has drawn the most attention since it has a closed-form expression and is computationally tractable. However, the logit-based SUE model suffers from the independence assumption which causes flows on the overlapping paths to be overestimated. In order to overcome the drawbacks of the logit-based SUE model, various extended logit-based SUE models have been proposed, e.g., C-logit (Cascetta et al., 1996), cross-nested logit (Vovsha and Bekhor, 1998), paired combinatorial logit

(Prashker and Bekhor, 1999; Pravinvongvuth and Chen, 2005), and path-size logit models (Ben-Akiva et al., 1999; Chen et al., 2012). For more comprehensive reviews of these extended models, readers can refer to Prashker and Bekhor (2004) and Kitthamkesorn and Chen (2013). Nevertheless, solving the logit-based SUE model or its extensions is still challenging. The difficulty mainly comes from the large dimension of solution variables and the complicated objective function under congested and realistic networks (Chen et al., 2014).

In optimization, line search and trust region are two basic iterative approaches to find a local minimum solution. The former first generates a search direction and then focuses on determining a suitable step size along the search direction; the latter defines a region around the current approximate solution within which a quadratic function is used to approximate the original objective function. Zhou et al. (2014) recently explored the trust region method to solve the MNL SUE model. For the trust-region method, the fast convergence can only be obtained when the approximate solution is close enough to the optimal solution. If this assumption does not hold, the trust-region radius has to be limited to a small range to satisfy the strictly positive constraints, which eventually degrades its overall performance. Besides, the trust-region method requires to compute second-order derivatives, which is computationally expensive on realistic networks, especially for the SUE problem embedded with advanced choice models (e.g., the extended logit-based models). Consequently, this study will explore the line search approach (or step size determination) and contribute to the computational efficiency for solving the SUE models.

In the literature, step size determination strategies adopted to solve the SUE problem can be divided into three types: predetermined, inexact and exact (Chen et al., 2014). The predetermined step size sequence does not require to evaluate the objective function value or the derivative information and thus is easy to implement. The most widely used predetermined scheme is the method of successive averages (MSA). The MSA scheme generates a sequence $\{\alpha_n\}$ that satisfies Blum's theorem (Blum, 1954) to guarantee the convergence (i.e. $\alpha_n \rightarrow 0$, $\sum_{n=0}^{\infty} \alpha_n = \infty$). However, the MSA suffers from a sublinear convergence rate. Following the MSA, Liu et al. (2009) developed the self-regulated averaging (SRA) scheme that improves the convergence speed of the MSA scheme whilst retains its simplicity. On the other hand, the exact and inexact step size determination schemes are proposed to overcome the sublinear convergence rate of the MSA scheme. By contrast, with fewer evaluation times of the objective function value and/or the derivative, the inexact one is more efficient and preferred in practice. Among others, the Armijo scheme is perhaps the most widely used inexact step size determination scheme for solving the SUE problem with an extended logit path choice model (Bekhor et al., 2007; Bekhor et al., 2008).

Although the SRA and Armijo schemes usually perform satisfactorily, they still retain some unattractive properties. The step size sequence of the SRA scheme is monotonically decreasing with iterations. If the step size has decreased sharply in the early iterations, the convergence will be rather slow. On the other hand, the Armijo scheme requires to evaluate the objective function and/or its gradient, and the computational cost is not cheap, especially for the models with a large dimension of solution variables and complicated objective functions (e.g., cross-nested logit SUE model).

This paper aims to develop a more efficient step size determination scheme for solving the SUE problem. Specifically, we will adopt the Barzilai-Borwein (BB) step size to develop an efficient path-based traffic assignment algorithm for solving two logit-based SUE models (i.e., the multinomial logit (MNL) and the cross-nested logit (CNL) SUE models). We discover that the BB step size scheme has following attractive properties. First, the BB scheme determines the step size by using the information from the last two consecutive iterations and does not require any function evaluation and/or its gradient. This property is rather attractive for solving the SUE problem on realistic networks, especially the SUE models embedded with an advanced discrete choice model. Second, the convergence speed is faster. Our numerical results show that fewer iterations are required compared with the Armijo and SRA schemes. This outstanding performance comes from the merit of the BB step size which enables to utilize the second-order derivative information (i.e., the derivative of the mapping function) for accelerating the convergence. Third, no parameter is assumed for the BB step size determination, which makes it rather practical for large-scale transportation networks.

The remainder of this paper is organized as follows. Section 2 gives a description of the BB step size. After that, we briefly introduce two well-known stochastic traffic equilibrium models (i.e. the MNL SUE and the CNL SUE models) and review some well-known existing step size schemes. Then, we develop a path-based traffic assignment algorithm incorporated with the BB step size for solving the SUE problem, and gives the convergence results of the proposed algorithm under some restrictive assumptions. Next to that, numerical experiments are conducted to examine the convergence characteristics and sensitivity analysis of the BB scheme, and compare the flow allocation results between the MNL SUE model and the CNL SUE model. In addition, we introduce some extended applications of the proposed algorithm. At last, some concluding remarks are provided.

2. Review of the BB Step Size Determination Scheme

The BB step size was proposed by Barzilai and Borwein (1988) for the gradient-based descent method for solving the unconstrained optimization problem. The BB step size can be derived based on the quasi-Newton method which is regarded to be efficient in solving optimization problems. Before the discussion of the BB step size, the quasi-Newton method is briefly discussed.

Consider the following unconstrained nonlinear programming:

$$\min f(x) \tag{1}$$

where $f(x)$ is assumed to be a continuously differentiable convex function. Let x_n be an approximate solution at iteration n, and $g(x_n) = \nabla f(x_n)$ denotes the gradient of $f(x)$ at x_n. The steepest descent method defines the next approximate solution as

$$x_{n+1} = x_n - \bar{\alpha}_n g(x_n) \tag{2}$$

where the step size is determined as follows: $\bar{\alpha}_n = \underset{\alpha_n>0}{\operatorname{argmin}} f(x_n - \alpha_n g(x_n))$. The steepest descent method is quite simple and does not require to compute the second-order derivative. It has been modified to solve the constrained optimization problems, e.g., Rosen's gradient projection method (Rosen, 1960), and further employed to solve the user equilibrium (UE) traffic assignment problem, e.g., the projected gradient algorithm (Florian et al., 2009).

However, it is well known that the steepest descent method suffers from the zigzag phenomenon (Yuan, 2008) and performs poorly in ill-conditioned problems (i.e., the ratio between the largest eigenvalue and the smallest eigenvalue of the Hessian matrix is very large). As an alternative to the steepest descent method, Newton's method has often been adopted.

Solving problem (1) can be transformed into solving its stationary point. The first-order necessary condition of optimality is

$$\mathrm{g}(x) = 0. \tag{3}$$

Note that Eq. (3) is a system of nonlinear equations. The essence of Newton's method is to iteratively linearize $\mathrm{g}(x)$ and solve the linearized problem. If $\mathrm{g}(x)$ is differentiable (i.e., $f(x)$ is second-order differentiable), it can be approximated as follows

$$\mathrm{g}(x) \approx \mathrm{g}(x_n) + \mathrm{H}(x_n)(x - x_n) = 0 \tag{4}$$

where $\mathrm{H}(x_n) = \nabla^2 f(x)$ is the Hessian matrix at x_n. Newton's method defines the next approximate solution by solving Eq. (4). Hence, the next approximate solution is defined as

$$x_{n+1} = x_n - [\mathrm{H}(x_n)]^{-1}\mathrm{g}(x_n). \tag{5}$$

Newton's method has a local second-order convergence rate, but it requires to compute the inverse of Hessian matrix in Eq. (5). Hence, it may be only suitable for solving small-scale problems, e.g., trajectory control of autonomous vehicles (Shivam et al., 2019), and computationally unattractive for solving large-scale traffic assignment problems.

If the second-order derivative information is unavailable or the computational cost of the Hessian inverse is expensive, the quasi-Newton method will often be adopted as an alternative. The idea of the quasi-Newton method is to replace the exact inverse of the Hessian matrix with an approximate matrix A_n as follows:

$$x_{n+1} = x_n - \mathrm{A}_n\mathrm{g}(x_n). \tag{6}$$

Different ways to construct A_n yield different schemes of the quasi-Newton method. One approach is to approximate the Hessian matrix with its diagonal. This approach has been widely adopted for solving various traffic assignment problems, for example, the gradient projection algorithm (Jayakrishnan et al., 1994), the origin-based algorithm (Bar-Gera, 2002), and the greedy path-based algorithm (Xie et al., 2018) for the UE traffic assignment problem, the gradient projection algorithm (Bekhor and Toledo, 2005), the trust region method (Zhou et al., 2014), and the truncated Newton method (Xu et al., 2019) for the logit-based SUE traffic assignment problem, and the modified gradient projection algorithm (Ryu et al., 2014) for the elastic demand traffic assignment problem. However, the second-order derivative is still required, which limits its applicability. A more robust way is to construct the approximate matrix with only the first-order derivative. This approach has also been adopted to solve various bi-level network design problems, e.g., road network design problem (Huang et al., 2001) and area traffic control problem (Chiou, 2007). In this study, we refer to this approach as the *secant equation scheme*, in order to avoid confusion.

Note that the Hessian matrix $\mathrm{H}(x_n)$ represents the local change rate of the gradient at x_n. Instead of computing $\mathrm{H}(x_n)$, it is reasonable to use the average change rate of the gradient to approximate it. If the inverse operation is also avoided, the approximate matrix A_n should satisfy

$$x_n - x_{n-1} = \mathrm{A}_n(\mathrm{g}(x_n) - \mathrm{g}(x_{n-1})). \tag{7}$$

Eq. (7) is called the *secant equation* and is underdetermined. The essence of the secant equation scheme is to use the average change rate of the gradient to approximate the second-order derivative.

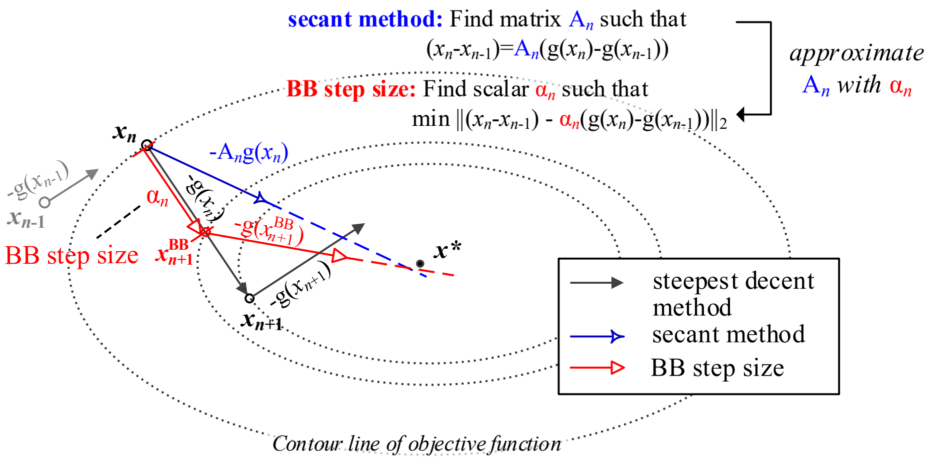


FIGURE 1 Illustration to the principle of the BB step size scheme

As indicated in Eq. (6), the secant equation scheme uses matrix A_n to modify the anti-gradient direction so as to produce a preferred search direction as shown in Figure 1. For the secant equation scheme, a superlinear convergence rate can be attained if the sequence of matrix A_n is designed appropriately (Nocedal and Stephen, 1999). Nevertheless, the matrix A_n needs to be stored and updated at each iteration so as to progressively approximate the inverse Hessian of $f(x)$, which is computationally burdensome especially for large-scale problems. These advantages and disadvantages motivated researchers to simplify the secant equation scheme by approximating the matrix A_n with a scalar α_n, which gives rise to the idea of the BB step size scheme. As shown in Figure 1, the BB step size scheme utilizes a special step size along the gradient direction to approximate the second-order derivative information while satisfying the secant equation. Although the BB scheme seems not to take the optimal movement along the anti-gradient direction at iteration n, the descent direction obtained at the next solution x_{n+1}^{BB} is actually better than that at point x_{n+1}. The BB step size can be regarded as a simplified version of the secant equation scheme.

Barzilai and Borwein (1988) first suggested substituting the matrix A_n with a scalar α_n (can also be regarded as $\alpha_n\mathrm{I}$, where I is the unit matrix)

$$x_{n+1} = x_n - \alpha_n \mathrm{g}(x_n). \tag{8}$$

In general, the *secant equation* in Eq. (7) does not hold if the matrix A_n is replaced with a scalar α_n, since a scalar cannot complete the linear transformation from one vector to another. In order to force the scalar to have a certain quasi-Newton property, it is reasonable to minimize the absolute difference between the left- and right-hand sides as follows:

$$\min \ \|(x_n - x_{n-1}) - \alpha_n(\mathrm{g}(x_n) - \mathrm{g}(x_{n-1}))\|_2. \tag{9}$$

Consequently, we have

$$\alpha_{BB1}=\underset{\alpha_n}{\mathrm{argmin}}\{\|(x_n\text{-}x_{n\text{-}1})\text{-}\alpha_n(\mathrm{g}(x_n)\text{-}\ \mathrm{g}(x_{n\text{-}1}))\|_2\}=\frac{(x_n\text{-}x_{n\text{-}1})^{\mathrm{T}}(\mathrm{g}(x_n)\text{-}\ \mathrm{g}(x_{n\text{-}1}))}{\|\mathrm{g}(x_n)\text{-}\ \mathrm{g}(x_{n\text{-}1})\|_2} \tag{10}$$

where 'T' is the transpose operation. The step size calculated by Eq. (10) is called BB1. Similarly, if the scalar α_n is placed the term $(x_n\text{-}x_{n\text{-}1})$ of Eq. (9), the step size is called BB2.

$$\alpha_{BB2}=\underset{\alpha_n}{\mathrm{argmin}}\left\{\left\|\frac{1}{\alpha_n}(x_n\text{-}x_{n\text{-}1})\text{-}(\mathrm{g}(x_n)\text{-}\ \mathrm{g}(x_{n\text{-}1}))\right\|_2\right\}=\frac{\|x_n\text{-}x_{n\text{-}1}\|_2}{(x_n\text{-}x_{n\text{-}1})^{\mathrm{T}}(\mathrm{g}(x_n)\text{-}\ \mathrm{g}(x_{n\text{-}1}))}. \tag{11}$$

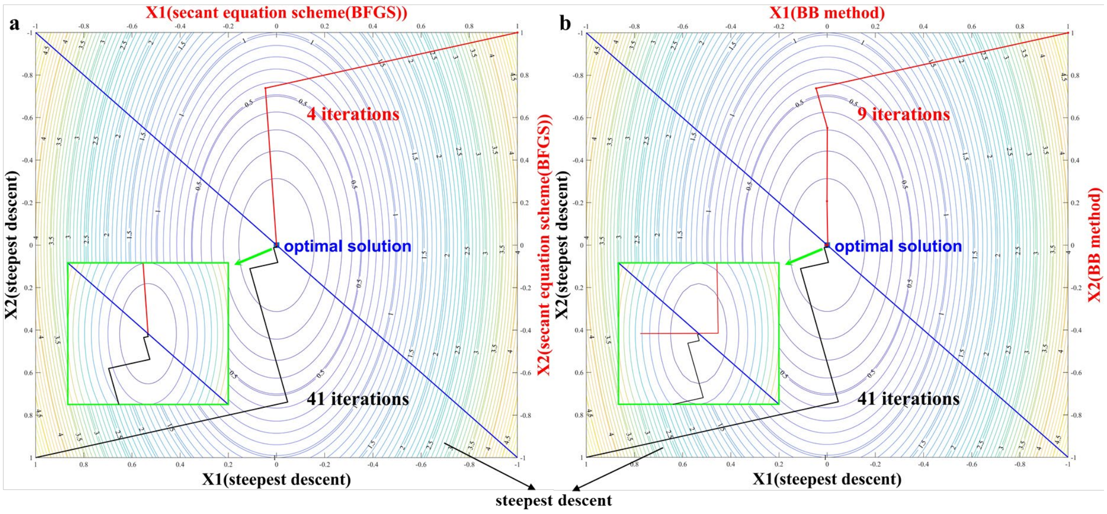


Figure 2 Solution trajectories of steepest descent, secant equation and BB methods

An example is employed to illustrate the performance of steepest descent, secant equation and BB methods. The objective function is $f(x_1, x_2) = 4x_1^2 + x_2^2$ and the initial point is (1, 1). Figure 2 shows the approximate solution trajectories of these three methods. The secant equation scheme adopted here is the BFGS method[1] with an exact line search. It is clear that the steepest descent method suffers from the zigzag phenomenon and consumes 41 iterations to achieve convergence. The BFGS and BB methods do not show this behavior and converge much faster (take only 4 and 9 iterations, respectively). It is noteworthy that the objective function value is not monotonically decreasing in the BB method from Figure 2b. There is a slight increase in the objective function value, resulting from an aggressive movement along the X1 axis. Although the BFGS seems to be more efficient, it involves many matrix operations and is usually bundled with a step size determination (e.g., using an exact or inexact line search), which requires frequent evaluations of the objective function and/or gradient value. On the other hand, the calculation of the BB step size is quite simple, as indicated by Eqs. (10) and (11). Consequently, the BB step size could be more appealing for solving large-scale problems.

The BB step size has been extended to solve other mathematical problems, e.g., unconstrained/constrained system of equations (Cruz and Raydan, 2003; Liu and Feng, 2019) and variational inequality with convex constraints (He et al., 2012). Also, the BB step size has been employed in practice to solve the travel demand forecasting problem (Gibb, 2016). Besides, the convergence of the BB step size has been extensively discussed by Barzilai and Borwein (1988), Raydan (1993) and Yuan (2008). Interested readers can refer to these papers for more details. The motivation of the BB step size scheme is summarized in Figure 3.

---

[1] The BFGS method, named for its discoverers Broyden, Fletcher, Goldfarb and Shanno, is one of the most prevailing secant equation algorithms to date.

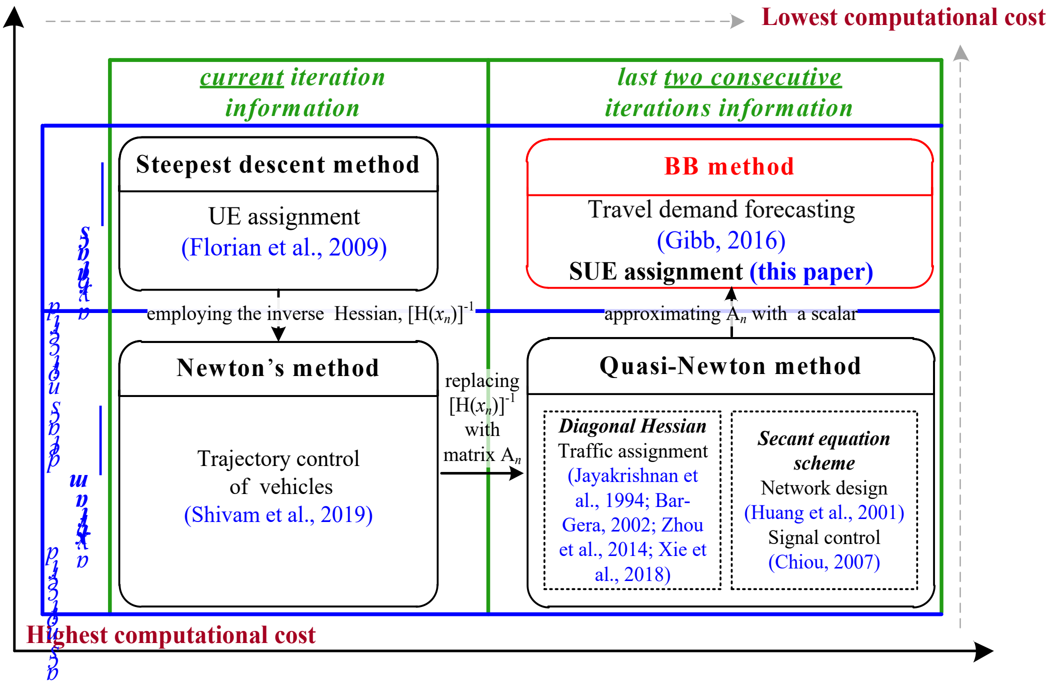


FIGURE 3 Derivation of the BB step size scheme with applications

From Figure 3, the steepest descent method is simple to implement, but it suffers from the zigzag phenomenon in ill-conditioned problems. Newton's method defines a new search direction with the inverse Hessian matrix which, however, is expensive to compute in large-scale problems. With a relatively modest computational cost per iteration, the quasi-Newton method alternatively employs either a diagonal Hessian or a matrix that satisfies the secant equation in Eq. (7). Furthermore, in order to avoid the second-order derivative computation or frequent matrix operations, the BB step size simplifies the quasi-Newton method by replacing matrix A_n with a scalar α_n. In summary, the steepest descent and BB methods use the gradient direction, while the Newton and Quasi-Newton methods modify the search direction with a matrix, which in general will take more computational cost per iteration. The steepest descent method takes more time per iteration than the BB method due to the frequent function evaluations by the exact/inexact line search schemes. On the other hand, the steepest and Newton's methods only use the information in the current iteration, but the secant method and BB step size utilize the information of the last two consecutive iterations.

3. Two Logit-based SUE Models and Step Size Determination Schemes

In this section, we first give a brief introduction to two SUE models, the multinomial logit (MNL) and cross-nested logit (CNL) SUE models. The former is a widely used SUE model, but it suffers from two drawbacks (i.e., path overlapping and identical perception variance problems); the latter is an extension of the MNL SUE model to overcome the path overlapping drawback (Prashker and Bekhor, 1999, 2004). In addition, some prevailing step size determination schemes for solving the SUE problem are also reviewed.

3.1. Two well-known SUE models

3.1.1. Multinomial logit (MNL) SUE model

As introduced, the SUE principle incorporates a random error term into the travel cost. Usually, the term is assumed to follow the Gumbel, Normal, and Weibull distributions, which correspond to the logit-based, probit-based and weibit-based path choice models respectively. The logit-based SUE model is perhaps the most widely used in the transportation literature (Prashker & Bekhor, 2004; Kitthamkesorn & Chen, 2013).

According to the logit-based SUE model, the probability of choosing path k, P_k, can be expressed as

$$P_k = \frac{\exp(-\theta c_k)}{\sum_{l\in K_{rs}}\exp(-\theta c_l)}, \forall k\in K_{rs}, r\in R, s\in S \tag{12}$$

where c_k is the travel cost on path k, K_{rs} denotes the active path set between origin r and destination s, and θ is a positive dispersion parameter, which reflects an aggregate measure of travelers' perception of travel costs. Fisk (1980) developed the following mathematical programming (MP) formulation for the MNL SUE problem.

$$min\, z = \sum_{a\in A}\int_0^{x_a} t_a(\omega)\, d\omega + \frac{1}{\theta}\sum_{r\in R}\sum_{s\in S}\sum_{k\in K_{rs}} f_k^{rs} ln f_k^{rs} \tag{13}$$

subject to

$$\sum_{k\in K_{rs}} f_k^{rs} = q_{rs}, \forall r\in R, s\in S \tag{14}$$

$$f_k^{rs} \geq 0, \forall k\in K_{rs}, r\in R, s\in S \tag{15}$$

$$x_a = \sum_{r\in R}\sum_{s\in S}\sum_{k\in K_{rs}} f_k^{rs}\delta_{a,k}^{rs}, \forall a\in A \tag{16}$$

where A is the set of links, R and S denote the origin and destination set respectively, x_a is the flow on link a, the cost of link a is denoted by t_a, which is a function of x_a, f_k^{rs} is the flow on path k and $\delta_{a,k}^{rs}$ is a link/path indicator, which equals 1 if link a is on path k, and 0 otherwise. Eqs. (14)-(16) are the conservation, non-negativity and definitional constraints, respectively.

The MNL SUE model is straightforward and easy to implement in practice. However, it has some inherent drawbacks. The independently and identically distributed (IID) assumption of the logit-based SUE model implies that all paths connecting an O-D pair are unrelated to each other and travelers have the same perception variance on all available paths. Consequently, the logit-based SUE model is not capable of accounting for path overlapping and perception variance with respect to different trip lengths. To overcome the path overlapping problem and to keep a closed-form probability expression, various extended logit-based models have been developed, for example, C-logit (Cascetta et al., 1996), cross-nested logit (CNL) (Vovsha & Bekhor, 1998), paired combinatorial logit (PCL) models (Prashker & Bekhor, 1999; Pravinvongvuth & Chen, 2005), and path-size logit (Ben-Akiva et al., 1999; Chen et al., 2012). In the following section, we will introduce the CNL SUE model, which is considered as the most flexible and general SUE model among the four extended logit-based SUE models (Prashker & Bekhor, 2004).

3.1.2. Cross-Nested logit (CNL) SUE model

The CNL model is one of the well-known extended logit-based models. It was originally proposed by Vovsha (1997) to handle the mode similarity issue in the mode choice problem. Vovsha & Bekhor (1998) and Prashker and Bekhor (1999) further exploited it to model travelers' path choice behavior and developed an equivalent MP formulation for the CNL SUE problem. This model is designed to overcome the inability of the MNL model to account for the path overlapping problem. The CNL model adopts a two-level structure in which the upper level consists of links (called nest) and the lower level represents the paths. This structure decomposes the path choice probability into the marginal probability and the conditional probability as follows:

$$P(k)=\sum_{m} P(m)P(k|m)\,,\ \forall m\in A,\, k\in K_{rs},\, r\in R,\, s\in S \tag{17}$$

where $P(m)$ is the marginal probability of nest m being chosen which can be expressed as

$$P(m)=\frac{(\sum_l (\alpha_{ml}\exp(-\theta c_l))^{1/\mu})^{\mu}}{\sum_b(\sum_{l\in K_{rs}} (\alpha_{bl}\exp(-\theta c_l))^{1/\mu})^{\mu}}\,,\ \forall m\in A,\, k\in K_{rs},\, r\in R,\, s\in S \tag{18}$$

where θ reflects an aggregate measure of drivers' perception of travel costs, and μ is the degree of nesting, $0\le\mu\le 1$, and α_{mk} is the inclusion coefficient allocating alternatives to nests.

$P(k|m)$ is the conditional probability of path k given a nest (link) has been chosen and can be calculated by

$$P(k|m)=\frac{(\alpha_{mk}\exp(-\theta c_k))^{1/\mu}}{\sum_{l\in K_{rs}} (\alpha_{ml}\exp(-\theta c_l))^{1/\mu}}\,,\ \forall m\in A,\, k\in K_{rs},\, r\in R,\, s\in S \tag{19}$$

Prashker and Bekhor (1998) suggested the following specification:

$$\alpha_{mk}=(\frac{L_m}{L_k})^{\gamma}\delta_{mk} \tag{20}$$

where L_m and L_k are the lengths of link m and path k, respectively; δ_{mk} equals to 1 if link m is on path k and 0, otherwise; γ is a parameter that reflects drivers' perception of similarity among paths and is assumed to be 1 in this case, which aims at forcing the inclusion coefficient to satisfy the regularity constraint

$$\sum_{m} \alpha_{mk}=1. \tag{21}$$

Here, a simple network is employed to explain the differences between the MNL and CNL models and how the CNL model accounts for the path overlapping issue. Figure 4 shows a simple network where there are four links and three paths connecting origin r and destination s. Path 1 is independent with the others, while path 2 and path 3 have a common link B. As shown in Figure 4, the MNL model assumes that path 2 and path 3 are irrelevant to each other, which indicates that the MNL model is not capable of capturing the path overlapping problem, because it views all paths are independent. In the CNL model, the upper level consists of the nests (links) and the lower level shows the paths may belong to several nests. Consequently, each path is decomposed into several parts corresponding to each nest it belongs to, and the inclusion coefficients represent the proportion of each part. In this two-level hierarchical structure, paths that share common links are explicitly captured.

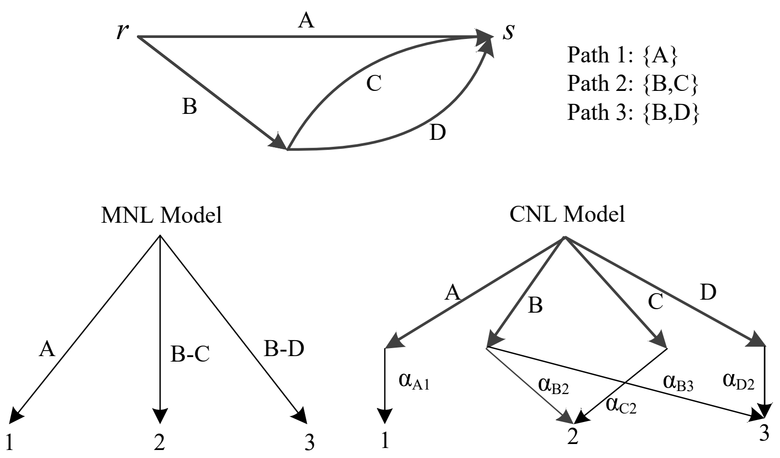


FIGURE 4 Illustration of path overlapping in MNL and CNL Models

Prashker and Bekhor (1999) developed an equivalent MP formulation for the CNL SUE model as follows:

$$min\ z = z_1 + z_2 + z_3 \tag{22}$$

$$z_1 = \sum_{a \in A} \int_0^{x_a} t_a(\omega) d\omega \tag{23}$$

$$z_2 = \frac{\mu}{\theta} \sum_{rs} \sum_{m} \sum_{k} f_{mk}^{rs} \ln \frac{f_{mk}^{rs}}{(\alpha_{mk}^{rs})^{1/\mu}} \tag{24}$$

$$z_3 = \frac{1-\mu}{\theta} \sum_{rs} \sum_{m} (\sum_{k} f_{mk}^{rs}) \ln (\sum_{k} f_{mk}^{rs}) \tag{25}$$

subject to

$$\sum_{m} \sum_{k} f_{mk}^{rs} = q_{rs},\ \forall m, k \in K_{rs}, r \in R, s \in S \tag{26}$$

$$f_{mk}^{rs} \geq 0,\ \forall m, k \in K_{rs}, r \in R, s \in S \tag{27}$$

where f_{mk}^{rs} is the flow on path k of nest m between origin r and destination s. The term, $f_{mk}^{rs} \ln \frac{f_{mk}^{rs}}{(\alpha_{mk}^{rs})^{1/\mu}}$, is defined as zero if either $f_{mk}^{rs} = 0$ or $\alpha_{mk}^{rs} = 0$. It can also be observed from Eq. (18) that $f_{mk}^{rs} = 0$ once α_{mk}^{rs} equals to 0 (i.e., link m is not on path k).

Compared with the MNL SUE model, there are two entropy terms in the CNL SUE model, which correspond to the two-level hierarchical structure of the CNL path choice model (i.e., z_2 corresponds to the conditional probability in Eq. (18) and z_3 corresponds to the marginal probability in Eq. (19)). Another main difference is the decision variable (f_{mk}^{rs}) defined as the flow on path k of nest m between origin r and destination s, which is more complex than the simple path flow defined in the MNL SUE model. For more details, see Prashker & Bekhor (1999).

3.2. Step size determination for solving the logit-based SUE models

The classical iterative approach of solving the SUE problem is to identify a search direction first and then determine a step size to guide how far the solution should move along the direction.

The search direction, in the standard SUE solution algorithm, is typically determined by obtaining an auxiliary solution at each iteration. Thus, much attention has been paid to design good step size determination schemes to improve solution efficiency. In this section, we briefly introduce the standard search direction for solving the logit-based SUE models, and focus on reviewing some existing step size determination schemes.

In the SUE traffic assignment problem, the path flow pattern is determined by a path choice model, which can be expressed as follows (for convenience, we omit the superscript related to OD pairs)

$$f_k = qP_k(\mathbf{f}), \forall k \in K \tag{28}$$

where $P_k(\mathbf{f})$ represents the probability of path k being chosen, and is a function of the flows on all paths represented by vector **f**. Note that the solution of Eq. (28) satisfies both the demand conservation and non-negativity constraints due to the properties of probability theory (i.e., $P_k(\mathbf{f}) \geq 0$, $\forall k \in K$ and $\sum_{k \in K} P_k(\mathbf{f}) = 1$) despite that these constraints have not been explicitly presented.

If we denote $qP_k(\mathbf{f})$ by $F_k(\mathbf{f})$ and rewrite Eq. (28) in a vector form as

$$\mathbf{f} = \mathbf{F}(\mathbf{f}) \tag{29}$$

where $\mathbf{F}(\mathbf{f})$ denotes the vector $(\ldots, F_k(\mathbf{f}), \ldots)$. The SUE problem is equivalent to a fixed-point problem (Cantarella et al., 2015).

For a fixed-point problem given in Eq. (29), the standard solution procedure can be represented as

$$f_k^{n+1} = f_k^n + \alpha(F_k(\mathbf{f}^n) - f_k^n), \forall k \in K \tag{30}$$

where f_k^n denotes the flow on path k at iteration n, $\mathbf{f}^n$ is the path flow pattern at iteration n and α denotes the step size. The term $F_k(\mathbf{f}^n)$ (i.e., $qP_k^n(\mathbf{f}^n)$) $\forall k \in K$, called the *auxiliary solution*, is to allocate the demand q onto the network based on the path flow pattern $\mathbf{f}^n$. The procedure for computing the auxiliary solution is called *stochastic network loading*. Note that if the vector $F_k(\mathbf{f}^n) - f_k^n$, $\forall k \in K$ is equal to zero, the current flow pattern is a solution of Eq. (29); else, it defines a new search direction towards the auxiliary solution.

The efficiency of the above solution procedure heavily relies on the step size determination, and therefore it has drawn great attention to design good step size determination schemes. Sheffi and Powell (1981) employed the method of successive averages (MSA) to solve the probit-based SUE problem. The MSA determines the step size as $\alpha_n = 1/n$, where n is the iteration number. The MSA scheme satisfies the Blum's Theorem (Blum, 1954) (i.e., $\alpha_n \to 0$, $\sum_{n=0}^{\infty} \alpha_n = \infty$) and guarantees the convergence. It is simple to implement as it avoids evaluating the objective function value and/or its gradient. However, it has been found that the MSA step size gets too small when the iteration number is large, which may lead to slow convergence speed. Some researchers have proposed alternative predetermined line search schemes to slow down the decrease of the step sizes. For instance, Polyak (1990) suggested a new predetermined step size sequence as $\alpha_n = n^{-2/3}$, and Nagurney and Zhang (1996) proposed the following sequence $\{\alpha_n\} = \{1, \frac{1}{2}, \frac{1}{2}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \ldots, n \text{ repetitions of } \frac{1}{n}\}$. Following this, Liu et al. (2009) pointed out that these predetermined schemes share the same inherent drawback: the step size may be too large or too small at some iterations so that the next solution gets farther away from the optimal solution than the previous one or moves towards the optimal solution quite slowly. In order to overcome this drawback whilst maintaining its simplicity, Liu et al. (2009) developed a self-regulated averaging (SRA) scheme on the basis of the MSA scheme as follows:

SRA scheme: This scheme utilizes the information of consecutive iterations to guide the choice of step size to either "speed up" or "slow down". The absolute residual error, $\|y_n - x_n\|$, is used to monitor the convergence. The step size of SRA is calculated by

$$\alpha_n = \frac{1}{\mu_n},\ \mu_n = \begin{cases} \mu_{n-1} + \psi,\ \psi > 1, \text{ if } \|y_n - x_n\| \geq \|y_{n-1} - x_{n-1}\| \\ \mu_{n-1} + \varphi,\ \varphi < 1, \text{ if } \|y_n - x_n\| < \|y_{n-1} - x_{n-1}\| \end{cases} \tag{31}$$

where y_n is the auxiliary solution based on the current flow pattern, and ψ and φ are two parameters. When the absolute error $\|y_n - x_n\|$ gets larger (i.e., the iteration tends to diverge), the step size, α_n, will decrease quickly (ψ is active); otherwise, the step size should decrease slowly (φ is active). The SRA scheme also satisfies Blum's theorem (Blum, 1954) to guarantee the convergence.

Liu et al. (2009) conducted several numerical experiments and reported that the SRA scheme outperforms the aforementioned predetermined schemes. Besides, the SRA scheme has also been adopted to solve various SUE models, e.g., the C-logit SUE problem with elastic demand (Xu & Chen, 2013), the PCL SUE problem (Chen et al., 2014), the unconstrained weibit-based SUE problem (Kitthamkesorn & Chen, 2014), the unconstrained combined distribution and assignment problem (Yao et al., 2014), and the weibit-based SUE problem with elastic demand (Kitthamkesorn et al., 2015). The results in these studies verified the good performance of the SRA scheme over the MSA scheme.

On the other hand, other studies attempted to determine the step size based on the principle that reduces the objective function along the given search direction, instead of focusing on modifying the MSA scheme. Chen and Alfa (1991) suggested the exact line search method for determining the step size, while Maher (1998) utilized a quadratic interpolation technique and put forward the optimal step length algorithm (OSLA) for the MNL SUE model. Recently, Bekhor and Toledo (2008) and Chen et al. (2014) adopted Armijo's rule for solving the CNL and PCL SUE models, respectively. Both reported promising numerical results. Armijo's rule is reviewed as follows.

Armijo's rule: Armijo's rule (Armijo, 1966) was originally proposed for unconstrained optimization problems. Bertsekas (1976) developed a generalized Armijo strategy for constrained optimization problems. Recently, Chen et al. (2013) incorporated a self-adaptive technique, which had been adopted to solve the TA problem with path-specific costs (Chen et al., 2001), into the Armijo strategy (i.e., the self-adaptive Armijo (SAA) strategy). The SAA scheme can adjust the starting step size at each iteration based on certain rules. However, more parameters need to be calibrated. Hence, the generalized Armijo strategy is adopted as follows:

Given that $\mathbf{x}_k$ is not an optimal solution, set

$$\alpha_k = \beta^m \gamma_k \tag{32}$$

where m_k is the first non-negative integer m such that

$$\mathrm{Z}(\mathbf{x}_k) - \mathrm{Z}\big(\mathbf{x}_k(\beta^m \gamma_k)\big) \geq \sigma \nabla \mathrm{Z}(\mathbf{x}_k)^T \big(\mathbf{x}_k - \mathbf{x}_k(\beta^m \gamma_k)\big) \tag{33}$$

where $\sigma \in (0, 1)$ and $\beta \in (0, 1)$ are fixed scalars, γ_k is the upper bound, $\mathrm{Z}(\cdot)$ represents the objective function value and $\mathbf{x}_k(\beta^m \gamma_k)$ is a new approximate solution based on $\mathbf{x}_k$ and $\beta^m \gamma_k$. Inequality (33) is also called the *adequate decrease condition*. As an inexact line search scheme, Armijo's rule typically requires fewer evaluations of the objective function and/or its gradient compared to the exact line search schemes, e.g., the golden section and the bi-section methods.

In this section, we introduce the residual error as the search direction by reformulating the SUE problem into a fixed-point problem. In the literature, there are different methods to determine the search direction. Damberg et al. (1996) adapted the disaggregate simplicial decomposition (DSD) algorithm for solving the MNL SUE problem. By partially linearizing the objective function, the DSD algorithm defines the difference between the auxiliary and the current flow patterns as the search direction. The auxiliary flow pattern is obtained by reassigning the demand based on the current cost pattern. The search direction of the DSD algorithm coincides with the residual error. However, the DSD algorithm can only be applied to solve SUE models formulated as a mathematical program since it needs to partially linearize the objective function, and our approach can be extended to more general cases (e.g., the fixed-point problem). Bekhor and Toledo (2005) also adapted the gradient projection (GP) algorithm to solve the MNL SUE model. At each iteration, the GP algorithm approximates the original problem with a quadratic program and replaces the Hessian matrix with its diagonal elements so as to easily solve the quadratic subproblem. For each subproblem, the search direction is determined by projecting the gradient (scaled by the diagonal Hessian) onto the equality constraints. Yu et al. (2014) proposed the interior-point method to solve the MNL SUE model with elastic demand. The interior-point method identifies the search direction by solving a linear system of equations. Nevertheless, both the GP and interior-point methods need to compute the second-order derivative, which is expensive for the SUE problem embedded with advanced discrete choice models (e.g., the CNL model) on realistic networks (Chen et al., 2014). In conclusion, these alternative search directions maybe not very appealing for solving extended logit-based SUE models on realistic networks.

Here we should also clarify that the BB scheme is not applicable for all descent directions (e.g., the aforementioned GP and interior-point methods). Recall that the BB scheme was designed to incorporate into the gradient method in an unconstrained optimization problem. In the next section, we will develop an efficient path-based traffic assignment algorithm embedded with the BB step size scheme to solve the SUE problem. Besides, we will also show that restricting the search direction does not impede the extendibility of our algorithm.

4. A Faster Path-based Traffic Assignment Algorithm with the BB Step Size Scheme

In this section, we will link the SUE problem to the system of equations and embed the BB step size scheme into a path-based traffic assignment algorithm to solve the SUE problem in a simple and elegant way. Then we will conduct the convergence analysis of the proposed algorithm for solving SUE models which shows that the convergence can be ensured under some mild assumptions.

4.1. The implementation of the proposed algorithm

As discussed in Section 3, the SUE problem can be equivalently represented as a fixed-point problem in Eq. (29). Specifically, based on the MNL and CNL path choice models in Eq. (12) and Eqs. (17)-(19), respectively, the network flow pattern defined by the MNL and CNL

SUE models can be written as

$$\textbf{MNL}\ f_k = q_{rs} \cdot \frac{\exp(-\theta c_k)}{\sum_{l \in K_{rs}} \exp(-\theta c_l)},\ \forall k \in K_{rs},\ r \in R,\ s \in S \tag{34}$$

$$\textbf{CNL}\ f_k = q_{rs} \cdot \frac{\sum_m (\alpha_{mk}\exp(-\theta c_k))^{1/\mu}\left(\sum_{l \in K_{rs}} (\alpha_{ml}\exp(-\theta c_l))^{1/\mu}\right)^{\mu-1}}{\sum_m\left(\sum_{l \in K_{rs}} (\alpha_{ml}\exp(-\theta c_l))^{1/\mu}\right)^{\mu}}, \tag{35}$$

$$\forall k \in K_{rs}, \forall m \in A,\ r \in R,\ s \in S$$

Note that the path cost c_k is a function of the path flow pattern $\mathbf{f}$. Hence, Eqs. (34) and (35) can be simply expressed as a fixed-point problem in Eq. (29). Furthermore, Eq. (29) can be regarded as a system of equations and rewritten in vector form as follows:

$$\mathbf{f} - \mathbf{F}(\mathbf{f}) = 0. \tag{36}$$

Recall that the BB step size was derived on the basis of solving the necessary conditions of the unconstrained optimization problem. This provides a way to solve the system of equations $g(x) = 0$ in Eq. (3), which allows us to bridge the BB step size and the SUE problem. As a result, the aforementioned formulas in Section 2 can be directly used to solve Eq. (36). According to Eq. (8), the BB scheme defines the following iterative formula:

$$\mathbf{f}^{n+1} = \mathbf{f}^{n} + \alpha_{\text{BB}}(\mathbf{F}(\mathbf{f}^{n}) - \mathbf{f}^{n}). \tag{37}$$

Following Eqs. (10) and (11), the calculation of α_{BB} is given by

$$\alpha_{\text{BB1}} = \frac{(\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\text{T}}[(\mathbf{f}^{n}-\mathbf{f}^{n-1}) - (\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1}))]}{\left\|(\mathbf{f}^{n}-\mathbf{f}^{n-1}) - (\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1}))\right\|_2} \tag{38}$$

and

$$\alpha_{\text{BB2}} = \frac{\left\|\mathbf{f}^{n}-\mathbf{f}^{n-1}\right\|_2}{(\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\text{T}}[(\mathbf{f}^{n}-\mathbf{f}^{n-1}) - (\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1}))]}. \tag{39}$$

It can be observed from Eq. (37) that the search direction is consistent with the one of Eq. (30). However, the BB scheme attempts to obtain the derivative information of the mapping function, instead of modifying the MSA scheme (e.g., the SRA scheme) or pursuing an adequate decrease in the objective function value (e.g., the Armijo scheme). Compared with other step size determination schemes, the BB scheme only needs to record the mapping function values of two consecutive iterations, without any parameter settings and requirement to evaluate the objective function value and/or its derivatives.

There is still a potential problem that remains unsolved in the iterative procedure. The term $\mathbf{F}(\mathbf{f}^{n})$ in Eq. (37) refers to the loading of the traffic demand onto the network based on the current flow pattern (i.e., stochastic network loading), which implies that every approximate solution $\mathbf{f}^{n}$ should also satisfy the demand conservation and non-negativity constraints.

We find that every approximate solution defined by the BB scheme indeed satisfies the constraints, which is proven below.

Proposition 1. *For solving the SUE problem, every approximate solution produced by Eqs. (37)-(39) satisfies the demand conservation and non-negativity constraints.*

Proof. The first approximate solution $\mathbf{f}^{0}$ is initialized by the stochastic network loading, which implies that the initialization solution satisfies the constraints. Besides, the constraint set is convex. Hence, we only need to prove that the new approximate solution $\mathbf{f}^{n+1}$ is the convex combination of the current approximate solution $\mathbf{f}^{n}$ and the auxiliary one $\mathbf{F}(\mathbf{f}^{n})$, i.e., $\alpha_{BB} \in (0,1]$.

Let $\mathbf{P}$ denote the path choice probability vector, and thus the path flow pattern can be given by $q\mathbf{P}$. Note that $\nabla\mathbf{P}$ is the Hessian matrix of the satisfaction function which is concave with respect to path flows (see Chapter 10 in Sheffi (1985) for details). Hence, the Jacobian matrix $\nabla\mathbf{P}$ (with respect to path flows) is semi-negative definite. From the Taylor's expansion, we obtain

$$\mathbf{P}^{n} = \mathbf{P}^{n-1} + \nabla\mathbf{P}^{*}(\mathbf{f}^{n}-\mathbf{f}^{n-1}) \tag{40}$$

where $\nabla\mathbf{P}^{*}$ denotes the Jacobian matrix at point $\vartheta\mathbf{f}^{n}$-$(1-\vartheta)\mathbf{f}^{n-1}$, $\vartheta\in(0,1)$. Clearly, $\nabla\mathbf{P}^{*}$ is semi-negative and the following inequality is obtained

$$(\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\mathrm{T}}(\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1})) = q(\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\mathrm{T}}(\mathbf{P}^{n}-\mathbf{P}^{n-1}) = q(\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\mathrm{T}}\nabla\mathbf{P}^{*}(\mathbf{f}^{n}-\mathbf{f}^{n-1}) \le 0. \tag{41}$$

It can be inferred from inequality (41) that α_{BB1} will satisfy

$$\alpha_{BB1} = \frac{(\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\mathrm{T}}(\mathbf{f}^{n}-\mathbf{f}^{n-1}) - (\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\mathrm{T}}(\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1}))}{\left\|(\mathbf{f}^{n}-\mathbf{f}^{n-1}) - (\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\mathrm{T}}(\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1}))\right\|_2} > 0 \tag{42}$$

and

$$\begin{aligned}\alpha_{BB1} &= \frac{(\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\mathrm{T}}(\mathbf{f}^{n}-\mathbf{f}^{n-1}) - (\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\mathrm{T}}(\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1}))}{\left\|(\mathbf{f}^{n}-\mathbf{f}^{n-1}) - (\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1}))\right\|_2} \\ &\le \frac{\left\|(\mathbf{f}^{n}-\mathbf{f}^{n-1})\right\|_2 - 2(\mathbf{f}^{n}-\mathbf{f}^{n-1})^{\mathrm{T}}(\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1})) + \left\|(\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1}))\right\|_2}{\left\|(\mathbf{f}^{n}-\mathbf{f}^{n-1}) - (\mathbf{F}(\mathbf{f}^{n})-\mathbf{F}(\mathbf{f}^{n-1}))\right\|_2} = 1.\end{aligned} \tag{43}$$

Inequalities (42) and (43) indicate that α_{BB1} ranges from 0 to 1. Similarly, the step size, α_{BB2}, has the same property. This completes the proof of Proposition 1.

Based on the above discussion, the flowchart for implementing the path-based traffic assignment algorithm with the BB step size is presented in Figure 5. Note that the BB step size takes the information of two consecutive iterations. In the first iteration (n=1), ρ can be obtained by an exact or inexact step size determination or other approaches. In this study, we simply set it to the value consistent with the SRA scheme. In Figure 5, $\mathbf{c}^{n}$ denotes the vector of path costs at the nth iteration; $\mathbf{\Delta}$ is the incidence matrix describing the relationship between links and paths; $\mathbf{t}(\cdot)$ is the vector of link cost functions.

Remark 1: Eqs. (36)-(39) extend the BB step size to solve the fixed-point problem which is associated with the system of equations. The search direction is determined by the residual error, which coincides with the search direction proposed by Damberg et al. (1996). However, our approach can solve not only the SUE problems formulated as a MP, but also more general mathematical formulations (e.g., fixed-point problem) that can account for more complex cost structures (e.g., non-separable link cost functions, non-additive path cost functions, etc.). In this sense, the principle of the BB scheme provides a simple and powerful tool for solving many traffic equilibrium problems formulated as a fixed-point problem with a point-to-point mapping function. Note that this method can be easily adapted to solve the SUE problem with elastic demand. The elasticity of demand is characterized by a monotonically decreasing function with respect to the expected minimum travel cost for each O-D pair (Yu et al., 2014). Since the travel cost is decided by the flow pattern, the SUE problem with elastic demand can be represented like Eq. (28) as follows

$$f_k = q(\mathbf{f})P_k(\mathbf{f}),\ \forall k \in K. \tag{44}$$

Clearly, Eq. (44) has the same form as Eq. (36) and hence can be directly solved by the proposed algorithm. Due to the monotonicity of the mapping function $-q(\mathbf{f})P_k(\mathbf{f}),\ \forall k \in K$, the properties are also applicable in this case. More extended applications of the proposed method will be discussed in Section 6.

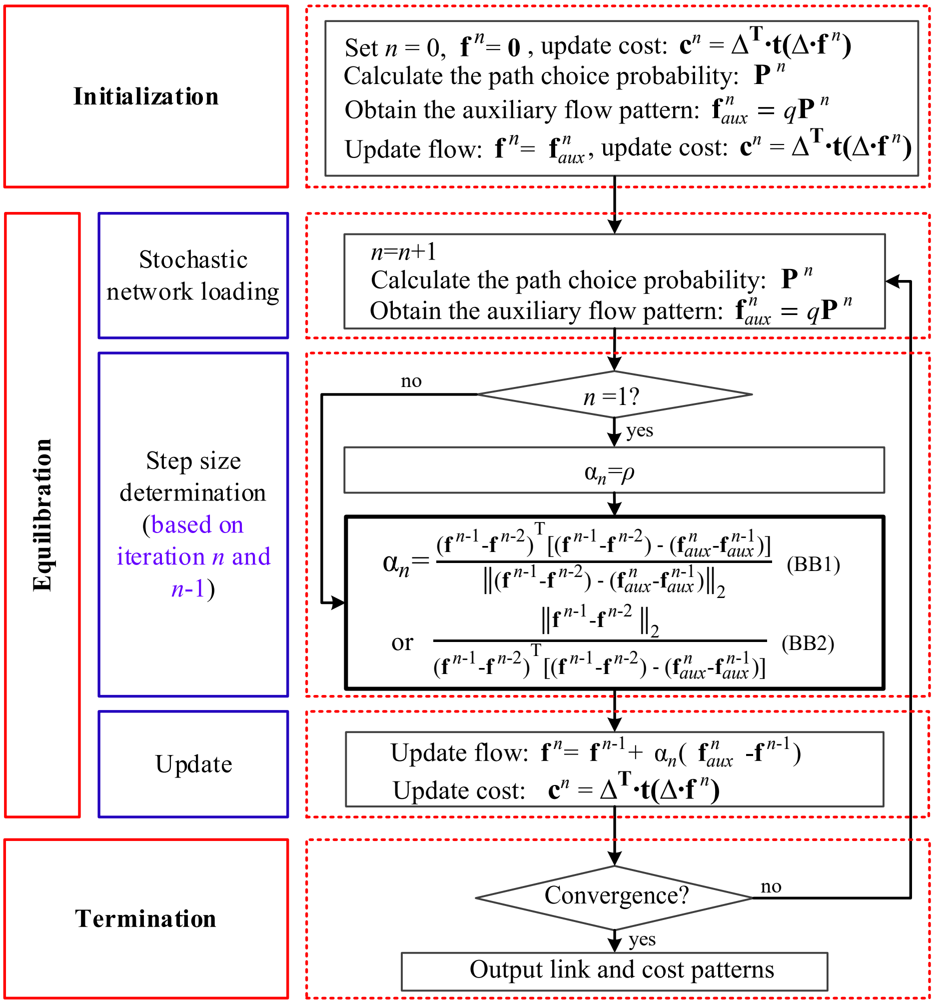


FIGURE 5 Flowchart for implementing the path-based traffic assignment algorithm with the BB step size

Remark 2: The BB step size is preferred to the SRA and Armijo schemes in several ways. For the SRA scheme, the information in two consecutive iterations may not be fully utilized. According to Eq. (30), only the absolute residual is obtained to determine the choice of the step size. Moreover, the step size sequence of this scheme is monotonically decreasing. If the step size decreases sharply in the early iterations, the overall convergence speed will be rather slow.

For the Armijo method, the main drawback is the requirement to evaluate the objective function and/or its gradients. When the objective function is complicated (e.g., the CNL SUE model), the computation cost of function evaluations is quite expensive, which will lead to poor performance. In addition, another drawback of both schemes is that their convergence performance heavily relies on setting the initial parameter values. Inappropriate parameter settings may greatly affect the convergence speed. Consequently, the BB step size seems quite appealing for solving the SUE problem under congested and realistic networks.

4.2. *Convergence results*

Raydan (1993) has established the convergence of the BB method when the mapping function $\mathbf{T}(\mathbf{f}) = \mathbf{f} - \mathbf{F}(\mathbf{f})$ is affine and its Jacobian matrix is symmetric positive definite (i.e., it is equivalent to a MP formulation whose objective function is convex quadratic). Despite that the BB method is often shown to converge in practical applications (He et al., 2012), it still lacks a theoretical convergence proof to show that the BB method converges in the case of fixed-point problem. Here we provide the convergence results for solving the SUE problem under some mild assumptions.

The following proposition presents some useful properties of the mapping function in the SUE problem.

Proposition 2. *The mapping function* $\mathbf{T}(\mathbf{f}) = \mathbf{f} - \mathbf{F}(\mathbf{f})$ *is strongly monotone and Lipschitz continuous, i.e.,* $(\mathbf{T}(\mathbf{f}_1)\text{-}\mathbf{T}(\mathbf{f}_2))^{\mathrm{T}}(\mathbf{f}_1\text{-}\mathbf{f}_2) \geq \eta(\mathbf{f}_1\text{-}\mathbf{f}_2)^{\mathrm{T}}(\mathbf{f}_1\text{-}\mathbf{f}_2)$ *and* $\|\mathbf{T}(\mathbf{f}_1)\text{-}\mathbf{T}(\mathbf{f}_2)\| \leq L\|\mathbf{f}_1\text{-}\mathbf{f}_2\|$ *for any* $\mathbf{f}_1$ *and* $\mathbf{f}_2$ *in the feasible region where* $\|\cdot\|$ *represents the Euclidean-norm.*

Proof. Note that Eq. (41) indicates that $-\mathbf{F}(\mathbf{f})$ is monotone. Obviously, $\mathbf{T}(\mathbf{f})$ is strongly monotone. Specifically, according to inequality (41) and the definition of $\mathbf{T}(\mathbf{f})$, we have

$$(\mathbf{T}(\mathbf{f}_1)\text{-}\mathbf{T}(\mathbf{f}_2))^{\mathrm{T}}(\mathbf{f}_1\text{-}\mathbf{f}_2) \geq (\mathbf{f}_1\text{-}\mathbf{f}_2)^{\mathrm{T}}(\mathbf{f}_1\text{-}\mathbf{f}_2). \tag{45}$$

Inequality (45) indicates that the largest strongly monotone modulus is not smaller than 1.

On the other hand, path flow variables are bounded from 0 to q_{rs}. Note that we can always find a sufficiently small amount of flow ε not larger than any path flow (i.e., the demand has an upper bound and every auxiliary path flows has a positive lower bound), which indicates the domain is bounded and closed. In this case, $\mathbf{F}(\mathbf{f})$ and $\mathbf{T}(\mathbf{f})$ are bounded and continuous. Thus, $\mathbf{T}(\mathbf{f})$ is Lipschitz continuous. This completes the proof of Proposition 2.

The following proposition describes some relationship between the mapping function values of two consecutive iterations.

Proposition 3. *The sequence* $\{\mathbf{f}^n\}$ *generated by Eq. (37) satisfies*

$$\left\|\mathbf{T}(\mathbf{f}^{n+1})\right\|_2 \leq (1+{\alpha_{\mathrm{BB}}^n}^2 L^2 - 2\alpha_{\mathrm{BB}}^n \eta)\|\mathbf{T}(\mathbf{f}^n)\|_2$$

Proof. $\left\|\alpha_{\mathrm{BB}}^n \mathbf{T}(\mathbf{f}^{n+1})\right\|_2 = \left\|(\mathbf{f}^n\text{-}\alpha_{\mathrm{BB}}^n \mathbf{T}(\mathbf{f}^n))\text{-}(\mathbf{f}^{n+1}\text{-}\alpha_{\mathrm{BB}}^n \mathbf{T}(\mathbf{f}^{n+1}))\right\|_2$

$$= \left\|(\mathbf{f}^n\text{-}\mathbf{f}^{n+1})\text{-}\alpha_{\mathrm{BB}}^n(\mathbf{T}(\mathbf{f}^n)\text{-}\mathbf{T}(\mathbf{f}^{n+1}))\right\|_2$$

$$= \left\|(\mathbf{f}^n\text{-}\mathbf{f}^{n+1})\right\|_2 \text{-} 2\alpha_{\mathrm{BB}}^n(\mathbf{f}^n\text{-}\mathbf{f}^{n+1})(\mathbf{T}(\mathbf{f}^n)\text{-}\mathbf{T}(\mathbf{f}^{n+1})) + \left\|\alpha_{\mathrm{BB}}^n(\mathbf{f}^n\text{-}\mathbf{f}^{n+1})\right\|_2$$

Invoking the strong monotonicity and Lipschitz continuity conditions of $\mathbf{T}(\cdot)$, we obtain

$$\left\|\alpha_{\mathrm{BB}}^n \mathbf{T}(\mathbf{f}^{n+1})\right\|_2 \leq \left\|(\mathbf{f}^n\text{-}\mathbf{f}^{n+1})\right\|_2 + \left({\alpha_{\mathrm{BB}}^n}^2 L^2 \text{-} 2\alpha_{\mathrm{BB}}^n \eta\right)\left\|(\mathbf{f}^n\text{-}\mathbf{f}^{n+1})\right\|_2$$

$=(1+{\alpha_{BB}^{n}}^{2}L^{2}\text{-}2\alpha_{BB}^{n}\eta)\|\alpha_{BB}^{n}\mathbf{T}(\mathbf{f}^{n})\|_{2}$

The proof is completed.

Proposition 4. *For n>1, two versions of the BB step size defined by Eqs. (38) and (39) are bounded with* $\alpha_{BB1}^{n}\in[\eta/L^{2},1/\eta]$ *and* $\alpha_{BB2}^{n}\in[1/L,\ 1/\eta]$.

Proof. For the BB1 scheme, it follows from the strong monotonicity and Lipschitz continuity conditions of $\mathbf{T}(\cdot)$ that

$$\alpha_{BB1}^{n}=\frac{(\mathbf{f}^{n}\text{-}\mathbf{f}^{n\text{-}1})^{\mathrm{T}}(\mathbf{T}(\mathbf{f}^{n})\text{-}\mathbf{T}(\mathbf{f}^{n\text{-}1}))}{\left\|\mathbf{T}(\mathbf{f}^{n})\text{-}\mathbf{T}(\mathbf{f}^{n\text{-}1})\right\|_{2}}\geq\frac{\eta(\mathbf{f}^{n}\text{-}\mathbf{f}^{n\text{-}1})^{\mathrm{T}}(\mathbf{f}^{n}\text{-}\mathbf{f}^{n\text{-}1})}{\left\|\mathbf{T}(\mathbf{f}^{n})\text{-}\mathbf{T}(\mathbf{f}^{n\text{-}1})\right\|_{2}}\geq\frac{\eta}{L^{2}}.$$

Using the Cauchy-Schwarz inequality and the strong monotonicity condition, we have

$$\alpha_{BB1}^{n}=\frac{(\mathbf{f}^{n}\text{-}\mathbf{f}^{n\text{-}1})^{\mathrm{T}}(\mathbf{T}(\mathbf{f}^{n})\text{-}\mathbf{T}(\mathbf{f}^{n\text{-}1}))}{\left\|\mathbf{T}(\mathbf{f}^{n})\text{-}\mathbf{T}(\mathbf{f}^{n\text{-}1})\right\|_{2}}\leq\frac{\left\|\mathbf{f}^{n}\text{-}\mathbf{f}^{n\text{-}1}\right\|}{\left\|\mathbf{T}(\mathbf{f}^{n})\text{-}\mathbf{T}(\mathbf{f}^{n\text{-}1})\right\|}\leq\frac{1}{\eta}$$

For the BB2 scheme, it follows from the Cauchy-Schwarz inequality and the Lipschitz continuity condition that

$$\alpha_{BB2}=\frac{\left\|\mathbf{f}^{n}\text{-}\mathbf{f}^{n\text{-}1}\right\|_{2}}{(\mathbf{f}^{n}\text{-}\mathbf{f}^{n\text{-}1})^{\mathrm{T}}[\mathbf{T}(\mathbf{f}^{n})\text{-}\mathbf{T}(\mathbf{f}^{n\text{-}1})]}\geq\frac{1}{L}$$

Invoking the strong monotonicity condition, we obtain

$$\alpha_{BB2}=\frac{\left\|\mathbf{f}^{n}\text{-}\mathbf{f}^{n\text{-}1}\right\|_{2}}{(\mathbf{f}^{n}\text{-}\mathbf{f}^{n\text{-}1})^{\mathrm{T}}[\mathbf{T}(\mathbf{f}^{n})\text{-}\mathbf{T}(\mathbf{f}^{n\text{-}1})]}\leq\frac{1}{\eta}$$

The proof is completed.

The following assumptions are necessary to derive our results.

Assumption 1. *The strongly monotone modulus* η *and the Lipschitz constant L satisfy*

$$L\leq\sqrt{\frac{\sqrt{\xi^{\mathrm{I}}+8}\text{-}\xi^{\mathrm{I}}}{2}}\eta,\quad \xi^{\mathrm{I}}\in(0,1).$$

Assumption 2. *The strongly monotone modulus* η *and the Lipschitz constant L satisfy* $L\leq\lambda\eta$, *where* λ *is a positive solution of the following equation:*

$$x^{2}\text{-}\frac{2}{x}+\xi^{\mathrm{II}}=0,\quad \xi^{\mathrm{II}}\in(0,1).$$

The following propositions give the convergence results.

Proposition 5. *If assumption 1 holds, the BB1 scheme is globally convergent to the solution of Eq. (36).*

Proof. It follows from Propositions 3 and 4 and assumption 1 that

$$\left\|\mathbf{T}(\mathbf{f}^{n+1})\right\|_{2}\leq\left(1+{\alpha_{BB1}^{n}}^{2}L^{2}\text{-}2\alpha_{BB1}^{n}\eta\right)\|\mathbf{T}(\mathbf{f}^{n})\|_{2}\leq\left(1+\frac{L^{2}}{\eta^{2}}\text{-}2\frac{\eta^{2}}{L^{2}}\right)\|\mathbf{T}(\mathbf{f}^{n})\|_{2}$$

$$\leq\left(1+\frac{\sqrt{\xi^{\mathrm{I}}+8}\text{-}\xi^{\mathrm{I}}}{2}\text{-}\frac{4}{\sqrt{\xi^{\mathrm{I}}+8}\text{-}\xi^{\mathrm{I}}}\right)\|\mathbf{T}(\mathbf{f}^{n})\|_{2}=\left(1\text{-}\xi^{\mathrm{I}}\right)\|\mathbf{T}(\mathbf{f}^{n})\|_{2}\leq\left(1\text{-}\xi^{\mathrm{I}}\right)^{n}\left\|\mathbf{T}(\mathbf{f}^{1})\right\|_{2}$$

We have $\left\|\mathbf{T}(\mathbf{f}^{n+1})\right\|_2=\left\|\frac{\mathbf{f}^{n+2}\text{-}\mathbf{f}^{n+1}}{\alpha_{\mathrm{BB1}}^{n+1}}\right\|_2\leq\left(1\text{-}\xi^{\mathrm{I}}\right)^n\left\|\mathbf{T}(\mathbf{f}^1)\right\|_2$, which indicates

$$\left\|\mathbf{f}^{n+2}\text{-}\mathbf{f}^{n+1}\right\|_2\leq{\alpha_{\mathrm{BB1}}^{n+1}}^2\left(1\text{-}\xi^{\mathrm{I}}\right)^n\left\|\mathbf{T}(\mathbf{f}^1)\right\|_2\leq\frac{1}{\eta^2}\left(1\text{-}\xi^{\mathrm{I}}\right)^n\left\|\mathbf{T}(\mathbf{f}^1)\right\|_2.$$

Since $\left(1\text{-}\xi^{\mathrm{I}}\right)\in(0,1)$, $\{\mathbf{f}^n\}$ is a Cauchy sequence and converges to its cluster point which is denoted as $\mathbf{f}^{\infty}$. Besides, $\lim_{n\to\infty}\|\mathbf{f}^n\text{-}\mathbf{F}(\mathbf{f}^n)\|_2=\lim_{n\to\infty}\left\|\mathbf{T}(\mathbf{f}^{n+1})\right\|_2=0$ which means $\mathbf{f}^{\infty}$ is the solution of Eq. (36). This completes the proof.

Proposition 6. *If assumption 2 holds, the BB2 scheme is globally convergent to the solution of Eq. (36).*

Proof. It follows from Propositions 3 and 4 and assumption 2 that

$$\begin{aligned}\left\|\mathbf{T}(\mathbf{f}^{n+1})\right\|_2&\leq\left(1+{\alpha_{\mathrm{BB2}}^{n}}^2L^2\text{-}2\alpha_{\mathrm{BB2}}^{n}\eta\right)\|\mathbf{T}(\mathbf{f}^n)\|_2\leq\left(1+\frac{L^2}{\eta^2}\text{-}2\frac{\eta}{L}\right)\|\mathbf{T}(\mathbf{f}^n)\|_2\\&\leq\left(1+\lambda^2\text{-}2\frac{1}{\lambda}\right)\|\mathbf{T}(\mathbf{f}^n)\|_2=\left(1\text{-}\xi^{\mathrm{II}}\right)\|\mathbf{T}(\mathbf{f}^n)\|_2.\end{aligned}$$

We have $\left\|\mathbf{T}(\mathbf{f}^{n+1})\right\|_2=\left\|\frac{\mathbf{f}^{n+2}\text{-}\mathbf{f}^{n+1}}{\alpha_{\mathrm{BB2}}^{n+1}}\right\|_2\leq\left(1\text{-}\xi^{\mathrm{II}}\right)^n\left\|\mathbf{T}(\mathbf{f}^1)\right\|_2$, which indicates

$$\left\|\mathbf{f}^{n+2}\text{-}\mathbf{f}^{n+1}\right\|_2\leq{\alpha_{\mathrm{BB2}}^{n+1}}^2\left(1\text{-}\xi^{\mathrm{I}}\right)^n\left\|\mathbf{T}(\mathbf{f}^1)\right\|_2\leq\frac{1}{\eta^2}\left(1\text{-}\xi^{\mathrm{II}}\right)^n\left\|\mathbf{T}(\mathbf{f}^1)\right\|_2.$$

The rest of the proof is similar to Proposition 5 and it is omitted here.

We have provided the theoretical convergence proof of the BB method under mild assumptions. In the next section, we will show detailed convergence characteristics for the BB method solving various SUE models in practical applications.

5. Numerical Results

This section presents several experiments to examine the convergence characteristics of the BB step size compared to two existing prevailing step size schemes (i.e., Armijo and SRA) in a path-based traffic assignment algorithm for solving the SUE problem. The sensitivity of the BB step size is also analyzed with respect to different demand levels and dispersion parameter values. In addition, we compare the flow allocation results of the MNL SUE and CNL SUE models.

In order to draw a relatively general conclusion, we test the proposed algorithm on two well-known SUE models (the MNL and CNL SUE models) and two realistic road networks, the Winnipeg and Chicago Sketch networks (obtained from http://www.bgu.ac.il/~bargera/tntp/) whose maps are exhibited in Figure 6 and Figure 7, respectively. In both networks, the traffic zones are the origins and destinations of the O-D flows. To have a fair comparison of different step size determination schemes, fixed working path sets are used for the computational tests. The working path set of the Winnipeg network was generated by Bekhor et al. (2008), and the working path set of the Chicago Sketch network was generated by using the seSUE, an open-source software obtained from http://sesue.uurari.com/#secDownloads. More details about the test examples are reported in Table 1.

FIGURE 6 Map of the Winnipeg network

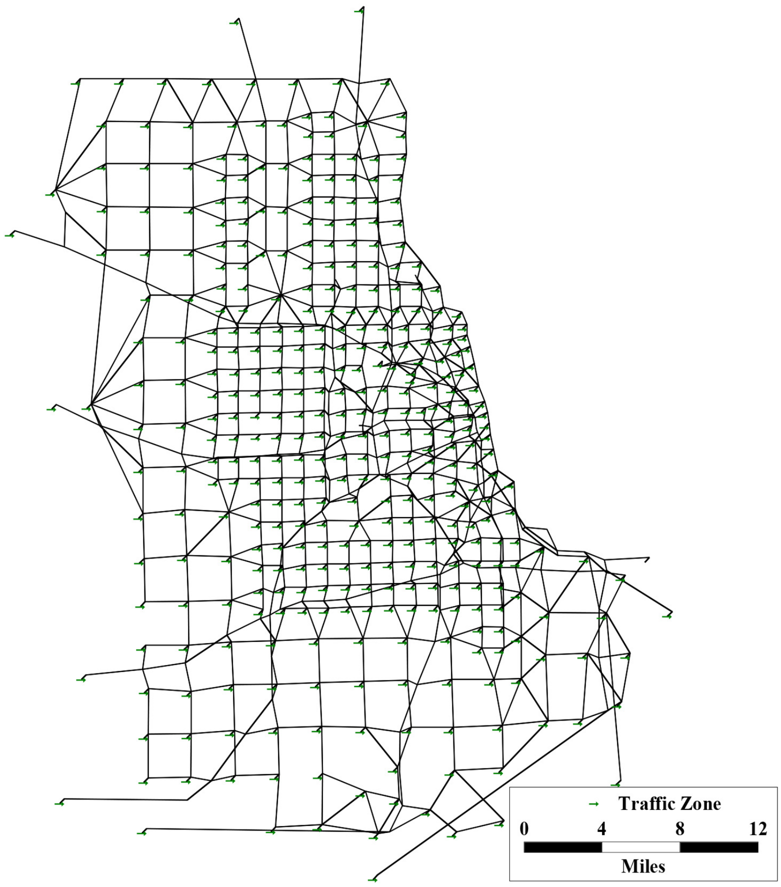


FIGURE 7 Map of the Chicago Sketch network

TABLE 1 Characteristics of test networks

Network	Zones #	Nodes #	Links #	O-D Pairs #	Paths #
Winnipeg	154	1,067	2,535	4,345	174,491
Chicago Sketch	387	933	2,950	93,135	836,346

The convergence criterion measure employed in this paper is based on the relative gap (RGAP), which is widely used to monitor the convergence process for solving the deterministic UE problem. For the MNL SUE model, it is calculated by

$$\text{RGAP(MNL SUE)}=\frac{\sum_r \sum_s \sum_k f_k^{rs} \cdot (g_k - g_{min}^{rs})}{\sum_r \sum_s \sum_k f_k^{rs} \cdot |g_k|}, \tag{46}$$

where g_k is the first-order derivative of the objective function with respect to f_k^{rs} and can be interpreted as the perceived path cost of path k; g_{min}^{rs} denotes the minimum perceived cost from origin r to destination s. For the CNL SUE model, it is calculated in the same manner as follows:

$$\text{RGAP(CNL SUE)}=\frac{\sum_r \sum_s \sum_m \sum_k f_{mk}^{rs} \cdot (g_k^m - g_{min}^{rs})}{\sum_r \sum_s \sum_m \sum_k f_{mk}^{rs} \cdot |g_k^m|}, \tag{47}$$

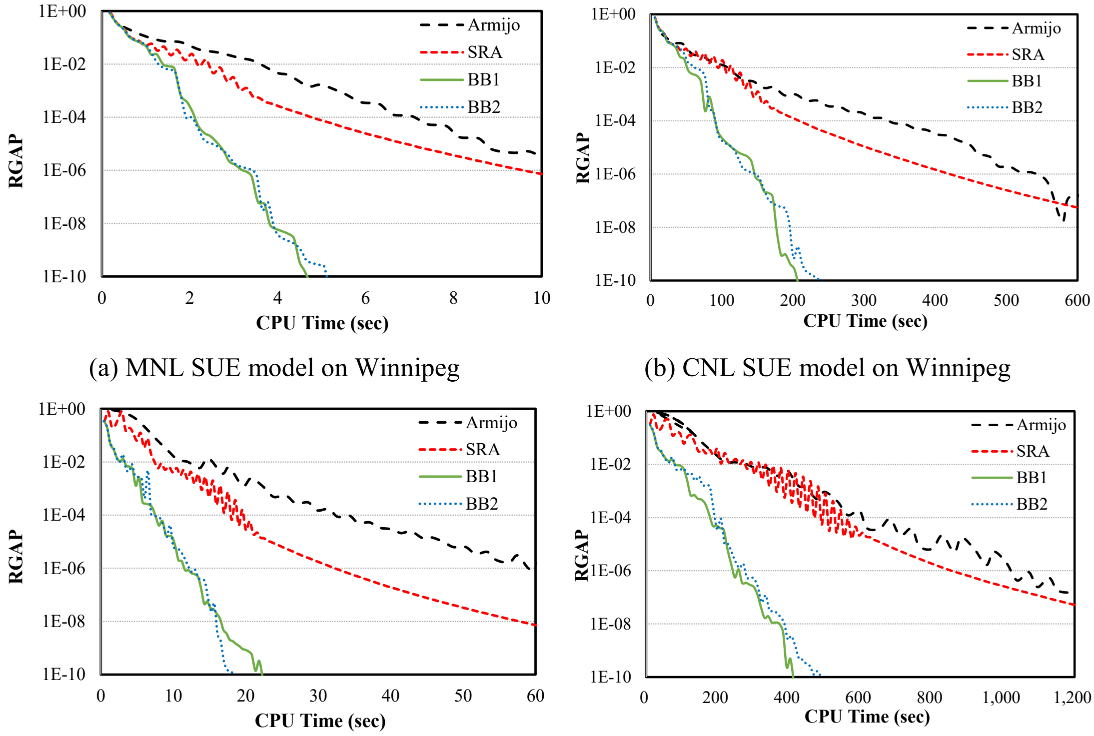
where g_k^m is the first-order partial derivative of the objective function with respect to f_{mk}^{rs} and g_{min}^{rs} is the minimum of all g_k^m for O-D pair rs. The difference between the above two formulas comes from the decision variables of the MNL SUE and CNL SUE models.

The nesting coefficient of the CNL SUE model, μ, is set to 0.5 for all experiments. In addition, we set the upper bound, γ_k, to 1 for the Armijo scheme, which can guarantee the feasibility of every approximate solution. Other parameters are set as: $\psi = 1.9$ and $\varphi = 0.1$ for the SRA scheme as suggested by Liu et al. (2009); for the Armijo scheme, $\beta = 0.6$ and $\sigma = 0.5$ which was adopted by Chen et al. (2014). The numerical experiments are conducted on Microsoft Windows 10 operating system with Intel Core i3-6100 CPU @ 3.70 GHz, 4GB RAM. The path-based traffic assignment algorithm embedded with different step size determination schemes is coded in Visual C#.

5.1. Convergence characteristics

Figure 8 shows the convergence curves of various step size schemes embedded in the path-based traffic assignment algorithm for solving the MNL and CNL SUE models on the two test networks. The horizontal and vertical axes represent the CPU time and RGAP, respectively. All algorithms are terminated when RGAP=1E-10 is achieved or the maximum CPU time is taken (i.e., 10 and 60 seconds for the MNL and CNL SUE models on the Winnipeg network; 600 and 1, 200 seconds for the MNL and CNL SUE models on the Chicago Sketch network). As shown in Figure 8, the performances of the two versions (BB1 and BB2) of the BB scheme are quite similar in all experiments. Note that both versions of the BB scheme can converge to the desirable precision (i.e., RGAP=1E-10) within the maximum CPU time, while the SRA and Armijo can only achieve RGAP=1E-6~1E-8.

Moreover, Table 2 exhibits more specific information about the comparative performance of different step size schemes. Due to the similar trend reported for both networks, here we only list the detailed comparative performance on the Winnipeg network to avoid duplication. Specifically, the BB scheme performs 1.75 and 2.25 times faster than the SRA and Armijo schemes for solving the MNL SUE model, and 1.89 and 2.75 times faster for the CNL SUE model. These results indicate that the proposed BB scheme can outperform the Armijo and SRA schemes in terms of computational efficiency with a wide margin for solving the two logit-based SUE problems.



(a) MNL SUE model on Winnipeg (b) CNL SUE model on Winnipeg

(c) MNL SUE model on Chicago Sketch (d) CNL SUE model on Chicago Sketch

FIGURE 8 RGAP versus CPU time for solving different SUE models on different networks.

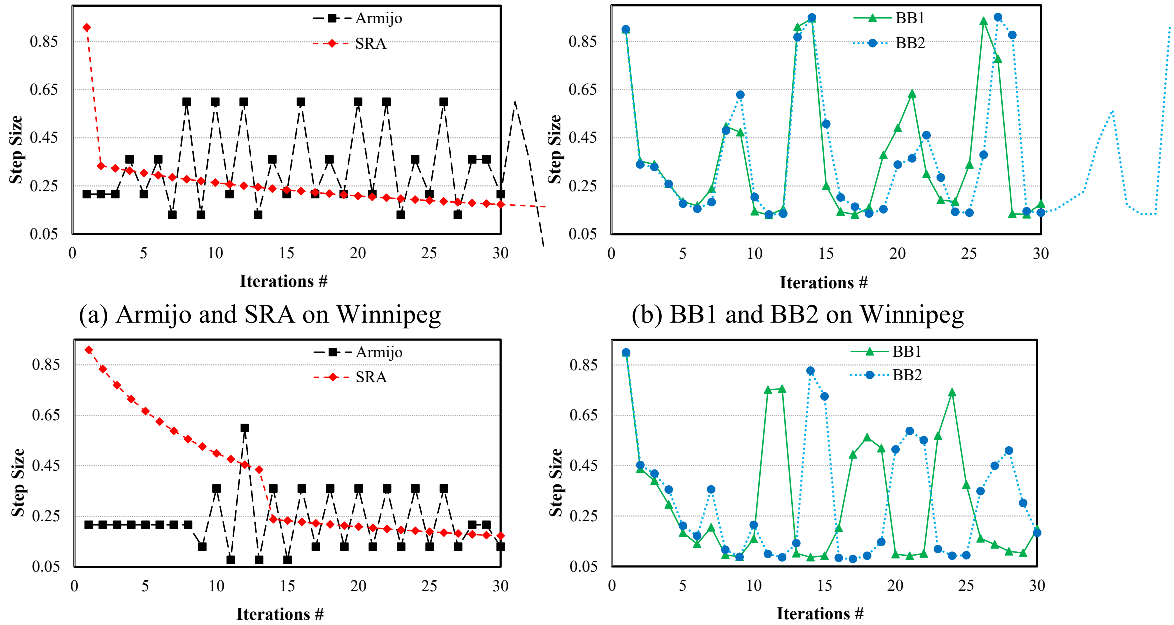
Table 2 also reveals some reasons behind the outstanding performance of the BB scheme. Note that the computational efforts can be roughly separated into two parts: the network loading step and the step size determination step. Since we use a working path set, the network loading time is similar for different step size schemes at each iteration. Hence, the differences in computational costs among different step size schemes mainly come from the step size determination step for each iteration. As shown in Table 2, the Armijo scheme takes fewer iterations than the SRA scheme during the whole convergence process, but the computational cost per iteration is much more expensive due to the need of evaluating the objective function value, which degrades the overall performance of the Armijo scheme. On the other hand, the BB scheme requires fewer iterations than that of the Armijo scheme. The average time per iteration of the BB scheme is 0.14 and 5.9 seconds for the MNL SUE and CNL SUE models, respectively. At each iteration, the computational cost of the BB scheme is quite close to that of the SRA scheme. Both only record the information from the last two consecutive iterations and do not require additional function evaluations of the objective function and/or its gradient. However, the BB scheme uses much fewer iterations than that of the SRA. In summary, the BB scheme needs fewer iterations to reach a desired level of convergence than the Armijo scheme, while it takes similar computational efforts per iteration compared to the SRA scheme. Table 2 also shows that the calculation of the BB scheme does not need to assume any parameter values compared to the Armijo and SRA schemes. The above analysis demonstrates that the BB scheme is quite efficient and practical.

TABLE 2 Comparative performance of different step size schemes to achieve RGAP = 1E-6 on the Winnipeg network

SUE Models	Step size Schemes	# of iterations	CPU time (sec)	# of function evaluations	Parameter settings
MNL SUE	Armijo	28	11.39	97	$\beta = 0.6,\ \sigma = 0.5$
	SRA	63	9.62	-	$\psi = 1.9,\ \varphi = 0.1$
	BB1	24	3.30	-	-
	BB2	26	3.52	-	-
CNL SUE	Armijo	37	552.15	140	$\beta = 0.6,\ \sigma = 0.5$
	SRA	68	425.59	-	$\psi = 1.9,\ \varphi = 0.1$
	BB1	25	147.53	-	-
	BB2	25	147.50	-	-

In order to further explore why fewer iterations are used for the BB scheme, we plot the step size trajectories of all step size determination schemes in the first 30 iterations for both networks in Figure 9. The horizontal and vertical axes separately represent the iteration number and the step size determined by each scheme. To improve the readability, we plot the step size trajectories of the BB scheme and the Armijo and SRA schemes separately. For brevity, we only present the trajectories of solving the MNL SUE model on both networks here. As shown in Figure 9, the SRA scheme produces a strictly decreasing step size sequence as required by the Blum's Theorem to guarantee the convergence (see Eq. (31)). One the other hand, the Armijo scheme produces a non-monotone (i.e., not strictly decreasing) step size sequence as long as it satisfies the adequate decrease condition given in Eq. (33). It is interesting to see that the step size sequence of the SRA scheme lies within an acceptable range of the sequence produced by the Armijo scheme, which suggests that using the solution information from the last two consecutive iterations without the need to evaluate the objective function and/or its derivative could provide a good approximate step size. Compared with the Armijo scheme, the BB scheme also produces a non-monotone step size sequence, but it takes a more aggressive (or larger) step size to achieve a better solution with a lower objective value as opposed to just satisfying the adequate decrease condition in the Armijo scheme. In summary, the BB step size scheme takes the advantages of the SRA and Armijo schemes (i.e., solution from last two consecutive iterations and non-monotone step size sequence) in determining a more flexible step size sequence to achieve a better solution with lower computational efforts.

An interesting observation is also revealed by examining the convergence characteristics in Figure 8 and step size sequence in Figure 9 together. It can be observed that the SRA scheme performs about 18% faster than the Armijo scheme in solving the MNL SUE model on the Winnipeg network (Figure 8a), but around 83% faster on the Chicago Sketch network (Figure 8c). Now we focus on the differences of the corresponding step size trajectories (Figures 9a and 9c) to find out the reason. We can see that the SRA step size trajectories in both networks experience a sharp decrease, which indicates that ψ in Eq. (30) is active at the 2^{nd} and 14^{th} iteration for the Winnipeg and Chicago Sketch networks, respectively. In the Chicago Sketch network, more aggressive movements are made in the early iterations, which result in a better overall performance. In fact, the monotonically decreasing property of the SRA scheme will eventually force the step size to be very small when the step size decreases sharply (ψ is active) in early iterations. Consequently, the convergence speed will also be slow. In addition, we can see in Figure 8c that the proposed BB scheme is much more efficient than the SRA scheme in early iterations. Specifically, the BB scheme only takes 5 iterations and 2.1 seconds to achieve the precision which the SRA scheme takes 13 iterations and 6.4 seconds to achieve. This implies that the BB schemes can greatly outperform the SRA scheme, even when the SRA scheme is in the best condition (φ is active all the time) due to the need to ensure the strictly decreasing condition required by the Blum's Theorem to ensure convergence.



(a) Armijo and SRA on Winnipeg (b) BB1 and BB2 on Winnipeg

(c) Armijo and SRA on Chicago Sketch (d) BB1 and BB2 on Chicago Sketch

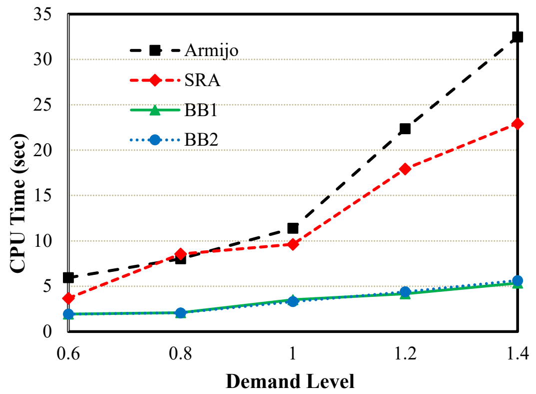
FIGURE 9 Step size trajectories of all step size determination schemes on different networks.

5.2. Sensitivity analysis

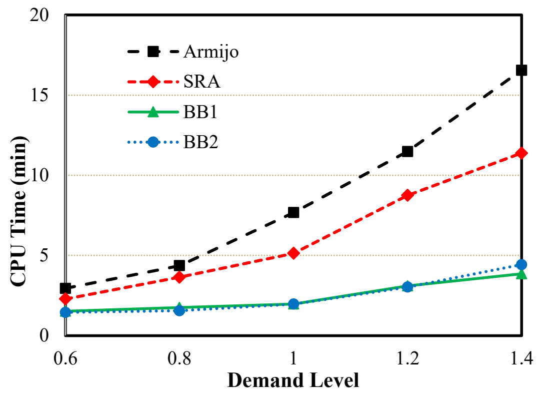
The above analysis indicates that the BB scheme outperforms the prevailing Armijo and SRA schemes in terms of computational efficiency. In the following, we examine the sensitivity of the BB scheme with respect to different demand levels and dispersion parameter values for both MNL and CNL SUE models on the Winnipeg network. For completeness, we also include the Armijo and SRA schemes in the comparison. The convergence criterion is set to RGAP=1E-6 for the MNL SUE model. Since it is much more computationally expensive to solve the CNL SUE model, the convergence criterion is set to a higher value, i.e. RGAP=1E-5.

The experiment results are shown in Figure 10. First, we investigate the effects of different demand levels on computational performance by varying the demand level from 0.6 to 1.4 of the base demand. As shown in Figures 9a and 9b, the CPU time generally increases as the demand level increases for all step size determination schemes in solving both logit-based SUE models. However, the CPU time of the Armijo and SRA schemes grows at a faster rate compared to that of the BB scheme. Specifically, the Armijo and SRA schemes respectively take 4.47 and 5.26 times of computational efforts to solve the MNL SUE model for the demand level from 0.6 to 1.4, while the BB scheme just takes 1.8 times for the same demand range. It clearly reveals that the Armijo and SRA schemes are quite sensitive to the congestion level. Next, we investigate the effects of different dispersion parameter values on the computational performance by varying the dispersion parameter value from 0.1 to 1.6 with an interval of 0.5.

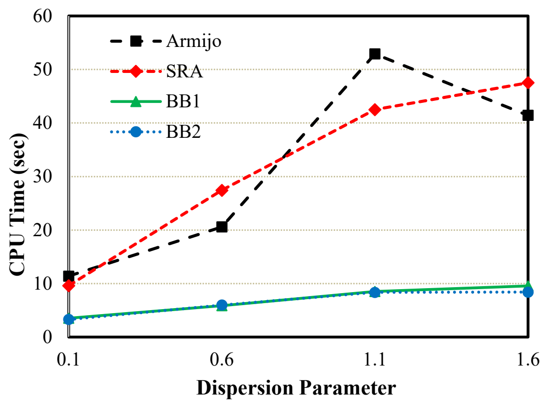
Figures 9c and 9d also show the CPU time generally increases as the dispersion parameter value increases (i.e., flow allocations are more concentrated on the minimum cost paths). The trend for the BB scheme is quite stable and shows only a minor increase in CPU time as the dispersion parameter value increases. However, the trend for the Armijo and SRA schemes is less obvious and requires significantly more CPU times than the BB scheme for both logit-based SUE models.



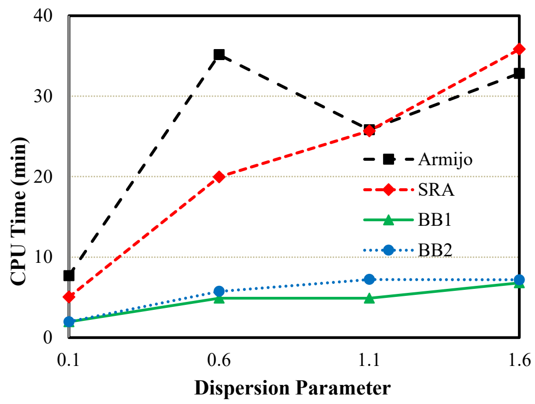
(a) Effect of demand level on the MNL SUE model



(b) Effect of demand level on the CNL SUE model



(c) Effect of dispersion parameter on the MNL SUE model



(d) Effect of dispersion parameter on the CNL SUE model

FIGURE 10 Computational effort under different demand levels and dispersion parameter values on the Winnipeg network.

5.3. Flow allocation comparison

This section compares the differences in path and link flows between the MNL SUE and CNL SUE models on the Winnipeg network. These two models share a common dispersion parameter θ which reflects an aggregate measure of driver's perception of path cost (Sheffi, 1985). For a fair comparison, θ is set to 0.1 in both models. In addition, the CNL model has another parameter μ which accounts for the similarity among paths. The CNL model collapses to the MNL model if μ is set to 1 and reduces to a deterministic choice probability pattern when μ tends to 0. In this comparison, μ is set to 0.5. Recall that the CNL SUE model can overcome the path overlapping issue which the MNL SUE model suffers from. Thus, different flow patterns can be expected on realistic networks.

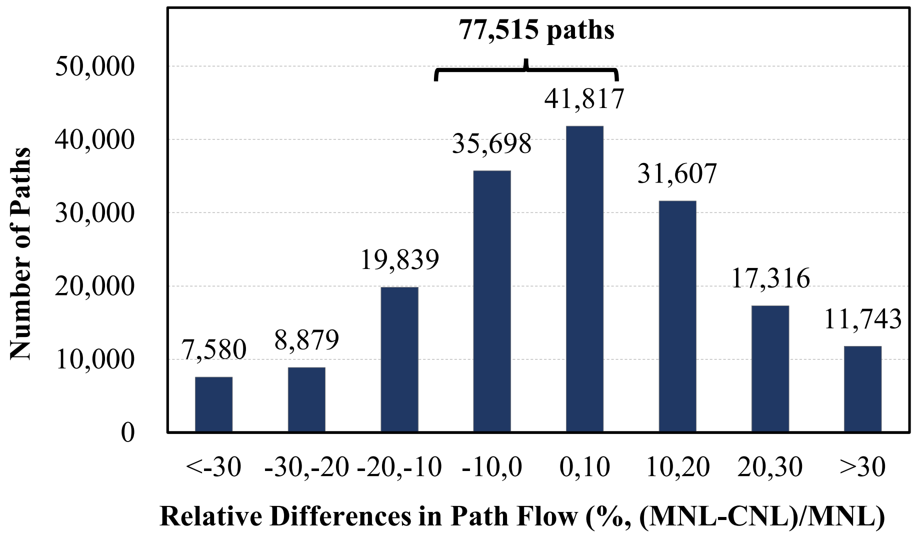


FIGURE 11 Differences in path flow between the MNL SUE and CNL SUE models

We first examine the differences in path flow shown in Figure 11. The horizontal axis represents the percentage of relative differences in path flow (i.e., $(f^{MNL}- f^{CNL})/ f^{MNL}$) and the vertical axis represents the corresponding numbers of paths. It can be seen that the around 55.6% of the paths have a relative difference larger than 10% (or 96,976 paths), while 44.4% of the paths, have a relative difference smaller than 10%. Furthermore, a large number of paths (specifically, 45,518 paths) have a relative difference larger than 20%. This shows that the two logit-based SUE models allocate significantly different path flow patterns, because the CNL SUE model can account for the path overlapping issue while the MNL SUE cannot.

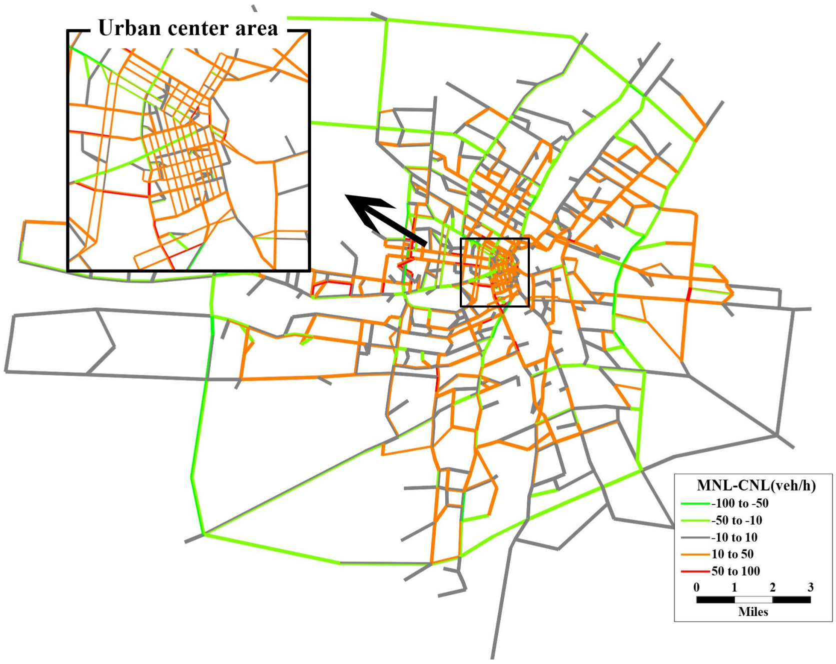


FIGURE 12 Differences in link flows between MNL SUE and CNL SUE models

On the other hand, we examine the differences in link flow pattern to see how these differences are distributed on the network and study the effects of the path overlapping issue on the link flow pattern. Figure 12 shows the differences in link flow between the MNL SUE and CNL SUE models (i.e., $f^{MNL}- f^{CNL}$) with a Geographical Information System (GIS) map. We use different colors for each link to highlight the magnitude of the flow difference. Compared with the MNL SUE model, the CNL SUE model allocates less flows to the links in the urban center area. It is because that the links in center area are shared by more paths and thus the similarity is higher among these paths. As the CNL SUE model accounts for the effect of route overlapping problem, less flow will be allocated to the center area. The situation is opposite in the outer area.

Overall, the flow patterns are significantly different for the MNL SUE and CNL SUE models. It is important to handle the path overlapping issue and this problem gets more severe in the urban center area than the outer area.

6. Extensions

In this section, we introduce some extended applications of the proposed algorithm following the discussion in **Remark 1**.

6.1. Link-based SUE models

We have proposed an efficient path-based algorithm incorporated with the BB step size scheme for solving the SUE problem. In this subsection, we take the link-based MNL SUE model as an example to show how to adapt the BB step size scheme into the link-based solution algorithm. Compared to path-based algorithms, link-based algorithms do not need an explicit path set and thus take less storage memory, but are less flexible (Maher, 1998; Kitthamkesorn and Chen, 2014). The link-based formulation of the MNL SUE model is shown as the following unconstrained program.

$$\min z = -\sum_{rs} q_{rs} S_{rs}[c^{rs}(x)] + \sum_{a\in A} x_a t_a(x_a) - \sum_{a\in A} \int_0^{x_a} t_a(\omega)\,\mathrm{d}\omega \tag{48}$$

where $S_{rs}[c^{rs}(x)]=E[\min_{k\in \mathrm{K}_{rs}}\{c_k^{rs}\}]$ and $\frac{\partial S_{rs}[c^{rs}(x)]}{c_k^{rs}}=P_k^{rs}$. Note that the decision variable is link flow x_a in this formulation. The meanings of other notations are the same as in Section 3.

The first-order necessary condition of optimality of the unconstrained program in Eq. (48) is

$$\nabla_{\mathbf{x}} z = \nabla_{\mathbf{x}}\mathbf{t}\cdot[\mathbf{x} - \Delta\cdot(\mathbf{\Lambda}^{\mathbf{T}}\cdot\mathbf{q}\circ\mathbf{P})] = \mathbf{0} \tag{49}$$

where $\mathbf{x}, \mathbf{t}, \mathbf{q}$, and $\mathbf{P}$ represent the vectors of link flow, link cost, O-D demand, and path choice probability, respectively; '$\circ$' is the Hadamard product; Δ and $\mathbf{\Lambda}$ are incidence matrices for describing the relationship between links and paths and the relationship between O-D pairs and paths, respectively. Since $\nabla_{\mathbf{x}}\mathbf{t}$ is a diagonal positive definite matrix here, we only need to solve the following system of equations

$$\mathbf{x} - \Delta\cdot(\mathbf{\Lambda}^{\mathbf{T}}\cdot\mathbf{q}\circ\mathbf{P}) = \mathbf{0}. \tag{50}$$

Following the approach developed in section 4, the BB scheme can be easily adapted to solve Eq. (50). The solution procedure can be concluded as follows.

Algorithm. Link-based SUE solution algorithm

Step 0: Initialization

Set n=0, update cost, and perform the SNL to obtain initial link flows $x_a^0,\ \forall a$

Step 1: Direction finding

Set n=n+1, update cost, perform the SNL to obtain auxiliary link flows $\tilde{x}_a^n,\ \forall a$, and define the direction $\tilde{x}_a^n - x_a^n,\ \forall a$.

Step 2: Step size determination

Determine the BB step size α_{BB}^n and obtain new link flows: $x_a^{n+1} = x_a^n + \alpha_{\mathrm{BB}}^n(\tilde{x}_a^n - x_a^n),\ \forall a$.

Step 3: Convergence test

If the stopping criterion is satisfied, terminate; otherwise, go to step 1.

Due to the monotonicity of $-\Delta\cdot(\mathbf{\Lambda}^{\mathbf{T}}\cdot\mathbf{q}\circ\mathbf{P})$ (Sheffi, 1985), the properties we have established in Section 4 also holds in this situation. In fact, Eq. (50) is equivalent to Eq. (28). The only difference is that link flows are decision variables in Eq. (50) while path flows in Eq. (28). Further, we can compute $-\Delta\cdot(\mathbf{\Lambda}^{\mathbf{T}}\cdot\mathbf{q}\circ\mathbf{P})$ by performing a link-based stochastic network loading procedure. Readers can refer to Dial (1971) and Bell (1995) for more details about the logit-based stochastic loading schemes.

6.2. Hierarchical travel choice models

Hierarchical (or combined) travel choice models aim to overcome the inherent drawbacks of the traditional four-step model, which considers trip generation, trip distribution, modal split and traffic assignment sequentially, in travel demand forecasting. In the four-step model, the outputs of one step serve as the inputs of the next step, which leads to inconsistency in travel times and congestion effects among different steps (Yao et al., 2014; Zhou et al., 2009). In contrast, the hierarchical travel choice model assumes that travelers make different travel choices simultaneously in a combined model. Under the random utility theory framework in microeconomics, an integrated model can be conveniently constructed to account for these steps (not necessarily all steps) simultaneously.

Zhou et al. (2009) presented the structure of the hierarchical travel choice model (see Figure 1 in their paper), the demand of each step is the multiplication of the demand in the last step and the corresponding conditional probability. Hence, the path flow pattern can be concluded as follows

$$T_{rsek} = N_r P_{t|r} P_{s|r} P_{e|rs} P_{k|rse},\ \forall r, s, e, k. \tag{51}$$

where T_{rsek} is the travel demand taking path k from origin r to destination s on mode e; N_r is the potential number of travelers in origin r; $P_{t|r}$ is the probability of making a trip given N_r; $P_{s|r}$ is the probability of choosing destination s from origin r; $P_{e|rs}$ is the probability of choosing mode e between O-D (r, s); $P_{k|rse}$ is the probability of choosing path k of O-D (r, s) on mode e. Note that the right-hand side of Eq. (51) is a function with respect to path flows T_{rsek}, and thus Eq. (51) is actually a fixed-point problem. Further, as the conditional probability is monotone with respect to the utility (Sheffi, 1985; Zhou et al., 2009), the proposed algorithm in Section 4 can be adapted to solve this model.

6.3. Dynamic stochastic user equilibrium (DSUE) models

The aforementioned traffic assignment models all belong to the category of static traffic assignment (STA). Compared to STA models, dynamic traffic assignment (DTA) models are able to deal with time-varying flows and capture traffic flow dynamics more realistically (Jayakrishnan et al., 1995; Peeta and Ziliaskopoulos, 2001; Huang and Lam, 2002). Among the DTA models, the DSUE model relaxes the assumption of traveler's perfect perception about network conditions and are considered to be more realistic than the deterministic counterparts (Han, 2003; Lim and Heydecker, 2005; Long et al., 2015). The DSUE equilibrium state can be stated as, no traveler can improve the perceived travel cost by unilaterally change the departure time and path combination (Long et al., 2015).

The combination of departure time and path choices can be described using a hierarchical structure discussed in Section 6.2 (i.e., combine the departure time choice and the path choice in a hierarchical structure). Hence, the combined departure time and path choice model is presented as follows:

The time period of interest is discretized into a finite set of time intervals, $U = \{u: u = 1, 2, \ldots, |U|\}$ where $|U|$ is the maximum number of intervals. Without loss of generality, we here take the MNL model as an example. Given an O-D pair, the probability of a traveler departing at interval u is obtained by

$$P(u)=\frac{\exp(-\theta_t \bar{c}(u))}{\sum_{v\in U}\exp(-\theta_t \bar{c}(v))} \tag{52}$$

where $\bar{c}(u)$ denotes the expected perceived cost for travelers departing at interval u, and θ_t is a dispersion parameter associated with the departure time choice. On the other hand, given an O-D pair and interval u, the probability of a traveler choosing path k can be obtained by

$$P_k(u)=\frac{\exp(-\theta c_k(u))}{\sum_{l\in K}\exp(-\theta c_l(u))} \tag{53}$$

where $c_k(u)$ denotes the actual travel cost for travelers choosing path k departing at interval u; K is the feasible path set; θ is a dispersion parameter associated with the path choice, and $\theta \geq \theta_t$ (Lim and Heydecker, 2005). For the MNL model, traveler's expected perceived cost is given as follows (Lim and Heydecker, 2005; Long et al., 2015)

$$\bar{c}(u) = E[\min_{k\in K}\{c_k(u)\}] = -\frac{1}{\theta_r} ln \sum_{l\in K}\exp(-\theta_r c_l(u)). \tag{54}$$

Combine Eqs. (52)-(54), we obtain the combined departure time and path choice model as follows

$$f_k(u) = qP(u)\,P_k(u) = q\frac{\exp(\frac{\theta_t}{\theta} ln \sum_{l\in K}\exp(-\theta c_l(u)))}{\sum_{v\in U}\exp(\frac{\theta_t}{\theta} ln \sum_{l\in K}\exp(-\theta c_l(u)))}\frac{\exp(-\theta c_k(u))}{\sum_{l\in K}\exp(-\theta c_l(u))} \tag{55}$$

where $f_k(u)$ denotes the number of trips on path k departing at interval u, and q is the demand. Note that Eq. (55) is a fixed-point problem and our proposed algorithm can be applied. However, the monotonicity of the path cost with respect to path flow may not be established for various dynamic network loading (DNL) models. The DNL models depict how traffic flow propagates in a transportation network and are performed to obtain the travel times on all paths with a given path inflow profile. Given the travel time pattern, the combined departure time and path choice model guides how to adjust path inflows. Readers can refer to Peeta and Ziliaskopoulos (2001), Nie and Zhang (2005), Mun (2007), and Bliemer et al. (2017) for more details about the DNL models. Consequently, Proposition 1 in Section 4 may not always hold for the DSUE model.

Hence, before applying the proposed algorithm, one should check whether the monotonicity of the path cost holds. If it holds, the proposed algorithm can be directly adopted to solve the DSUE model; otherwise, the BB method may fail and can only be used as a heuristic method. In this case, to ensure the positivity of path flows, one can design some practical approaches like setting lower and upper boundaries for the BB step size sequence.

7. Concluding Remarks

The Barzilai-Borwein (BB) step size is a special step size determination scheme incorporated into the gradient method by using the solution information from the last two consecutive iterations to enhance its computational efficiency without the need to explicitly compute the second-order derivative, i.e., the Hessian. The BB step size has been found to significantly outperform the gradient method at nearly no extra cost. In this paper, we explored the BB step size to develop an efficient and robust path-based traffic assignment algorithm for solving the stochastic user equilibrium (SUE) problem, which often involves a large dimension of decision variables (i.e., path flows) and complex objective function (i.e., link integral and path entropy terms).

To examine the computational efficiency and robustness of the BB scheme implemented in a path-based traffic assignment problem, we solved two well-known SUE models (i.e., the multinomial logit (MNL) and cross-nested logit (CNL) SUE models) on two real transportation networks (i.e., the Winnipeg and Chicago Sketch networks) and compared the performance with two popular step size schemes (i.e., the Armijo scheme and the self-regulated averaging (SRA) scheme). The numerical results indicated that the two versions of the BB scheme significantly outperformed the Armijo and SRA schemes in terms of both computational efficiency and robustness with respect to congestion and dispersion levels. In addition, the BB scheme does not require any parameter setting, which is required in the Armijo and SRA schemes. All these good features make the BB step size rather attractive for implementing path-based traffic assignment algorithms. Note that the above conclusions were based on solving the MNL SUE and CNL SUE models on the Winnipeg and Chicago Sketch networks. In the future, more tests should be conducted to further examine the performance of the BB step size scheme on various network configurations, other SUE models (e.g., the weibit-based SUE models of Kitthamkesorn & Chen, 2013; Kitthamkesorn & Chen, 2014; Xu et al., 2015), combined travel demand models with multi-dimensional travel choices (e.g., Xu et al., 2008; Zhou et al., 2009; Yao et al., 2014; Kitthamkesorn et al., 2016; Ryu et al., 2017), and DSUE models (e.g., Han, 2003; Lim and Heydecker, 2005; Long et al., 2015).

ACKNOWLEDGMENTS

This research is supported by the Natural Science Foundation of China (No. 71801079), the Research Grants Council of the Hong Kong Special Administrative Region (No. 115212217), the Research Committee of the Hong Kong Polytechnic University (No. 1-ZVJV), and the Research Institute for Sustainable Urban Development at the Hong Kong Polytechnic University (1-BBWF).

REFERENCES

Armijo, L. (1966). Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific Journal of Mathematics*, *16*(1), 1-3.

Bar-Gera, H. (2002). Origin-based algorithm for the traffic assignment problem. *Transportation Science*, *36*(4), 398-417.

Barzilai, J., & Borwein. J. (1988). Two-point step size gradient methods. *IMA Journal of Numerical Analysis*, *8*(1), 141-148.

Bekhor S., Reznikova L., & Toledo T. (2007). Application of cross-nested logit path choice model in stochastic user equilibrium traffic assignment. Transportation Research Record 2003, 41-49.

Bekhor, S., & Toledo, T. (2005). Investigating path-based solution algorithms to the stochastic user equilibrium problem. *Transportation Research Part B*, *39*(3), 279-295.

Bekhor, S., Toledo, T., & Reznikova, L. (2008). A path-based algorithm for the cross-nested logit stochastic user equilibrium traffic assignment. *Computer-Aided Civil and Infrastructure Engineering*, *24*(1), 15-25.

Bliemer, M. C. J., Raadsen, M. P. H., Brederode, L. J. N., Bell, M. G. H., Wismans, L. J. J., & Smith, M. J. (2017). Genetics of traffic assignment models for strategic transport planning. *Transport Reviews*, *37*(1), 56–78.

Bell, M. G. H. (1995). Alternatives to Dial's logit assignment algorithm. *Transportation Research Part B*, *29*(4), 287–295.

Ben-Akiva, M. & Bierlaire, M. (1999). *Discrete choice methods and their applications to short term travel decisions*. Handbook of Transportation Science. Boston, MA: Springer.

Bertsekas, D. (1976). On the Goldstein-Levitin-Polyak gradient projection method. *IEEE Transactions on Automatic Control*, *21*(2), 174-184.

Blum, J. (1954). Multidimensional stochastic approximation methods. *The Annals of Mathematical Statistics*, *4*, 737-744.

Cantarella, G., Carteni, A., & Luca, S. (2015). Stochastic equilibrium assignment with variable demand: Theoretical and implementation issues. *European Journal of Operational Research, 241*, 330-347.

Cascetta, E., Nuzzolo, A., Russo, F., & Vitetta, A. (1996). A modified logit path choice model overcoming path overlapping problems: specification and some calibration results for interurban networks. *Proceedings of 13th International Symposium on Transportation and Traffic Theory, Lyon*, 697–711.

Castillo, E., Menendez, J., Jimenez, P., & Rivas, A. (2008). Closed form expression for choice probabilities in the Weibull case. *Transportation Research Part B, 42*(4), 373-380.

Chen, A., Lo, H., & Yang, H. (2001). A self-adaptive projection and contraction algorithm for the traffic assignment problem with path-specific costs. *European Journal of Operational Research, 135*(1), 27–41.

Chen, A., Oh, J.-S., Park, D., & Recker, W. (2010). Solving the bicriteria traffic equilibrium problem with variable demand and nonlinear path costs. *Applied Mathematics and Computation, 217* (7), 3020–3031.

Chen, A., Ryu, S., Xu, X., & Choi, K. (2014). Computation and application of the paired combinatorial logit stochastic user equilibrium problem. *Computers & Operations Research*, *43*, 68-77.

Chen, A., Xu, X., Ryu, S., & Zhou, Z. (2013). A self-adaptive Armijo step size strategy with application to traffic assignment models and algorithms. *Transportmetrica*, *9*(8), 695-712.

Chen, A., Zhou, Z., & Ryu, S. (2011). Modeling physical and environmental side constraints in traffic equilibrium problem. *International Journal of Sustainable Transportation*, *5*(3), 172–197.

Chen, A., Zhou, Z., & Xu, X. (2012). A self-adaptive gradient projection algorithm for the nonadditive traffic equilibrium problem. *Computer & Operation Research*, 39(2), 127-138.

Chen, M., & Alfa, A. (1991). Algorithms for solving Fisk's stochastic traffic assignment model. *Transportation Research Part B, 25* (6), 405-412.

Chiou, S. (2007). An efficient computation algorithm for area traffic control problem with link capacity expansions. *Applied Mathematics and Computation, 188*, 1094–1102.

Cruz, W., & Raydan, M. (2003). Nonmonotone spectral methods for large-scale nonlinear systems. *Optimization*

Methods and Software. *18*(5), 583-599.

Daganzo, C., & Sheffi, Y. (1977). On stochastic models of traffic assignment. *Transportation Science*, *11*, 253-274.

Damberg, O., Lundgren, J. T., & Patriksson, M. (1996). An algorithm for the stochastic user equilibrium problem. *Transportation Research Part B: Methodological*, *30*(2), 115–131.

Dial, R. (1971). A probabilistic multipath traffic assignment model which obviates path enumeration. *Transportation Research, 5*(2), 83-111.

Fisk, C. (1980). Some developments in equilibrium traffic assignment. *Transportation Research Part B*, *14B*, 243-255.

Florian, M., Constantin, I., & Florian, D. (2009). A new look at projected gradient method for equilibrium assignment. *Transportation Research Record, 2090*, 10–16.

Gibb, J. (2016). Solving travel demand model equilibrium with Barzilai-Browein step sizes. Presented at 95th Annual Meeting of the Transportation Research Board, Washington, D.C.

Han, S. (2003). Dynamic traffic modelling and dynamic stochastic user equilibrium assignment for general road networks. *Transportation Research Part B: Methodological*, *37*(3), 225–249.

He, H., Han, D., & Li, Z. (2012). Some projection methods with the BB step sizes for variational inequalities. *Journal of Computational and Applied Mathematics*, *236*(9), 2590–2604.

Huang, H. J., & Lam, W. H. K. (2002). Modeling and solving the dynamic user equilibrium path and departure time choice problem in network with queues. *Transportation Research Part B: Methodological*, *36*(3), 253–273.

Huang, H., & Li, Z. (2007). A multiclass, multicriteria logit-based traffic equilibrium assignment model under ATIS. *European Journal of Operational Research*, *176*(3), 1464–1477.

Huang, H., Wang, S., & Bell, M. (2001). A bi-level formulation and quasi-Newton algorithm for stochastic equilibrium network design problem with elastic demand. *Journal of Systems Science and Complexity*, 14(1), 40-53.

Jayakrishnan, R., Tsai, W., & Prashker, J. (1994). A faster path-based algorithm for traffic assignment. *Transportation Research Record, 1554*, 75-83.

Jayakrishnan, R., Tsai, W. K., & Chen, A. (1995). A dynamic traffic assignment model with traffic-flow relationships. *Transportation Research Part C: Emerging Technologies*, *3*(1), 51–72.

Kitthamkesorn, S., & Chen, A. (2013). A path-size weibit stochastic user equilibrium model. *Transportation Research Part B*, *57*, 378-397.

Kitthamkesorn, S., & Chen, A. (2014). Unconstrained weibit stochastic user equilibrium model with extensions. *Transportation Research Part B*, *59*, 1-21.

Kitthamkesorn, S., Chen, A., & Xu, X. (2015). Elastic demand with weibit stochastic user equilibrium flows and application in a motorized and non-motorized network. *Transportmetrica A: Transport science*, *11*(2), 158-185.

Kitthamkesorn, S., Chen, A., Xu, X., & Ryu, S. (2016). Modeling mode and path similarities in network equilibrium problem with go-green modes. *Networks and Spatial Economics*, *16*(1), 33-60.

Kitthamkesorn, S., & Chen, A. (2017). Alternate weibit-based model for assessing green transport systems with combine mode and path travel choices. *Transportation Research Part B*, *103*, 291-310.

Leurent F. (1993). Cost versus time equilibrium over a network. *European Journal of Operations Research，71*, 205–221.

Lim, Y., & Heydecker, B. (2005). Dynamic departure time and stochastic user equilibrium assignment. *Transportation Research Part B: Methodological*, *39*(2), 97–118.

Liu, H., He, X., & He, B. (2009). Method of successive weighted averages (MSWA) and self-regulated averaging schemes for solving stochastic user equilibrium problem. *Network & Spatial Economics*, *9*, 485-503.

Liu, J., & Feng, Y. (2019). A derivative-free iterative method for nonlinear monotone equations with convex constraints. *Numerical Algorithms*, *82*(1), 245-262.

Long, J., Szeto, W. Y., Shi, Q., Gao, Z., & Huang, H. J. (2015). A nonlinear equation system approach to the dynamic stochastic user equilibrium simultaneous path and departure time choice problem. *Transportmetrica A: Transport Science*, *11*(5), 388–419.

Maher, M. (1998). Algorithm for logit-based stochastic user equilibrium assignment. *Transportation Research Part B, 32*(8), 539-549.

Mun, J. S. (2007). Traffic performance models for dynamic traffic assignment: An assessment of existing models. *Transport Reviews*, *27*(2), 231–249.

Nagurney, A., & Zhang, D. (1996). *Projected dynamical systems and variational inequalities with applications*. Boston: Kluwer.

Nie, X., & Zhang, H. M. (2005). A comparative study of some macroscopic link models used in dynamic traffic assignment. *Networks and Spatial Economics*, *5*(1), 89–115.

Nocedal, J., & Stephen J. (1999). *Numerical optimization*. New York: Springer.

Peeta, S., & Ziliaskopoulos, A. (2001). Foundations of Dynamic Traffic Assignment: The Past, the Present and the Future. *Networks and Spatial Economics*, *1*, 233–265.

Perederieieva, O., Raith, A., & Schmidt, M. (2018). Non-additive shortest path in the context of traffic assignment. *European Journal of Operational Research, 268*, 325-338.

Polyak, B. (1990). New method of stochastic approximation type. *Automation and Remote Control, 51*(7), 937-946.

Prashker, J., & Bekhor, S. (1998). Investigation of stochastic network loading procedures. *Transportation Research Record*, *1645*, 94-102.

Prashker, J., & Bekhor, S. (1999). Stochastic user-equilibrium formulations for extended-Logit assignment models. *Transportation Research Record*, *1676*, 145-152.

Prashker, J., & Bekhor, S. (2004). Path choice models used in the stochastic user equilibrium problem: A review. *Transport Reviews, 24*(4), 437–463.

Pravinvongvuth, S., & Chen, A. (2005). Adaptation of the paired combinatorial logit model to the path choice problem. *Transportmetrica*, *1*(3), 223-240.

Raydan, M. (1993). On the Barzilai and Borwein choice of step length for the gradient method. *IMA Journal of Numerical Analysis, 13*(3), 321-326.

Rosen, J. (1960). The gradient projection method for nonlinear programming. Part I. Linear Constraints. *Journal of the Society for Industrial and Applied Mathematics, 8*(1), 181–217.

Ryu, S., Chen, A., & Choi, K. (2014). A modified gradient projection algorithm for solving the elastic demand traffic equilibrium problem. *Computers and Operations Research*, 47, 61-71.

Ryu, S., Chen, A., & Choi, K. (2017). Solving the combined modal split and traffic assignment problem with two types of transit impedance function. *European Journal of Operational Research, 257*(3), 870–880.

Sheffi, Y., & Powell, W. (1981). A comparison of stochastic and deterministic traffic assignment over congested networks. *Transportation Research Part B, 15B*, 53-64.

Sheffi, Y. (1985). *Urban transportation networks: Equilibrium Analysis with Mathematical Programming Methods*. N.J: Prentice-Hall, Englewood Cliffs.

Shivam, S., Buckley, I., Wardi, Y., Seatzu, C., & Egerstedt. M. (2019). Tracking control by the Newton-Raphson flow: applications to autonomous vehicles. *Proceedings of 18th European Control Conference, Napoli*, 1562–1567.

Szeto, W.Y., Jiang, Y., Wang, D.Z.W., & Sumalee, A. (2015). A sustainable road network design problem with land use transportation interaction over time. *Networks and Spatial Economics* 15(3), 791-822.

Tan, H., Du, M., Jiang, X., & Chu, Z. (2019). The combined distribution and assignment model: A new solution algorithm and its applications in travel demand forecasting for modern urban transportation. *Sustainability, 11*(7).

Vovsha, P. (1997). The Cross-Nested logit model: application to mode choice in the Tel Aviv metropolitan area. Presented at 76th Annual Meeting of the Transportation Research Board, Washington, D.C.

Vovsha, P., Bekhor, S. (1998). Link-Nested logit model of path choice overcoming path overlapping problem. *Transportation Research Record*, *1645*, 133-142.

Wardrop, J. G. (1952). Some theoretical aspects of road traffic research. *Proceedings of Institution of Civil Engineers*, *2*(1), 325–378.

Xie, J., Nie, Y., & Liu, X. (2018). A greedy path-based algorithm for traffic assignment. *Transportation Research Record*, *2672*, 36-44.

Xu, M., Chen, A., & Gao, Z. (2008). An improved origin-based algorithm for solving the combined distribution and assignment problem. *European Journal of Operational Research, 188*(2), 354–369.

Xu, M., Zhou, B., & He, J. (2019). Improving truncated Newton method for the logit-based stochastic user equilibrium problem. *Mathematical Problems in Engineering*, Article ID 7313808, 15 pages.

Xu, X., & Chen, A. (2013). C-logit stochastic user equilibrium problem with elastic demand. *Transportation Planning and Technology*, *36*(5), 463-478.

Xu, X., Chen, A., Kitthamkesorn, S., Yang, H., & Lo. H.K. (2015). Modeling absolute and relative cost differences in stochastic user equilibrium problem. *Transportation Research Part B*, *87*, 686-703.

Xu, X., Chen, A., Zhou, Z., & Behkor, S. (2012). Path-based algorithms for solving C-logit stochastic user equilibrium assignment problem. *Transportation Research Record* 2279, 21-30.

Yang, C., & Chen, A. (2009). Sensitivity analysis of the combined travel demand model with applications. *European Journal of Operational Research* 198(3), 909-921.

Yang, H., & Bell, M. (1997). Traffic restraint, road pricing and network equilibrium. *Transportation Research Part B: Methodological*, *31*(4), 303–314.

Yang, H., & Bell, M.G.H. (1998). A capacity paradox in network design and how to avoid it. *Transportation Research Part A*, *32* (7), 539-545.

Yin, Y., Madanat, S.M., & Lu, X.-Y. (2009). Robust improvement schemes for road networks under demand uncertainty. *European Journal of Operational Research*, *198*(2), 470-479.

Yao, J., & Chen, A. (2014) An analysis of logit and weibit path choices in stochastic assignment paradox. *Transportation Research Part B*, *69*, 31-49.

Yao, J., Chen, A., Ryu, S., & Shi, F. (2014) A general unconstrained optimization formulation for the combined distribution and assignment problem. *Transportation Research Part B* 59, 137-160.

Yao, J., Huang, W., Chen, A., An, S., & Xu, G. (2019). Paradox links can improve system efficiency: An illustration in traffic assignment problem. *Transportation Research Part B*, *129*, 35-49.

Yu, Q., Fang, D., & Du, W. (2014). Solving the logit-based stochastic user equilibrium problem with elastic demand based on the extended traffic network model. *European Journal of Operational Research, 239*, 112-118.

Yuan, Y. (2008). Step-Sizes for the gradient method. *AMS/IP Studies in Advanced Mathematics*, *42*, 785-796.

Zhou, B., Li, X., & He, J. (2014). Exploring trust region method for the solution of logit-based stochastic user equilibrium problems. *European Journal of Operational Research, 239*, 46-57.

Zhou, Z., Chen, A., & Wong, S. (2009). Alternative formulations of a combined trip generation, trip distribution, modal split, and trip assignment model. *European Journal of Operational Research*, *198*, 129-138.

Zhou, Z., Chen, A., & Bekhor, S. (2012). C-logit stochastic user equilibrium model: formulations and solution algorithm. *Transportmetrica*, *8*(1), 17-41.