Available online at www.sciencedirect.com

ScienceDirect

European Journal of Operational Research 188 (2008) 354–369

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/ejor

Discrete Optimization

An improved origin-based algorithm for solving the combined distribution and assignment problem

Meng Xu [a,*], Anthony Chen [b], Ziyou Gao [a]

[a] *Institute of Systems Science, School of Traffic and Transportation, Beijing Jiaotong University, No. 3 of Shangyuan Residence, Haidian District, Beijing 100044, China*

[b] *Department of Civil and Environmental Engineering, Utah State University, Logan, UT 84322-4110, USA*

Received 24 July 2006; accepted 28 April 2007
Available online 10 May 2007

Abstract

This paper is to further study the origin-based (OB) algorithm for solving the combined distribution and assignment (CDA) problem, where the trip distribution follows a gravity model and the traffic assignment is a user-equilibrium model. Recently, the OB algorithm has shown to be superior to the Frank–Wolfe (FW) algorithm for the traffic assignment (TA) problem and better than the Evans' algorithm for the CDA problem in both computational time and solution accuracy. In this paper, a modified origin–destination (OD) flow update strategy proposed by Huang and Lam [Transportation Research Part B 26 (4), 1992, 325–337] for CDA with the Evans' algorithm is adopted to improve the OB algorithm for solving the CDA problem. Convergence proof of the improved OB algorithm is provided along with some preliminary computational results to demonstrate the effect of the modified OD flow update strategy embedded in the OB algorithm.

Keywords: Combined distribution and assignment problem; Evans' algorithm; Origin-based algorithm; Trip distribution; Traffic assignment

1. Introduction

Transportation system can be regarded as the combination of different elements and their interactions, which produce the demand for travel and the supply of transportation services to satisfy this demand. When studying improvements to a transportation system, there are various alternatives. Evaluating each alternative requires forecasts of travel patterns. Travel forecasting is a complex problem and has been receiving much attention in the transportation field. Traditionally, this problem was simplified by considering a sequence of steps or stages: trip generation (travel choice), trip distribution (destination choice), modal split (mode choice) and traffic assignment (route choice). This simplification has been adopted in transportation planning studies for several decades, and continues to be the current practice (Boyce and Xiong, 2007).

* Corresponding author. Tel.: +86 10 5168 7070; fax: +86 10 5168 7127.
E-mail addresses: xumeng@jtys.bjtu.edu.cn (M. Xu), achen@cc.usu.edu (A. Chen), gaoziyou@jtys.bjtu.edu.cn (Z. Gao).

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2007.04.047

As noted by many researchers (e.g., Oppenheim, 1995; Garret and Wachs, 1996; McNally, 2000; Boyce, 2002), the sequential four-step procedure suffers from inconsistent consideration of travel times and congestion effects in various steps of the procedure. As a remedy, a feedback mechanism is often introduced to resolve the inconsistency, but generally, this approach does not fulfilled it completely and rigorously (Boyce, 2002). An alternative approach to this problem is to adopt a mathematical model using behavioral assumptions (typically from random utility theory) as mathematical conditions to seek solutions that satisfy these conditions. Such models are referred to as 'combined' or 'integrated' models.

Models that combine several travel choices together have been studied for several decades. The first constrained convex optimization formulation for the user-equilibrium assignment problem with elastic demand was first proposed by Beckmann et al. (1956) that considered travelers between every origin–destination (OD) pair to be a function of the travel service for that OD pair. Florian et al. (1975) and Evans (1976) extended the convex optimization formulation to consider destination choice by providing a combined distribution and assignment (CDA) model, where the trip distribution follows a gravity model with a negative exponential deterrence function and the traffic assignment follows a user-equilibrium model. Oppenheim (1993) added variable destination costs in the CDA model to capture the congestion effect at the destinations. Lam and Huang (1992), Boyce and Bar-Gera (2004) and Ho et al. (2006) extended the CDA model to account for multiple user classes, while Florian and Nguyen (1978) considered modal split in the CDA model. Friesz (1981) provided an equivalent optimization problem for combined multiclass distribution, assignment, and model split which obviates symmetry restrictions. Wong et al. (2004) proposed an optimization model for combining distribution, hierarchical mode choice, and assignment network with multiple user and mode classes. Safwat and Magnanti (1988) developed a combined model by integrating all four steps in sequential demand forecasting based on random utility theory. Oppenheim (1995) made further extension to simultaneously consider the travel-destination-mode-route choice, which is based on the multinomial logit model in a hierarchical structure by assuming each traveler is a customer of urban trip, whose choice is reflected by the utility and budget constraints.

Of all the combined models discussed above, the CDA model is particularly important in many transportation applications, because it can help transportation planners to understand the interactions between land use and transportation, and hence to determine the future direction of urban development and transportation systems improvement plan. For example, many researchers have adopted the CDA model as the lower-level problem in a bilevel programming framework to model different land use and transportation issues. Yang et al. (2000) used the CDA model for modeling the capacity and level of service of urban transportation networks; Tam and Lam (2000) adopted the CDA model for determining the maximum number of cars subject to the capacity of the road network and the number of parking spaces available; Lin and Feng (2003) applied the CDA model in a land use – network design problem for analyzing the integrated layouts of land uses, public facilities, transport network and travel demands; Lee et al. (2006) examined the equity issue associated with land use development in terms of the change of equilibrium OD travel costs; Yim (2005) used the CDA model as a mapping to map the land use pattern to the link-loading pattern in a network for optimizing the network reliability with respect to the residential and employment allocations and network enhancements; and Ho and Wong (2007) embedded the CDA model of housing locations and traffic equilibrium to determine the optimal housing provision pattern in a continuum transportation system.

For solution approaches, several algorithms have been proposed in the literature for solving the CDA model formulated as a constrained convex optimization problem. Evans (1976) presented a partial linearization method, which is a descent algorithm for continuous optimization problems (Patriksson, 1994). Florian et al. (1975) and Florian and Nguyen (1978) presented another algorithm based on the Frank and Wolfe (1956) method. However, Evans' algorithm has better convergence characteristics according to Boyce (1984) and LeBlanc and Farhangian (1981).

Based on two unproven conjectures, Horowitz (1989) proposed a modification to the Evans' algorithm with the intent of reducing computational times and memory requirements. However, as noted by Huang and Lam (1992), Horowitz's modified algorithm does not always converge to the optimal solution. To rectify the problem, Huang and Lam (1992) proposed a further modification to Horowitz's modified algorithm and provided rigorous proof of convergence. On the other hand, Lundgren and Patriksson (1998) proposed a solution method, which combines the Evans' algorithm and the disaggregate simplicial decomposition (DSD) method developed by Larsson and Patriksson (1992). This method is expected to have better computational efficiency and higher solution accuracy than the Evans' algorithm since the algorithm operates in the path-flow domain. However, combination of the Evans' algorithm and the second-order DSD algorithm may cause non-convergence.

Recently, Bar-Gera and Boyce (2003), Bar-Gera and Boyce (2006) extended the OB algorithm for solving the CDA model as a fixed point problem. Compared to the Evans' algorithm, the OB algorithm was more effective in achieving highly accurate solutions. When implementing the OB algorithm, it was observed that the step-size determination in the OD flow update step has a significant influence on the performance of the algorithm. As a means to improve the efficiency of the OB algorithm, this paper adopts the modified line search strategy proposed by Huang and Lam (1992) for the Evans' algorithm. This strategy is expected to be effective in improving the performance of the OB algorithm for solving the CDA model.

The rest of the paper is organized as follows. In the next section, a summary of Evans' CDA model and the OB algorithm is provided. The improved OB algorithm with a modified OD flow update method for the doubly constrained CDA problem is presented in Section 3. Convergence proof of the improved OB algorithm is provided Section 4. Some preliminary computational results are presented in Section 5 to demonstrate the effect of the modified OD flow update strategy embedded in the OB algorithm. Some concluding remarks are discussed in Section 6.

2. Summary of Evans' formulation and OB algorithm

This section provides a summary of Evans' formulation and the origin-based algorithm for solving the CDA model. The presentation in this summary section follows Evans (1976), Huang and Lam (1992), Bar-Gera and Boyce (2003) and Yim (2005). Notation is provided firstly for convenience, followed by the convex programming formulation and the OB algorithm for the CDA model.

2.1. Notation

$\boldsymbol{N}$	is the set of nodes in the network
$\boldsymbol{A}$	is the set of links in the network
$\boldsymbol{Z}$	is the set of zones in the network
$\boldsymbol{Z}_p$	is the set of destinations for origin p
$\boldsymbol{Z}_q$	is the set of origins for destination q
A_p	is a restricting subnetwork for origin p
a_{h}	is the head node of link a
a_{t}	is the tail node of link a
lcn_j	is the last common node to node j
NB_j	is the non-basic approaches to node j
d_{pq}	is the OD flow from origin p to destination q; $\boldsymbol{d} = (\ldots, d_{pq}, \ldots)$
O_p	is the number of trips per unit time that begin at origin p
D_q	is the number of trips per unit time that end at destination q
r_p	is a balancing factor for origin p
s_q	is a balancing factor for destination q
x_a	is the traffic flow on link a; $\boldsymbol{x} = (\ldots, x_a, \ldots)$
$c_a(x_a)$	is the travel cost on link a with traffic flow x_a and assumed to be a continuous, strictly increasing function; $\boldsymbol{c} = (\ldots, c_a(x_a), \ldots)$
x_{ap}	is the traffic flow on link a from origin p
α_{ap}	is the approach proportion of link a from origin p; $\boldsymbol{\alpha} = (\ldots, \alpha_{ap}, \ldots),\ 0 \leqslant \alpha_{ap} \leqslant 1;\ \sum_{\substack{a\in \boldsymbol{A}\\ a_h=i}} \alpha_{ap} = 1,\ \forall i \in \boldsymbol{N},\ p \in \boldsymbol{Z}$
g_{jp}	is the traffic flow to node j from origin p
o_{ip}	is the maximum cost to node i from origin p
u_{ij}	is the minimum cost of traveling from node i to node j

μ_a is the average cost for link a
v_a is the approximated derivative of μ_a with respect to flow on link a
σ_j is the average cost to node j
ρ_j is the approximated derivative of σ_j with respect to flow to node j
I_{Main} is the number of main iterations
I_{Inner} is the number of inner iterations
β is the impedance factor
Θ is an algorithmic map

2.2. Convex programming formulations

The constraints of the trip distribution and assignment problem are given as follows:

$$x_{ap} \geqslant 0 \quad \forall a \in \boldsymbol{A},\ p \in \boldsymbol{Z}, \tag{1}$$

$$x_a = \sum_{p\in \boldsymbol{Z}} x_{ap} \quad \forall a \in \boldsymbol{A}, \tag{2}$$

$$d_{pq} \geqslant 0 \quad \forall p \in \boldsymbol{Z},\ q \in \boldsymbol{Z}_p, \tag{3}$$

$$\sum_{\substack{a\in A\\ a_t=p}} x_{ap} - \sum_{\substack{a\in A\\ a_h=p}} x_{ap} = O_p \quad \forall p \in \boldsymbol{Z}, \tag{4}$$

$$\sum_{\substack{a\in A\\ a_h=q}} x_{ap} - \sum_{\substack{a\in A\\ a_t=q}} x_{ap} = d_{pq} \quad \forall p \in \boldsymbol{Z},\ q \in \boldsymbol{Z}_p, \tag{5}$$

$$\sum_{\substack{a\in A\\ a_h=i}} x_{ap} - \sum_{\substack{a\in A\\ a_t=i}} x_{ap} = 0 \quad \forall p \in \boldsymbol{Z},\ i \in \boldsymbol{N},\ i \notin \boldsymbol{Z}_p \cup \{p\}, \tag{6}$$

$$\sum_{q\in \boldsymbol{Z}_p} d_{pq} = O_p \quad \forall p \in \boldsymbol{Z}, \tag{7}$$

$$\sum_{p\in \boldsymbol{Z}_q} d_{pq} = D_q \quad \forall q \in \boldsymbol{Z}, \tag{8}$$

$$u_{pp} = 0 \quad \forall p \in \boldsymbol{Z}, \tag{9}$$

$$u_{pa_h} \leqslant u_{pa_t} + c_a(x_a) \quad \forall p \in \boldsymbol{Z},\ a \in \boldsymbol{A}, \tag{10}$$

$$u_{pa_h} = u_{pa_t} + c_a(x_a), \quad \text{if } x_{ap} > 0,\ a \in \boldsymbol{A}, \tag{11}$$

$$d_{pq} = r_p O_p s_q D_q \exp(-\beta u_{pq}) \quad \forall p \in \boldsymbol{P},\ q \in \boldsymbol{Z}_p. \tag{12}$$

If O_p, D_q and β are given, then conditions (3), (7), (8) and (12) define the trip distribution problem. This problem is also called the doubly constrained gravity model with a negative exponential deterrence function, and a unique and optimal solution (d^*_{pq}) can be obtained for the doubly constrained gravity model by using the iterative method from Fratar (1954). An equivalent minimization optimization program for the trip distribution problem can be written as follows:

$$\begin{aligned} &\text{Min} \quad G(\boldsymbol{d}) = \sum_{p\in \boldsymbol{Z}} \sum_{q\in \boldsymbol{Z}_p} d_{pq}(\ln d_{pq} + \beta u_{pq} - 1) \\ &\text{s.t.} \quad (3), (7) \text{ and } (8). \end{aligned} \tag{13}$$

As $G(\boldsymbol{d})$ is a strictly convex function with respect to $\boldsymbol{d}$, the solution to this problem is unique (Theorem 1 of Evans, 1973).

If d_{pq} and $c_a(x_a)$ are known, then conditions (1), (2), (4), (5), (6), (9), (10) and (11) define the traffic assignment problem. These conditions represent the first Wardrop principle (1952), and are equivalent to the Kuhn–Tucker necessary conditions of the following convex minimization program:

$$\begin{aligned} &\text{Min} \quad H(\boldsymbol{x}) = \sum_{a \in \boldsymbol{A}} \int_0^{x_a} c_a(x)\mathrm{d}x \\ &\text{s.t.} \quad (1), (2), (4), (5) \text{ and } (6). \end{aligned} \tag{14}$$

Function $H(\boldsymbol{x})$ is a strictly convex function with respect to x, as $c_a(x_a)$ is a strictly increasing function for all $a \in \boldsymbol{A}$. The solution to this problem is unique in terms of the link flows, but it is not unique in terms of the link flows by origin.

By using the separable convex programming theory of Rockafellar (1967), Evans (1976) defined a convex minimization program that combines the trip distribution problem and the traffic assignment problem into the following minimization program with a strictly convex objective function:

$$\begin{aligned} &\text{Min} \quad P(\boldsymbol{x}, \boldsymbol{d}) = \sum_{a \in \boldsymbol{A}} \int_0^{x_a} c_a(x)\mathrm{d}x + \frac{1}{\beta} \sum_{p \in \boldsymbol{Z}} \sum_{q \in \boldsymbol{Z}_p} d_{pq}(\ln d_{pq} - 1) \\ &\text{s.t.} \quad (1)\text{–}(8). \end{aligned} \tag{15}$$

To solve the above minimization program, Evans (1976) developed a descent algorithm based on the partial linearization method, which guarantees to converge to the unique, optimal solution. The next subsection discusses a recent development using the OB algorithm for solving the CDA model formulated as a convex minimization program.

2.3. Origin-based algorithm for solving the CDA model

Recent research on the OB algorithm (Bar-Gera, 2002) demonstrated that it is one of the state-of-the-art algorithms for solving the traffic assignment problem. There are two key steps in this algorithm: (1) restricting the origin-based subnetworks to be acyclic and (2) shifting flows within the acyclic subnetwork using cost deviations. Since the restricting subnetwork is always a-cyclic for a given origin, it permits a simple route flow

Initialization:
  Find the initial OD flows
  for each p in $\boldsymbol{Z}$ do
    A_p = tree of minimum cost routes from p
    f_{ap} = All-or-Nothing assignment using A_p with the initial OD flows
  end for

Main Loop:
  for $n=1$ to number of main iterations (I_{Main})
    Update O-D flows, retain route proportions
    Update approach proportions $\{\alpha_{ap}\}$ for each link a
    for p in $\boldsymbol{Z}$ do
      Update restricting subnetwork A_p
      Update approach proportions $\{\alpha_{ap}\}$ for each link a
    end for
    for $m=1$ to number of inner iterations (I_{Inner})
      for each p in $\boldsymbol{Z}$ do
        Update approach proportions $\{\alpha_{ap}\}$ for each link a
      end for
    end for
  end for

Fig. 1. Origin-based algorithm for the CDA model (Bar-Gera and Boyce, 2003).

interpretation, enables a definition of cost, and allows for a definition of topological order. Using the approach proportions, the memory required to store routes is significantly reduced. The OB algorithm is considered suitable for large-scale networks due to its computational efficiency and modest memory requirements.

In the implementation of the OB algorithm for the traffic assignment problem, it starts with trees of minimum cost routes as restricting subnetworks. Then, it considers the flows from each origin separately in a sequential order. For each origin, the restricting subnetwork is updated, and the origin-based approach proportions are adjusted within the given restricting subnetwork. Extending the OB algorithm to account for the trip distribution step requires adding a procedure to update the OD flows while keeping the route proportions fixed (Bar-Gera and Boyce, 2003), as shown in Fig. 1.

Experimental results in Bar-Gera and Boyce (2003) have demonstrated the effectiveness of the OB algorithm in achieving highly accurate solutions as compared to the Evans' algorithm and the four-step procedure with a "feedback" mechanism. In the next section, we will further study the OB algorithm to solve the doubly constrained CDA problem.

3. Modified OD flow update method

When implementing the OB algorithm, we observed that the step-size determination in the OD flow update step has a significant influence on the performance of the algorithm. Similar findings were also observed in the Evans' algorithm (Horowitz, 1989; Huang and Lam, 1992) and in the OB algorithm (Bar-Gera and Boyce, 2003). Majority of the computational times spent in the step-size determination is controlled by the trip distribution term in Eq. (15) since it is nearly flat with respect to the step size. As a means to improve the efficiency of the OB algorithm, we adopt the modified line search strategy proposed by Huang and Lam (1992) embedded in the OB algorithm for solving the CDA model. The modified OD flow update method (MODFUM) for a given iteration n is given as follows.

(1) Solve the doubly constrained gravity model to find the new auxiliary trip distribution v_{pq}^n (see Ortuzar and Willumsen, 2001).
(2) Find the new auxiliary link flow y_a^n using v_{pq}^n.
(3) Let

$$M(\boldsymbol{x}^{n-1},\boldsymbol{y}^n,\boldsymbol{d}^n)=\sum_{a\in \boldsymbol{A}} c_a(x_a^{n-1})y_a^n+\frac{1}{\beta}\sum_{p\in \boldsymbol{Z}}\sum_{q\in \boldsymbol{Z}_p} d_{pq}^n(\log d_{pq}^n-1). \tag{16}$$

If $M(\boldsymbol{x}^{n-1},\boldsymbol{x}^n,\boldsymbol{v}^n)=M(\boldsymbol{x}^{n-1},\boldsymbol{y}^n,\boldsymbol{d}^n)$, then $(\boldsymbol{x}^n,\boldsymbol{v}^n)$ is the optimal solution and the procedure is terminated; otherwise go to Step (4).
(4) Compute

$$\Delta_{n1}=\frac{1}{\beta}\sum_{p\in \boldsymbol{Z}}\sum_{q\in \boldsymbol{Z}_P}(v_{pq}^{n-1}\log v_{pq}^{n-1}-d_{pq}^n\log d_{pq}^n). \tag{17}$$

If $\Delta_{n1}=0$, then set $x_a^{n,\lambda}=(1-\lambda)x_a^{n-1}+\lambda y_a^n$, and solve the following one-dimensional line search problem to obtain the step size λ^*

$$\operatorname*{Min}_{\lambda} H(\lambda)=\sum_{a\in \boldsymbol{A}}\int_0^{x_a^{n,\lambda}} c_a(x)\mathrm{d}x. \tag{18}$$

Set

$$x_a^n=(1-\lambda^*)x_a^{n-1}+\lambda^* y_a^n, \tag{19}$$

$$d_{pq}^n=(1-\lambda^*)d_{pq}^{n-1}+\lambda^* v_{pq}^n, \tag{20}$$

where $\lambda\in[0,1]$ and x_a^{n-1} and $d_{pq}^{n-1}$are the link flows and OD flows at iteration $n-1$. Set n to $n+1$ and return; otherwise go to Step (5).

(5) If $\Delta_{n1} > 0$, then compute

$$\Delta_{n2} = \sum_{a \in \boldsymbol{A}} c_a(x_a^{n-1})(y_a^n - x_a^{n-1}). \tag{21}$$

If $\Delta_{n2} < 0$, then solve the one-dimensional line search problem (18) to obtain the step size λ^* and update (x_a^n, d_{pq}^n) using (19) and (20). Set n to $n+1$ and return; otherwise solve the original one-dimensional line search problem to obtain λ^*

$$\operatorname*{Min}_{\lambda} P(\boldsymbol{x}^{n,\lambda}, \boldsymbol{v}^{n,\lambda}) = \sum_{a \in \boldsymbol{A}} \int_0^{x_a^{n,\lambda}} c_a(x)\mathrm{d}x + \frac{1}{\beta} \sum_{p \in \boldsymbol{Z}} \sum_{q \in \boldsymbol{Z}_q} v_{pq}^{n,\lambda}(\log v_{pq}^{n,\lambda} - 1). \tag{22}$$

Update (x_a^n, d_{pq}^n) using (19) and (20), Set n to $n+1$ and return; otherwise go to Step (6).

(6) Since $\Delta_{n1} < 0$, solve problem (18) to obtain λ^*, use (19) and (20) to update (x_a^n, d_{pq}^n), and compute

$$\theta_{n1} = \sum_{a \in \boldsymbol{A}} \int_0^{x_a^{n-1}} c_a(x)\mathrm{d}x - \sum_{a \in \boldsymbol{A}} \int_0^{x_a^n} c_a(x)\mathrm{d}x, \tag{23}$$

$$\theta_{n2} = \frac{1}{\beta}\left[\sum_{p \in \boldsymbol{Z}} \sum_{q \in \boldsymbol{Z}_p} d_{pq}^{n-1}(\log d_{pq}^{n-1} - 1) - \sum_{p \in \boldsymbol{Z}} \sum_{q \in \boldsymbol{Z}_p} d_{pq}^n(\log d_{pq}^n - 1)\right]. \tag{24}$$

If $\theta_{n2} \geqslant 0$, return; otherwise.
If $\theta_{n2} < 0$ and $|\theta_{n1}| \geqslant |\theta_{n2}|$, return; otherwise.
If $\theta_{n2} < 0$ and $|\theta_{n1}| < |\theta_{n2}|$, then solve problem (22) to obtain the step size λ^*, update (x_a^n, d_{pq}^n) using (19) and (20), and return.
With the above MODFUM, the improved OB algorithm implementation for the CDA model is described as follows.

Initialization

(1) Find the initial OD flow d_{pq}^0 using the minimum route cost u_{pq} by solving the doubly constrained gravity model.
(2) Find the initial restricting subnetwork A_p^0 for each origin $p \in \boldsymbol{Z}$ using any shortest path algorithm (e.g., Dijkstra's algorithm).
(3) Find the origin-based link flows x_{ap}^0 for each origin $p \in \boldsymbol{Z}$ by performing an all-or-nothing assignment to obtain the initial link flow pattern.
(4) Update the link costs using the initial link flow pattern.
(5) Initialize the origin-based node flows g_{jp}^0 for each origin $p \in \boldsymbol{Z}$.
(6) Initialize the origin-based approach proportions α_{ap}^0 for each origin $p \in \boldsymbol{Z}$.

Main loop

For $n = 1$ to number of main iterations (I_{Main})
for p in $\boldsymbol{Z}$ do
Update restricting subnetwork A_p
(1) Remove all of links with $\alpha_{ap} = 0$ in the restricting subnetwork A_p.
(2) Compute the maximum cost o_{ip} to each node i from the origin p.
(3) Add link (i,j) that satisfies $o_{ip} < o_{jp}$ to the restricting subnetwork A_p.
(4) Find the last common node lcn_j to node j in the restricting subnetwork A_p.
Update approach proportions $\{\alpha_{ap}\}$ for each link a
Compute the step size $\lambda = 2^{-k}, k = 0, 1, 2, \ldots$, repeat the following Steps (1)–(4) until it satisfies

$$\sum_{a \in A} \Delta x_a c_a(x_a + \lambda \Delta x_a) > 0. \tag{25}$$

(1) Compute the following average costs
(1.1) Compute the average cost to all nodes:

$$\sigma_p(\boldsymbol{\alpha}, \boldsymbol{c}) = 0, \tag{26}$$

$$\sigma_j(\boldsymbol{\alpha}, \boldsymbol{c}) = \sum_{\substack{a\in A_p \\ a_h=j}} \alpha_a \mu_a(\boldsymbol{\alpha}, \boldsymbol{c}) \quad \forall j \notin \boldsymbol{P}. \tag{27}$$

(1.2) Compute the average approach cost for all links:

$$\mu_a(\boldsymbol{\alpha}, \boldsymbol{c}) = c_a + \sigma_{a_t}(\boldsymbol{\alpha}, \boldsymbol{c}) = c_a + \sum_{\substack{a'\in A_p \\ a'_h=a_t}} \alpha_{a'} \mu_{a'}(\boldsymbol{\alpha}, \boldsymbol{c}). \tag{28}$$

(2) Compute the Hessian approximations
(2.1) Compute the derivative of the average cost to all nodes:

$$\rho_p(\boldsymbol{\alpha}, \boldsymbol{c}') = 0, \tag{29}$$

$$\rho_j(\boldsymbol{\alpha}, \boldsymbol{c}') = \sum_{\substack{a\in A_p \\ a_h=j}} \alpha_a^2 v_a(\boldsymbol{\alpha}, \boldsymbol{c}') \quad \forall j \notin \boldsymbol{P}. \tag{30}$$

(2.2) Compute the derivative of the average approach cost for all links:

$$v_a(\boldsymbol{\alpha}, \boldsymbol{c}') = c'_a + \rho_{a_t}(\boldsymbol{\alpha}, \boldsymbol{c}') = c'_a + \sum_{\substack{a'\in A_p \\ a'_h=a_t}} \alpha_{a'}^2 v_{a'}(\boldsymbol{\alpha}, \boldsymbol{c}'). \tag{31}$$

(3) Compute the desirable amount of flow proportion $\Theta_1^{a\to b}(\boldsymbol{\alpha}, \boldsymbol{c}, \boldsymbol{c}')$ to be shifted between two alternative approaches a and b, $a_b = b_b = j$.

$$\Theta_1^{a\to b}(\boldsymbol{\alpha}, \boldsymbol{c}, \boldsymbol{c}') = \begin{cases} \min(\alpha_a, \lambda \frac{x_{a\to b}(\boldsymbol{\alpha}, \boldsymbol{c}, \boldsymbol{c}')}{g_{jp}(\boldsymbol{\alpha}))}, & g_{jp} > 0 \\ \alpha_a, & g_{jp} = 0, \mu_a > \mu_b \\ [0, \alpha_a], & g_{jp} = 0, \mu_a = \mu_b, \end{cases} \tag{32}$$

where the desirable shifted flow $x_{a\to b}(\boldsymbol{\alpha}, \boldsymbol{c}, \boldsymbol{c}')$ is defined as

$$x_{a\to b}(\boldsymbol{\alpha}, \boldsymbol{c}, \boldsymbol{c}') = \frac{\mu_a(\boldsymbol{\alpha}, \boldsymbol{c}) - \mu_b(\boldsymbol{\alpha}, \boldsymbol{c})}{\max(\varepsilon_v, v_a(\boldsymbol{\alpha}, \boldsymbol{c}') + v_a(\boldsymbol{\alpha}, \boldsymbol{c}') - 2\rho_{\mathrm{lcn}_j}(\boldsymbol{\alpha}, \boldsymbol{c}'))}, \tag{33}$$

where ε_v is a small positive constant to overcome the "zero" derivative estimation problem.
(4) Aggregate shifts from approaches to the same node j.

$$\Theta_1^{j:b}(\boldsymbol{\alpha}, \boldsymbol{c}, \boldsymbol{c}') = \left\{ \Delta\boldsymbol{\alpha} : \begin{array}{l} \Delta\alpha_a \in -\Theta_1^{a\to b}(\boldsymbol{\alpha}, \boldsymbol{c}, \boldsymbol{c}') \quad \forall a \in NB_j, \\ \Delta\alpha_a = -\sum\limits_{a\in NB_j} \Delta\alpha_a, \\ \Delta\alpha_{a'} = 0, \quad a' \neq j, \end{array} \right. \tag{34}$$

end for
for $m = 1$ to number of inner iterations (I_{Inner})
 for each p in $\boldsymbol{Z}$ do
  Update approach proportions $\{\alpha_{ap}\}$ for each link a (same as above).
 end for
end for
end for

4. Convergence of the improved origin-based algorithm

Before proving the convergence of the OB algorithm with the MODFUM for solving the doubly constrained CDA model, we first recall some intermediate results from Huang and Lam (1992):

Lemma 1. *Given $\boldsymbol{x}^{n-1}$, steps* (1) *and* (2) *of the MODFUM in Section* 3 *is equivalent to finding a solution for the following convex minimization program*:

$$\begin{aligned}\text{Min}\quad & M(\boldsymbol{x}^{n-1},\boldsymbol{y}^n,\boldsymbol{d}^n)=\sum_{a\in\boldsymbol{A}}c_a(x_a^{n-1})y_a^n+\frac{1}{\beta}\sum_{p\in\boldsymbol{Z}}\sum_{q\in\boldsymbol{Z}_p}d_{pq}^n(\log d_{pq}^n-1)\\ \text{s.t.}\quad & \Omega=\{(\boldsymbol{y}^n,\boldsymbol{d}^n)|(1)\text{–}(8)\text{ are satisfied}\}.\end{aligned}$$

Lemma 2. *Denote the optimal solution of the problem given above by $(\boldsymbol{y}^n,\boldsymbol{d}^n)$. If $(\boldsymbol{x}^{n-1},\boldsymbol{v}^{n-1})$ is a non-optimal point in Ω, then*

$$M(\boldsymbol{x}^{n-1},\boldsymbol{y}^n,\boldsymbol{d}^n)<M(\boldsymbol{x}^{n-1},\boldsymbol{x}^{n-1},\boldsymbol{v}^{n-1}). \tag{35}$$

If the inequality in (35) *changes to equality, then $(\boldsymbol{x}^{n-1},\boldsymbol{v}^{n-1})$ is a unique, optimal solution for problem* (22).

Lemma 3. *Denote a strictly convex function as*

$$H(\lambda)=\sum_{a\in\boldsymbol{A}}\int_0^{x_a^{n,\lambda}}c_a(x)\mathrm{d}x, \tag{36}$$

where $x_a^{n,\lambda}=(1-\lambda)x_a^{n-1}+\lambda y_a^n$. Then $\text{Min}_{\lambda\in[0,1]}\ H(\lambda)$ exists if and only if

$$\left.\frac{\mathrm{d}H}{\mathrm{d}\lambda}\right|_{\lambda=0^+}<0. \tag{37}$$

Lemma 4. *At every iteration, $\left.\frac{\mathrm{d}H}{\mathrm{d}\lambda}\right|_{\lambda=0^+}<\Delta_{n1}$.*

Proof. From the definition of derivative,

$$\frac{\mathrm{d}H}{\mathrm{d}\lambda}=\sum_{a\in\boldsymbol{A}}c_a[(1-\lambda)x_a^{n-1}+\lambda y_a^n](y_a^n-x_a^{n-1}).$$

Hence,

$$\left.\frac{\mathrm{d}H}{\mathrm{d}\lambda}\right|_{\lambda=0^+}=\Delta_{n2}.$$

From (35),

$$\sum_{a\in\boldsymbol{A}}c_a(x_a^{n-1})y_a^n+\frac{1}{\beta}\sum_{p\in\boldsymbol{Z}}\sum_{q\in\boldsymbol{Z}_p}d_{pq}^n(\log d_{pq}^n-1)<\sum_{a\in\boldsymbol{A}}c_a(x_a^{n-1})x_a^{n-1}+\frac{1}{\beta}\sum_{p\in\boldsymbol{Z}}\sum_{q\in\boldsymbol{Z}_p}v_{pq}^{n-1}(\log v_{pq}^{n-1}-1). \tag{38}$$

Note that $\sum_{p\in\boldsymbol{Z}}\sum_{q\in\boldsymbol{Z}_p}d_{pq}^n=\sum_{p\in\boldsymbol{Z}}\sum_{q\in\boldsymbol{Z}_p}v_{pq}^{n-1}$; therefore,

$$\Delta_{n2}<\frac{1}{\beta}\sum_{p\in\boldsymbol{Z}}\sum_{q\in\boldsymbol{Z}_P}(v_{pq}^{n-1}\log v_{pq}^{n-1}-d_{pq}^n\log d_{pq}^n).$$

That is,

$$\left.\frac{\mathrm{d}H}{\mathrm{d}\lambda}\right|_{\lambda=0^+}<\Delta_{n1}.\qquad\square$$

Theorem 1. *The MODFUM given in Section 3 guarantees that $\boldsymbol{d}^n$ is in a non-increasing direction.*

Proof. There are three cases based on Δ_{n1}:

Case 1. $\Delta_{n1} = 0$According to Lemma 3, there exists a unique solution $\lambda \in [0, 1]$ for the new one-dimensional minimization problem when $\frac{dH}{d\lambda}\big|_{\lambda=0^+} < 0$. Let

$$x_a^n = (1-\lambda)x_a^{n-1} + \lambda y_a^n \quad \text{and} \quad d_{pq}^n = (1-\lambda)d_{pq}^{n-1} + \lambda v_{pq}^n.$$

Then,

$$\sum_{a \in \boldsymbol{A}} \int_0^{x_a^n} c_a(x)\mathrm{d}x < \sum_{a \in \boldsymbol{A}} \int_0^{x_a^{n-1}} c_a(x)\mathrm{d}x. \tag{39}$$

By the convexity of function $f(x) = x(\log x - 1)$, it follows that

$$\sum_{p \in \boldsymbol{Z}} \sum_{q \in \boldsymbol{Z}_p} d_{pq}^n(\log d_{pq}^n - 1) < (1-\lambda) \sum_{p \in \boldsymbol{Z}} \sum_{q \in \boldsymbol{Z}_p} d_{pq}^{n-1}(\log d_{pq}^{n-1} - 1) + \lambda \sum_{p \in \boldsymbol{Z}} \sum_{q \in \boldsymbol{Z}_p} v_{pq}^n(\log v_{pq}^n - 1).$$

Thus,

$$\sum_{p \in \boldsymbol{Z}} \sum_{q \in \boldsymbol{Z}_p} d_{pq}^n(\log d_{pq}^n - 1) < \sum_{p \in \boldsymbol{Z}} \sum_{q \in \boldsymbol{Z}_p} d_{pq}^{n-1}(\log d_{pq}^{n-1} - 1). \tag{40}$$

Adding (39) and (40) gives

$$P(\boldsymbol{x}^n, \boldsymbol{d}^n) < P(\boldsymbol{x}^{n-1}, \boldsymbol{d}^{n-1}). \tag{41}$$

Case 2. For $\Delta_{n1} > 0$, there are two subcases:

(i) $\Delta_{n2} < 0$. Similar to *Case* 1, (41) is satisfied.

(ii) $0 \leqslant \Delta_{n2} < \Delta_{n1}$. From Lemma 3 and Lemma 4, it shows that if $\frac{dH}{d\lambda}\big|_{\lambda=0^+} \geqslant 0$, then λ takes its lowest bound for minimizing $H(\lambda)$ since $H(\lambda)$ is strictly convex on $[0, 1]$ (i.e., if $\lambda = 0$, it implies that $x^n = x^{n-1}$ and $d^n = d^{n-1}$). Therefore, the new feasible solution is no better than the old one. In this situation, a deadlock occurs. The algorithm should return to use (22) to find the new (x^n, d^n) in order to guarantee (41) is satisfied.

Case *3*. For $\Delta_{n1} < 0$, there are three subcases:

(i) $\theta_{n2} \geqslant 0$. Similar to *Case* 1, (41) is satisfied.

(ii) $\theta_{n2} < 0$ and $|\theta_{n1}| \geqslant |\theta_{n2}|$. Similar to *Case* 1, (41) is satisfied when $|\theta_{n1}| = |\theta_{n2}|$ and $P(\boldsymbol{x}^n, \boldsymbol{d}^n) = P(\boldsymbol{x}^{n-1}, \boldsymbol{d}^{n-1})$.

(iii) $\theta_{n2} < 0$ and $|\theta_{n1}| < |\theta_{n2}|$. From Lemma 3 and Lemma 4, it shows that $\theta_{n1} + \theta_{n2} < 0$. That is,

$$P(\boldsymbol{x}^n, \boldsymbol{d}^n) > P(\boldsymbol{x}^{n-1}, \boldsymbol{d}^{n-1}),$$

which implies the new solution is worse than the old one. Therefore, the algorithm should return to use (22) to find the new $(\boldsymbol{x}^n, \boldsymbol{d}^n)$ in order to guarantee (41) is satisfied.

From the above analysis of different cases, it shows that the MODFUM embedded in the OB algorithm can guarantee the new solution to be a non-increasing sequence. That is,

$$P(\boldsymbol{x}^n, \boldsymbol{d}^n) \leqslant P(\boldsymbol{x}^{n-1}, \boldsymbol{d}^{n-1}). \qquad \square$$

In the implementation of the above OB algorithm with the MODFUM, link array $\boldsymbol{x}$ at each iteration is essentially determined by the current OD flow $\boldsymbol{d}$ and the current approach proportion array $\boldsymbol{\alpha}$. Let $A = \Xi(\boldsymbol{d}, \boldsymbol{a})$ be the set of restricting subnetworks for the next iteration, and $\boldsymbol{\alpha}^{\text{next}} \in \Theta^{\alpha}(\boldsymbol{d}, \boldsymbol{\alpha})$ be the approach proportion for the next iteration. From Bar-Gera (2002) and Bar-Gera and Boyce (2003), A and $\boldsymbol{\Theta}^{\alpha}$ have the following properties:

Lemma 4. *If* $\boldsymbol{\alpha}^{\text{next}} \in \boldsymbol{\Theta}^{\alpha}(\boldsymbol{d},\boldsymbol{\alpha})$, *then* $P[\boldsymbol{x}(\boldsymbol{d},\boldsymbol{\alpha}^{\text{next}}),\boldsymbol{d}] \leqslant P[\boldsymbol{x}(\boldsymbol{d},\boldsymbol{\alpha}),\boldsymbol{d}]$. *Equality may hold only if* $\boldsymbol{x}(\boldsymbol{d},\boldsymbol{\alpha}^{\text{next}}) \leqslant \boldsymbol{x}(\boldsymbol{d},\boldsymbol{\alpha})$.

Lemma 5. *For a sequence with restricting subnetworks A, the mapping* $\Theta^{\alpha}(\boldsymbol{d},\boldsymbol{\alpha})$ *is close. That is, if*

$$\boldsymbol{\alpha}^k \to \boldsymbol{\alpha}^*, \quad \hat{\boldsymbol{\alpha}}^k \to \hat{\boldsymbol{\alpha}}^*, \quad \Xi(\boldsymbol{d},\boldsymbol{a}^k) = A, \quad \hat{\boldsymbol{\alpha}}^k \in \Theta^{\alpha}(\boldsymbol{d},\boldsymbol{\alpha}^k),$$

then $\hat{\boldsymbol{\alpha}}^* \in \Theta^{\alpha}(\boldsymbol{d},\boldsymbol{\alpha}^*)$.

From the algorithm, define

$$\Theta^d(\boldsymbol{d}^k) = \{d^k_{pq} = (1-\lambda)d^{k-1}_{pq} + \lambda v^k_{pq} : \lambda \text{ satisfies } (22)\}.$$

Lemma 6. *For a converging sequence* $\boldsymbol{d}^k \to \boldsymbol{d}^*$, *the mapping* $\Theta^d(\boldsymbol{d}^k)$ *is close.*

Lemma 7. *Given* $\boldsymbol{\alpha}^{k+1} \in \boldsymbol{\Theta}^{\alpha}$, $\boldsymbol{d}^{k+1} \in \boldsymbol{\Theta}^{d}$ *and a subsequence* $\boldsymbol{K}$ *such that*

$$\Xi(\boldsymbol{d}^{k+l},\boldsymbol{\alpha}^{k+l}) = \boldsymbol{A}^{*l}, \quad \boldsymbol{\alpha}^{k+l} \to \boldsymbol{\alpha}^{*l}, \quad \boldsymbol{x}(\boldsymbol{d}^{*l},\boldsymbol{\alpha}^{*l}) = \boldsymbol{x}^* \quad \forall k \in \boldsymbol{K} \ \forall 1 \leqslant l \leqslant |N|.$$

Every limit point of $\{\boldsymbol{\alpha}^k\}$ *satisfies the user-equilibrium conditions.*

Theorem 2. *The improved OB algorithm converges to the optimal solution of the doubly constrained CDA problem.*

Proof. In fact, there exists a subsequence K such that

$$\Xi(\boldsymbol{d}^{k+l},\boldsymbol{\alpha}^{k+l}) = \boldsymbol{A}^{*l}, \quad \boldsymbol{\alpha}^{k+l} \to \boldsymbol{\alpha}^{*l}, \quad \boldsymbol{d}^{k+l} \to \boldsymbol{d}^{*l} \quad \forall k \in K \ \forall 1 \leqslant l \leqslant |N|,$$

which satisfies $\boldsymbol{\alpha}^{k+1} \in \boldsymbol{\Theta}^{\alpha}$ from Lemmas 5 and 6. From Theorem 1 and Lemma 4, it is found that the objective function is a monotonically non-increasing bounded sequence. Therefore, $P(\boldsymbol{d}^{*l},\boldsymbol{\alpha}^{*l}) = P^*$. Furthermore, $\boldsymbol{\alpha}^{*l} = \boldsymbol{\alpha}^*$, $\boldsymbol{d}^{*l} = \boldsymbol{d}^*$, $\boldsymbol{x}(\boldsymbol{d}^{*l},\boldsymbol{\alpha}^{*l}) = \boldsymbol{x}^*$. From Lemma 7, every limit point of $\{\boldsymbol{\alpha}^k\}$ satisfies the user-equilibrium conditions. Therefore, $(\boldsymbol{x}^*,\boldsymbol{d}^*)$ are the optimal solutions of the doubly constrained CDA problem. □

5. Numerical experiment and analysis

In order to test the improved OB algorithm with the MODFUM for solving the doubly constrained CDA problem presented in Section 3, the well-known Sioux Falls network (LeBlanc et al., 1975) is used as an illustration to carry out the numerical experiment and analysis. The Sioux Falls network consists of 24 nodes, 76 links, and 528 OD pairs. The characteristics of the network and OD demands can be found in LeBlanc et al. (1975) or Yim (2005). The impedance factor for trip distribution is set at 0.1. This numerical experiment was conducted with double precision arithmetic on an Intel® Pentium® 4 CPU 1.75 GHz, 512 MB RAM, using the Microsoft Window XP (SP2) operating system. All of the coding was carried out in Compaq Visual Fortran Professional Edition 6.1.0. The precision of any given approximate solution $(\boldsymbol{x},\boldsymbol{d})$ can be measured by the relative gap of trip distribution and the relative gap of traffic assignment as follows:

$$\text{RG\_TD}^n = \frac{\sqrt{\sum_p\sum_q (v^n_{pq} - d^n_{pq})^2}}{\sum_p\sum_q d^n_{pq}},$$

$$\text{RG\_TR}^n = \frac{|H(\boldsymbol{x}^n) - \text{BLB}^n|}{\text{BLB}^n},$$

where $\text{BLB}^n = \max_{n' \leqslant n}[H(\boldsymbol{x}^{n'}) + \nabla H(\boldsymbol{x}^{n'})(\boldsymbol{y}^{n'} - \boldsymbol{x}^n)]$. The stopping criterion for both relative gaps (traffic assignment and trip distribution) is set at 1.0E−15.

The experiment conducted in this section is mainly to examine the difference between the original and improved OB algorithms, the effect of the number of inner iterations (I_{Inner}) and the tradeoff between the number of main iterations (I_{Main}) and the number of inner iterations (I_{Inner}) on the overall convergence of the OB algorithms. For a fixed $I_{\text{Main}} = 200$ and $I_{\text{Inner}} = 15$, Figs. 2 and 3 show the overall solution convergence and the relative gap of trip distribution in log scale versus computation time for both the original and improved OB algorithms. It is clear from the figures that the improved OB algorithm has a faster convergence compared to the original OB algorithm.

It has been found that updating the restricting subnetwork requires much more computational effort than updating the approach proportions and shifting the flows (Bar-Gera, 2002). Thus, an inner iteration loop is added to perform several updates of the approach proportions and flow shifts after each update of the restricting subnetwork. This strategy is similar to performing the equilibration step with a restricted path set several times in the master problem before returning to the column generation step of the disaggregate simplicial decomposition algorithm (Larsson and Patriksson, 1992). The next few figures and tables examines the effect of the number of inner iterations (I_{Inner}) and the tradeoff between I_{Main} and I_{Inner} for a fixed computation budget. Figs. 4 and 6 show the relative gaps of traffic assignment and trip distribution in log scale versus main loop and CPU time for different number of iterations for the inner loop of the OB algorithm (with $I_{\text{Main}} = 200$), while Figs. 5 and 7 show the same information for the improved OB algorithm. From the figures, it is found that inner iterations I_{Inner} have a clearly effect for the convergence of relative gap. With the fixed I_{Main}, the convergence of relative gap might be increases with different scales with the increase of I_{Inner}. For example, in Fig. 7, the convergence of relative gap of trip distribution improves clearly when changing I_{Inner} from 10 to 15, however, the convergence changes little when changing I_{Inner} from 15 to 20. Furthermore, it is sure that the CPU time will need longer when increasing the inner iterations I_{Inner}.

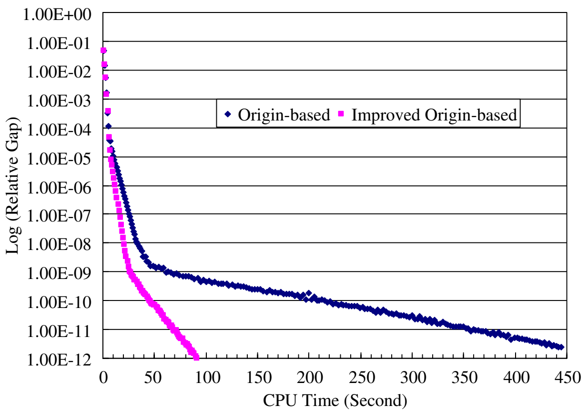

Fig. 2. Overall solution convergence versus CPU time.

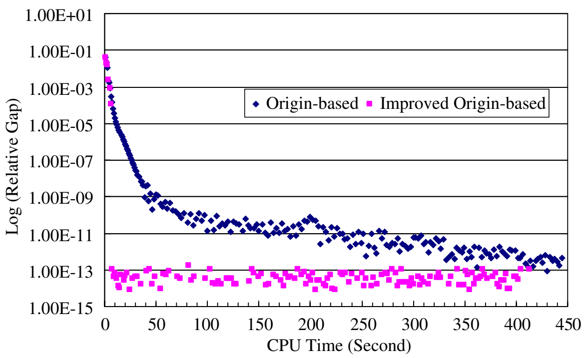

Fig. 3. Relative gap of trip distribution versus CPU time.

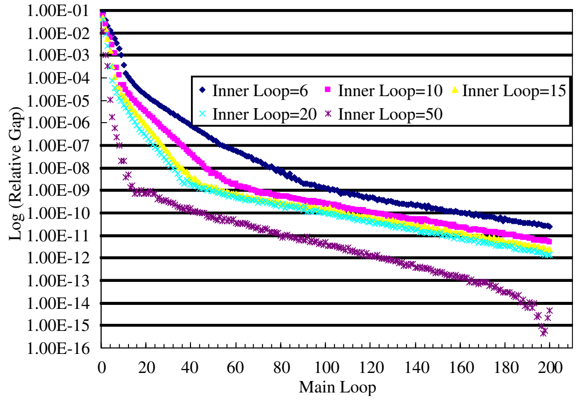

Fig. 4. Relative gap of traffic assignment versus main loop for the OB algorithm.

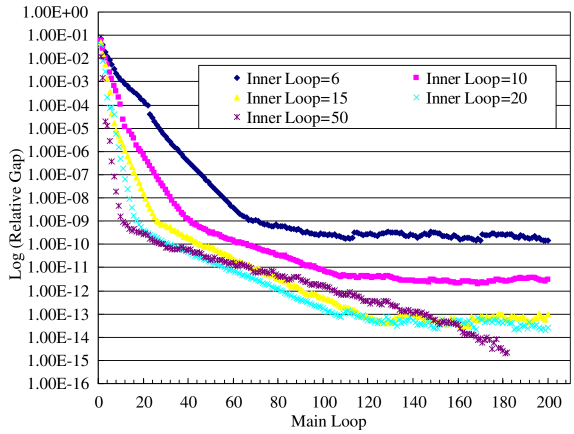

Fig. 5. Relative gap of traffic assignment versus main loop for the improved OB algorithm.

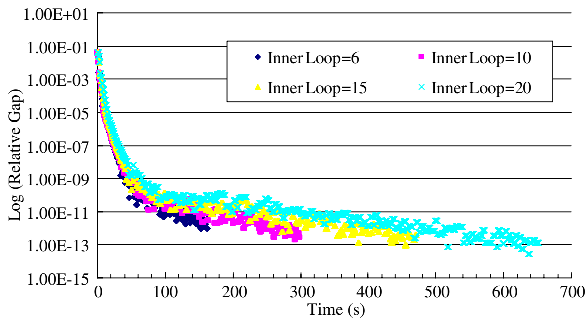

Fig. 6. Relative gap of trip distribution versus CPU time for the OB algorithm.

Tables 1–3 report the effect of computation allocations between I_{Main} and I_{Inner} for three levels of computation budget (i.e., $I_{\text{Main}} \times I_{\text{Inner}}$): 1600, 3000 and 4000. In general, the relative gaps of traffic assignment and trip distribution for both OB algorithms decrease as the number of inner iterations increases. Between the two

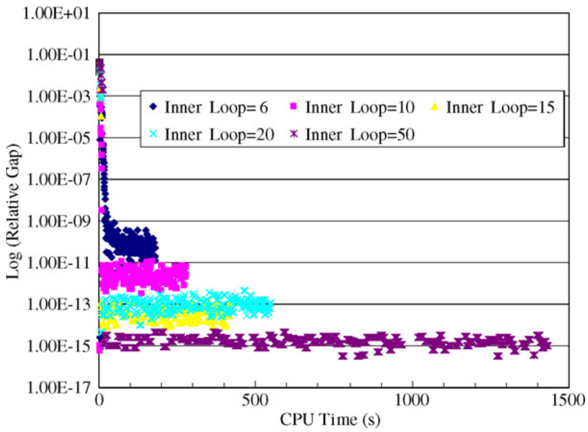

Fig. 7. Relative gap of trip distribution versus CPU time for the improved OB algorithm.

OB algorithms, the improved OB algorithm appears to have smaller relative gaps compared to those of the original OB algorithm. From the limited results shown here, it seems the combination of $I_{\text{Main}} = 200$ and $I_{\text{Inner}} = 15$ (shown in Figs. 2 and 3) can achieve a good balance between convergence accuracy and efficiency. Further, when the total computation budget is fixed (see Tables 1–3), it appears there is an optimal combination of I_{Main} and I_{Inner} to balance the convergence accuracy and computation efficiency. For a computation budget of 1600 in Table 1, $I_{\text{Main}} = 100$ and $I_{\text{Inner}} = 16$ is best. Similarly, $I_{\text{Main}} = 200$ and $I_{\text{Inner}} = 20$ is best for a computation budget of 3000 in Table 2, and $I_{\text{Main}} = 150$ and $I_{\text{Inner}} = 20$ is best for a computation budget of 4000 in Table 3. However, it should be noted that these results are network dependent (i.e., size of network, network characteristics, demand level, etc.). It may not be generalized to other networks.

Table 1
Effect of computation allocations between I_{Main} and I_{Inner} for a computation budget of 1600

ML NO	ML time	IL NO	IL time	RG_TA	RG_TD
320	545.187	5	428.76	1.26E−10	8.32E−12
160	462.594	10	406.852	2.18E−12	2.58E−12
100	212.219	16	196.234	1.10E−10	1.70E−10
80	469.485	20	441.251	1.20E−12	3.14E−14
40	563.25	40	548.72	2.88E−12	3.43E−16
20	577.062	80	569.733	1.36E−10	2.71E−16

Table 2
Effect of computation allocations between I_{Main} and I_{Inner} for a computation budget of 3000

ML NO	ML time	IL NO	IL time	RG_TA	RG_TD
600	488.689	5	389.338	7.05E−11	5.70E−11
300	415.968	10	367.9	3.81E−12	4.18E−12
200	412.859	15	380.376	9.36E−14	1.10E−13
150	398.546	20	374.431	2.27E−14	3.87E−14
75	457.781	40	445.438	3.10E−15	2.32E−15
50	945.797	60	930.984	1.36E−11	0.00E+00
25	1080.984	120	1071.888	2.09E−10	1.38E−15
10	973.937	300	970.5	3.00E−10	2.56E−16

Table 3
Effect of computation allocations between I_{Main} and I_{Inner} for a computation budget of 4000

ML NO	ML time	IL NO	IL time	RG_TA	RG_TD
800	679.546	5	542.868	8.11E−11	5.38E−11
400	596.968	10	528.52	3.91E−12	1.09E−11
200	536.766	20	504.16	2.62E−14	1.17E−13
100 (78)[a]	879.767	40	856.204	4.35E−16	1.37E−15
50	761.407	80	752.69	1.07E−11	5.54E−16
20	1533.703	200	1526.343	9.32E−11	1.04E−15

ML NO = Main loop number,
ML time = Total main loop time,
IL NO = Inner loop number,
IL time = Total inner loop time,
RG_TA = Relative gap of traffic assignment,
RG_TD = Relative gap of trip distribution.
[a] Convergence is reached at the 78th iteration.

6. Conclusions

Much of the attention has recently been focused on the OB algorithm due to its modest memory requirements, computational efficiency and ability to achieve highly accurate detailed solution. As a further improvement, this paper presented a modified OD flow update strategy embedded in the OB algorithm for solving the CDA problem. Convergence proof of the improved OB algorithm was provided. Initial computation results indicated that the modified OD flow update strategy could enhance the OB algorithm if proper choice of the numbers of main iterations and inner iterations was selected. How to determine suitable numbers of main iterations and inner iterations to balance the convergence accuracy and computation efficiency deserves further attention. In addition, more work is needed to examine the efficiency of the improved OB algorithm on larger networks.

Acknowledgements

The authors thank three anonymous referees for their helpful and constructive comments for improving the paper. In addition, the authors are indebted to Professor S.C. Wong of the University of Hong Kong for providing the source codes of the origin-based algorithm and initial ideas for starting this research. The study was supported by the National Science Foundation of the United States (Grant No. CMS-0134161), the National Basic Research Program of China (Project No. 2006CB705500) and the National Natural Science Foundation of China (Grant No. 70631001).

References

Bar-Gera, H., 2002. Origin-based algorithm for the traffic assignment problem. Transportation Science 36 (4), 398–417.
Bar-Gera, H., Boyce, D., 2003. Origin-based algorithms for combined travel forecasting models. Transportation Research B 37 (5), 405–422.
Bar-Gera, H., Boyce, D., 2006. Solving a non-convex combined travel forecasting model by the method of successive averages with constant step sizes. Transportation Research B 40 (5), 351–367.
Beckmann, M., McGuire, C.B., Winsten, C.B., 1956. Studies in the Economics of Transportation. Yale University Press, New Haven, CT.
Boyce, D.E., 1984. Urban transportation network equilibrium and design models: Recent achievements and future prospects. Environment and Planning A16, 1445–1474.
Boyce, D.E., 2002. Is the sequential travel forecasting procedure counterproductive?. Journal of Urban Planning and Development 128 169–183.
Boyce, D., Bar-Gera, H., 2004. Multiclass combined models for urban travel forecasting. Network and Spatial Economics 4, 115–124.
Boyce, D., Xiong, C., 2007. Forecasting travel for very large cities: Challenges and opportunities for China. Transportmetrica 3 (1), 1–19.
Evans, S.P., 1973. A relationship between the gravity model for trip distribution problem in linear programming. Transportation Research 7 (1), 39–61.
Evans, S.P., 1976. Derivation and analysis of some models for combining trip distribution and assignment. Transportation Research 10 (1), 37–57.

Florian, M., Nguyen, S., 1978. A combined trip distribution mode split and trip assignment model. Transportation Research 12 (3), 241–246.

Florian, M., Nguyen, S., Ferland, J., 1975. On the combined distribution–assignment of traffic. Transportation Science 9 (1), 43–53.

Frank, H., Wolfe, P., 1956. An algorithm for quadratic programming. Naval Research Logistics Quarterly 3, 95–110.

Fratar, T.S., 1954. Vehicular trip distribution by successive approximation. Traffic Quarterly 8, 53–56.

Friesz, T.L., 1981. An equivalent optimization problem for combined multiclass distribution, assignment, and model split which obviates symmetry restrictions. Transportation Research B 15 (5), 361–369.

Garret, M., Wachs, M., 1996. Transportation Planning on Trial. Thousand Oaks, California.

Ho, H.W., Wong, S.C., 2007. Housing allocation problem in a continuum transportation system. Transportmetrica 3 (1), 21–39.

Ho, H.W., Wong, S.C., Loo, B.P.Y., 2006. Combined distribution and assignment model for a continuum traffic equilibrium problem with multiple user classes. Transportation Research B 40, 633–650.

Horowitz, A.J., 1989. Tests of an ad hoc algorithm of elastic-demand equilibrium traffic assignment. Transportation Research B 23 (3), 309–313.

Huang, H.J., Lam, W.H.K., 1992. Modified Evans' algorithms for solving the combined trip distribution and assignment problem. Transportation Research B 26 (4), 325–337.

Lam, W.H.K., Huang, H.J., 1992. A combined trip distribution and assignment model for multiple user classes. Transportation Research B 26 (3), 275–287.

Larsson, T., Patriksson, M., 1992. Simplicial decomposition with disaggregated representation for the traffic assignment problem. Transportation Science 26 (1), 4–17.

LeBlanc, L.J., Farhangian, K., 1981. Efficient algorithm for solving elastic demand traffic assignment problems and model split-assignment problems. Transportation Science 15, 306–317.

LeBlanc, L.J., Morlok, E.K., Pierskalla, W.P., 1975. An efficient approach to solving the road network equilibrium traffic assignment problem. Transportation Research 9 (4), 309–318.

Lee, D.-H., Meng, Q., Wu, L., 2006. Equity based land use and transportation problem. Journal of Advanced Transportation 40 (1), 75–93.

Lin, J.J., Feng, C.M., 2003. A bi-level programming for the land use network design problem. Annals of Regional Science 37, 93–105.

Lundgren, J.T., Patriksson, M., 1998. An algorithm for the combined distribution and assignment model. In: Bell, M.G.H. (Ed.), Transportation Networks: Recent Methodological Advances. Pergamon Press, Amsterdam, pp. 239–253.

McNally, M.G., 2000. The activity-based approach. In: Hensher, D.A., Button, K.J. (Eds.), Handbook of Transport Modelling. Pergamon, Oxford, pp. 53–69.

Oppenheim, N., 1993. Equilibrium trip distribution/assignment with variable destination costs. Transportation Research B 27 (3), 207–217.

Oppenheim, N., 1995. Urban Travel Demand Modeling. John Wiley and Sons, New York.

Ortuzar, J.de D., Willumsen, L.G., 2001. Modeling Transport, third ed. Wiley, New York.

Patriksson, M., 1994. The Traffic Assignment Problem: Models and Methods. VSP, Utrecht, The Netherlands.

Rockafellar, R.T., 1967. Convex programming and systems of elementary monotonic relations. Journal of Mathematical Analysis and Applications 19, 543–564.

Safwat, K.N., Magnanti, T.L., 1988. A combined trip generation, trip distribution, model split, and trip assignment model. Transportation Science 18, 14–30.

Tam, M.L., Lam, H.K.W., 2000. Maximum car ownership under constraints of road capacity and parking space. Transportation Research B 34 (3), 145–170.

Wong, K.I., Wong, S.C., Wu, J.H., Yang, H., Lam, W.H.K., 2004. A combined distribution, hierarchical mode choice, and assignment network model with multiple user and mode classes. In: Lee, D.H. (Ed.), Urban and Regional Transportation Modeling: Essays in Honor of David Boyce. Edward Elgar Publishing, Northampton, USA, pp. 25–42.

Yim, K.W., 2005. A Reliability-based Land Use and Transportation Optimization Model. M.S. Thesis, University of Hong Kong, Hong Kong.

Yang, H., Bell, M.G.H., Meng, Q., 2000. Modeling the capacity and level of service of urban transportation networks. Transportation Research B 34 (2), 255–275.