COMPARISONS OF THE FRANK-WOLFE AND EVANS METHODS FOR THE DOUBLY CONSTRAINED ENTROPY DISTRIBUTION/ASSIGNMENT PROBLEM

Huey-Kuo Chen
Professor
Department of Civil Engineering,
National Central University, Taiwan
32054
Tel: +886-3-4227151 ext 4115, Email:
ncutone@cc.ncu.edu.tw

Ying-Chun Chen
Graduate Student
Department of Civil Engineering,
National Central University, Taiwan
32054
Tel: +886-3-4227151 ext 4144, Email:
wink@eagle.seed.net.tw

Abstract : The doubly constrained entropy distribution/assignment problem is to find the O-D demands assuming that both the total flow generated at each origin and the total flow attracted to each destination are fixed and known. This problem has been successfully formulated as an optimization program and two types of solution algorithms developed, i.e., the Frank-Wolfe (FW) and Evans methods. Nevertheless, comparisons of these two algorithms with respect to computational efficiency were not deliberately made in the literature and no clear consensus has been induced.

In this paper, the FW and Evans algorithm methods, or more specifically, Simplex method for Hitchcock problem and RAS algorithm for the gravity model, were intensively compared with four test networks. The results show that the RAS algorithm outperforms in terms of five performance measures, among which shorter computation time contradicts the points made by Sheffi (1985). To consolidate the experiments, further study on computational complexities will be conducted for their dynamic counterparts in the future.

1. INTRODUCTION

The doubly constrained entropy trip distribution/assignment (DCETDA) model is to find proper O-D demands subject to fixed trips both generated at each origin and attracted at each destination, while Wardrop's first principle is fully complied by travelers in searching for their routes from origin to destination. This *combined* model may be decomposed for easier description into two interrelated submodules in sequence, i.e., trip distribution and traffic assignment problems. The former is identical to the constrained matrix problem which is to compute the best possible estimate of an unknown matrix, given some information to constrain the solution set, and requiring that the matrix be a minimum distance from a given matrix (Nagurney, 1993), whereas the latter is an user-optimal route choice problem which is well known to transportation planners.

The DCETDA model was formulated as an optimization problem and a double-stage solution algorithm, along with its convergence, was suggested by Evans (1976). The double-stage solution algorithm is a partial linearization technique for solving a nonlinear problem with a set of linear constraints. In some circumstances, the double-stage solution algorithm is also referred to as Evans algorithm to attribute her original work and to distinguish it from the Frank-Wolfe (FW) algorithm, the fully linearization technique. The computational advantages of Evans algorithm over the FW method for the trip distribution/traffic assignment problem have been demonstrated by many researchers. An intensive literature review was conducted by Boyce et al. (1984). The advantages of Evans algorithm over FW type algorithms *appear* to extend to any network equilibrium problem in which the origin-destination flows are determined by a function of network-related costs.

However, in a book by Sheffi (1985), he stated that the auxiliary problem associated with Evans algorithm is a nonlinear problem representing the gravity model, it is considerably more difficult to solve than Hitchcock's transportation problem, which has to be solved as part of the FW method (or the convex combinations method). Therefore, unlike the case with singly constrained models, the double-stage (Evans) algorithm has not apparent advantage over the convex combinations method, when both are applied to the solution of doubly constrained models. This statement obviously contradicts the one made by Boyce et al. (1986) and certainly suspicious as the gravity model can be efficiently solved by the RAS algorithm (Chibini et al., 1994).

In the following, the DCETDA model will be formulated and described in Section 2. The two solution algorithms, i.e., Evans and FW algorithms, are described in Section 3. A numerical example is provided in Section 4 and comparisons of computational efficiency are made in Section 5. Finally, concluding remarks are given in Section 6.

2. DOUBLY CONSTRAINED DISTRIBUTION/ASSIGNMENT PROBLEM

2.1 Entropy Model Formulation

Assuming all states are equally likely to occur, the entropy trip distribution model is to find the pattern with maximum number of states subject to the trip production and attraction constraints. On the other hand, the traffic assignment problem is to solve for equilibrium subject to fixed O-D demands, flow conservation, and nonnegativity constraints. A joint doubly constrained entropy distribution/assignment (DCETDA) problem can be formulated as follows:

$$\min z(\mathbf{x},\mathbf{q}) = \sum_a \int_0^{x_a} t_a(\omega)d\omega + \frac{1}{\zeta}\sum_{rs}\left(q^{rs}\ln q^{rs} - q^{rs}\right) \tag{1}$$

S.t.

$$\sum_p f_p^{rs} = q^{rs} \quad \forall r,s \tag{2}$$

$$\sum_s q^{rs} = \bar{q}^r \quad \forall r \tag{3}$$

$$\sum_r q^{rs} = \bar{q}^s \quad \forall s \tag{4}$$

$$f_p^{rs} \geq 0 \quad \forall r,s,p \tag{5}$$

Equation (1) is the objective function to be minimized consisting of the first term for route choice behavior by travelers and of the second term for the most likely O-D trip rate pattern, where symbol ζ is a dispersion parameter calibrated from data. Equation (2) conserves flows for each O-D pair. Production constraint (3) requires that summing O-D flows over all destinations results in trip rates from origin r. Similarly, attraction constraint (4) requires that summing O-D flows over all destinations results in trip rates attracted to destination s. Equation (5) states that route flows cannot be negative.

2.2 Optimality Conditions

A Lagrangian of the DECTDA problem can be formed by dualizing equations (2)~(4) using the corresponding dual variables u^{rs}, μ^r and λ^s.

$$L(\mathbf{x},\mathbf{q},u\mu,\lambda)=\sum_a\int_0^{x_a}t_a(\omega)d\omega+\frac{1}{\zeta}\sum_{rs}\left(q^{rs}\ln q^{rs}-q^{rs}\right)$$
$$+\sum_{rs}u^{rs}\left(q^{rs}-\sum_p f_p^{rs}\right)+\sum_r\mu^r\left(\bar{q}^r-\sum_s q^{rs}\right)+\sum_s\lambda^s\left(\bar{q}^s-\sum_r q^{rs}\right) \quad (6)$$

Taking derivative with respect to O-D flow q^{rs} results in:

$$\frac{1}{\zeta}\ln q^{rs}+c^{rs}-\mu^r-\lambda^s=0 \quad \forall r,s \quad (7)$$

By manipulation, the O-D flow can be expressed as follows:

$$q^{rs}=e^{-\zeta\left(c^{rs}-\mu^r-\lambda^s\right)} \quad \forall r,s \quad (8)$$

Substituting equation (8) into equations (3) and (4) yields the following two equations, respectively:

$$e^{\zeta\mu^r}=\frac{\bar{q}^r}{\sum_s e^{-\zeta\left(c^{rs}-\lambda^s\right)}} \quad \forall r \quad (9)$$

$$e^{\zeta\lambda^s}=\frac{\bar{q}^s}{\sum_r e^{-\zeta\left(c^{rs}-\mu^r\right)}} \quad \forall s \quad (10)$$

Let

$$A^r=\frac{e^{\zeta\mu^r}}{\bar{q}^r} \quad \forall r \quad (11)$$

$$B^s=\frac{e^{\zeta\lambda^s}}{\bar{q}^s} \quad \forall s \quad (12)$$

The O-D flow q^{rs} can be further expressed as follows:

$$q^{rs}=A^r\bar{q}^rB^s\bar{q}^se^{-\zeta c^{rs}} \quad \forall r,s \quad (13)$$

By equations (3), (4) and (13), the balancing parameters A^r and B^s can be expressed by each other as follows:

$$A^r=\frac{1}{\sum_s B^s\bar{q}^se^{-\zeta c^{rs}}} \quad \forall r \quad (14)$$

$$B^s=\frac{1}{\sum_r A^r\bar{q}^re^{-\zeta c^{rs}}} \quad \forall s \quad (15)$$

3. SOLUTION ALGORITHMS

The general algorithmic scheme for solving a nonlinear programming problem with linear constraints includes the steps of initialization, direction finding, line search, update, and convergence check, as follows:

General algorithmic scheme

Step 1: **Initialization**.

Let n=0. Find an initial solution $\left(\{q^{rs}\}^n, \{x_a\}^n\right)$.

Step 2: **Direction Finding**.

Search for a decent direction by solving a fully or partial linearized auxiliary problem based on the link travel times $\{t_a^n(x_a^n)\}$. Denote the solution of the auxiliary problem as $\left(\{v^{rs}\}^n, \{y_a\}^n\right)$.

Step 3: **Line Search**.

Determine the optimal move size α by performing the following line search.

$$\min_{0\le\alpha\le 1} \sum_a \int_0^{x_a^n+\alpha(y_a^n-x_a^n)} t_a(w)dw + \frac{1}{\zeta}\sum_{rs}\left[q^{rs^n}+\alpha\left(v^{rs^n}-q^{rs^n}\right)\right]\left\{\ln\left[q^{rs^n}+\alpha\left(v^{rs^n}-q^{rs^n}\right)\right]-1\right\} \tag{16}$$

Step 4: **Update**.

Update the O-D flows and link flows by the following formulas:

$$x_a^{n+1} = x_a^n + \alpha^n\left(y_a^n - x_a^n\right) \quad \forall a \tag{17}$$

$$q^{rs^{n+1}} = q^{rs^n} + \alpha^n\left(v^{rs^n} - q^{rs^n}\right) \quad \forall r,s \tag{18}$$

Step 5: **Convergence Check**.

If $\left|\dfrac{z^{n+1}-z^*}{z^*}\right| \le \varepsilon$ is satisfied, stop. Otherwise, let n=n+1 and go to Step 2.

In Step 5, the convergence criterion is set as the prespecified tolerance between the current and *exact* objective values. In this experiment, the exact objective value for each test network is obtained by performing the FW method for 3600 seconds. To compare computational efficiency, two solution algorithms, i.e., FW and Evans methods are of interest. These two solution algorithms share the same general algorithmic scheme but differ in the auxiliary problem of Step 2. For the FW method, the corresponding auxiliary problem (also known as Hitchcock's transportation problem) is derived using linear approximation technique, whereas Evans method adopts partial linearization approach for the auxiliary problem (also known as the matrix balancing problem).

3.1 Hitchcock's Transportation Problem

Hitchcock's transportation problem corresponding to our auxiliary problem can be formulated as follows:

$$\min z^n(\mathbf{g},\mathbf{v}) = \sum_{rs}\sum_{p} \frac{\partial z(\mathbf{f}^n)}{\partial f_p^{rs}} g_p^{rs} = \sum_{rs}\sum_{p}\left[c_p^{rs^n} + \frac{1}{\zeta}\ln q^{rs^n}\right] g_p^{rs} \tag{19}$$

S.t.

$$\sum_p g_p^{rs} = v^{rs} \quad \forall r,s \tag{20}$$

$$\sum_s v^{rs} = \bar{q}^r \quad \forall r \tag{21}$$

$$\sum_r v^{rs} = \bar{q}^s \quad \forall s \tag{22}$$

$$g_p^{rs} \geq 0 \quad \forall p,r,s \tag{23}$$

where decision variables $\{v^{rs}\}$, $\{g_p^{rs}\}$, in contrast with decision variables $\{f_p^{rs}\}$, $\{q^{rs}\}$ that were shown in the main problem, denote O-D flows and path flows for each O-D pair, respectively, for the auxiliary problem and symbol n indicates iteration number. This auxiliary problem cannot be decomposed by O-D pair unless the auxiliary O-D flows $\{v^{rs}\}$ are known. Let the travel time on the shortest path connecting r to s, at the nth iteration, be denoted by u^{rs^n}. The problem of finding $\{v^{rs}\}$ can now be written as follows:

$$\min z^n(\mathbf{v}) = \sum_{rs}\left[u^{rs^n} + \frac{1}{\zeta}\ln q^{rs^n}\right] v^{rs} \tag{24}$$

S.t.

$$\sum_s v^{rs} = \bar{q}^r \quad \forall r \tag{25}$$

$$\sum_r v^{rs} = \bar{q}^s \quad \forall s \tag{26}$$

$$v^{rs} \geq 0 \quad \forall r,s \tag{27}$$

The above problem is known in the operations research literature as Hitchcock's transportation problem. An adaptation of the Simplex method for linear programming is described to solve this problem as follows:

Step 1: Select an initial feasible solution with R+S-1 flow-carrying links.

Step 2: Check whether the solution can be improved by using a currently empty link. If not, stop, if yes, continue.

Step 3: Determine the amount of flow that can be assigned to the new link without violating any constraint.

Step 4: Adjust the flow on all other flow-carrying links and update the network. Go to step 2.

3.2 Matrix Balancing Problem

The matrix balancing problem corresponding to our auxiliary problem can be formulated as follows:

$$\min z(\mathbf{v}) = \sum_{rs} c^{rs} v^{rs} + \frac{1}{\zeta}\sum_{rs}\left(v^{rs}\ln v^{rs} - v^{rs}\right) \tag{28}$$

S.T.

$$\sum_{s} v^{rs} = \bar{q}^{r} \quad \forall r \tag{29}$$

$$\sum_{r} v^{rs} = \bar{q}^{s} \quad \forall s \tag{30}$$

$$v^{rs} \geq 0 \quad \forall r, s \tag{31}$$

In view of objective function (28), the nonnegativity constraint (31) is really not needed. Similar to those for the DECTDA model, the optimality conditions for the problem (28)~(31) can be derived by forming a Lagrangian as follows:

$$L(\mathbf{x}, \mathbf{q}, \mu, \lambda) = \sum_{rs} c^{rs} v^{rs} + \frac{1}{\zeta}\sum_{rs}\left(v^{rs}\ln v^{rs} - v^{rs}\right) + \sum_{r}\mu^{r}\left(\sum_{s} v^{rs} - \bar{q}^{r}\right) + \sum_{s}\lambda^{s}\left(\sum_{r} v^{rs} - \bar{q}^{s}\right) \tag{32}$$

Taking derivative with respect to O-D flow v^{rs} results in:

$$\frac{1}{\zeta}\ln v^{rs} + c^{rs} - \mu^{r} - \lambda^{s} = 0 \quad \forall r, s \tag{33}$$

By manipulation, the O-D flow can be expressed as follows:

$$v^{rs} = e^{-\zeta\left(c^{rs} - \mu^{r} - \lambda^{s}\right)} \quad \forall r, s \tag{34}$$

Substituting equation (34) into equations (29) and (30) yields the following two equations, respectively:

$$e^{\zeta\mu^{r}} = \frac{\bar{q}^{r}}{\sum_{s} e^{-\zeta\left(c^{rs} - \lambda^{s}\right)}} \quad \forall r \tag{35}$$

$$e^{\zeta\lambda^{s}} = \frac{\bar{q}^{s}}{\sum_{r} e^{-\zeta\left(c^{rs} - \mu^{r}\right)}} \quad \forall s \tag{36}$$

Let

$$A^{r} = \frac{e^{\zeta\mu^{r}}}{\bar{q}^{r}} \quad \forall r \tag{37}$$

$$B^{s} = \frac{e^{\zeta\lambda^{s}}}{\bar{q}^{s}} \quad \forall s \tag{38}$$

The O-D flow q^{rs} can be further expressed as follows:

$$v^{rs} = A^{r}\bar{q}^{r}B^{s}\bar{q}^{s}e^{-\zeta c^{rs}} \quad \forall r, s \tag{39}$$

By equations (29), (30) and (39), the balancing parameters A^{r} and B^{s} can be expressed by each other as follows:

$$A^r = \frac{1}{\sum_s B^s \bar{q}^s e^{-\zeta c^{rs}}} \qquad \forall r \tag{40}$$

$$B^s = \frac{1}{\sum_r A^r \bar{q}^r e^{-\zeta c^{rs}}} \qquad \forall s \tag{41}$$

The auxiliary problem (28)~(31) is then equivalent to the matrix balancing problem of equations (29), (30) and (39). Note that total trip conservation is implicitly hypothesized, i.e., $\sum_r \bar{q}^r = \sum_s \bar{q}^s = Q$. The RAS algorithm (Bachem and Korte, 1979) can then be described as follows:

RAS algorithm

Step 0: **Initialization**.

Set $A^{r^0} = 1$ and let i=0.

Step 1: **Balancing Columns**.

$$B^{s^i} = \left[\sum_r A^{r^i} \bar{q}^r e^{-\zeta c^{rs^n}}\right]^{-1} \qquad \forall s \tag{42}$$

Step 2: **Balancing Rows**.

$$A^{r^i} = \left[\sum_s B^{s^i} \bar{q}^s e^{-\zeta c^{rs^n}}\right]^{-1} \qquad \forall r \tag{43}$$

Step 3: **Stopping Test**.

If $\underset{r}{Max}\left|\dfrac{A^{r^i} - A^{r^{i-1}}}{A^{r^i}}\right| \le \varepsilon$ and $\underset{s}{Max}\left|\dfrac{B^{s^i} - B^{s^{i-1}}}{B^{s^i}}\right| \le \varepsilon$, go to Step 4; otherwise, set i=i+1 and return to Step 1.

Step 4: **Compute Solution**.

$$v^{rs^n} = A^{r^*} \bar{q}^r B^{s^*} \bar{q}^s e^{-\beta c^{rs^n}} \qquad \forall r,s \tag{44}$$

4. NUMERICAL EXAMPLE

4.1 Input Data

A simple network shown in Figure 1 is used for testing. This test network contains 6 links and 5 nodes, in which nodes 1 and 2 are the origins, nodes 4 and 5 are the destinations, and node 3 is an intermediate node.

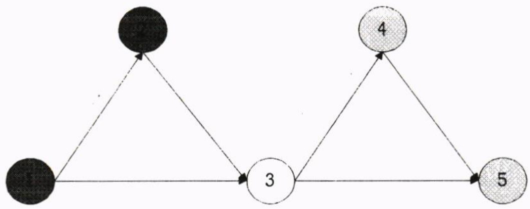

Figure 1: Test Network 1

The FHWA cost function is adopted throughout this paper and, without loss of generality, the dispersion parameter ζ is assumed to be 1. The total production trips at each origin and the attraction trips at each destination are hypothesized in Table 1.

Table 1: Production and Attraction Trips (Test Network 1)

	Origin		Destination	
	1	2	4	5
Trips	150	100	50	200

4.2 Test Results

A computer program coded with Borland C++5.01 was performed on a Pentium 166-MMX personal computer with 32M RAM. Both the FW and Evans methods result in identical flow pattern (Table 2), O-D demands (Table 3), production and attraction trips (Table 4), and route travel times (Table 5).

Table 2: Link Flows and Link Travel Times (Test Network 1)

Link	Link Flow	Link Travel Time
1→2	90.72	22.03
1→3	59.28	38.89
2→3	190.72	16.86
3→4	63.64	13.94
3→5	186.36	33.94
4→5	13.64	20.00

Table 3: O-D Demands (Test Network 1)

O-D Pair	Trips
1 - 4	30.00
1 - 5	120.00
2 - 4	20.00
2 - 5	80.00

Table 4: Trip Productions and Attractions (Test Network 1)

Ends	Computed Trips	Actual Trips
O_1	30.00+120.00=150.00	150
O_2	20.00+80.00=100.00	100
D_4	30.00+20.00=50.00	50
D_5	120.00+80.00=200.00	200

Table 5: Route Travel Times (Test Network 1)

Route	Route Travel Time	Route	Route Travel Time
1→3→4	52.83	2→3→4	30.80
1→2→3→4	52.83	2→3→5	50.80
1→3→5	72.83	2→3→4→5	50.80
1→3→4→5	72.83		
1→2→3→5	72.83		
1→2→3→4→5	72.83		

From Table 4, production and attraction constraints are satisfied. From Table 5, equilibrium conditions are also complied. The correctness of the computer code for the FW and Evans methods is thus verified.

5. COMPARISON OF COMPUTATIONAL EFFICIENCY

Two methods, i.e., FW and Evans methods are compared for four test networks in this section. The topology and associated network data for the first test network have been described in Section 4. The second test network is symmetric in shape (Figure 2). It consists of 10 links and 5 nodes, in which nodes 1, 2, 3 are the origins, nodes 5, 6, 7 are the destinations, and node 4 is an intermediate node. The associated productions and attractions are given in Table 6.

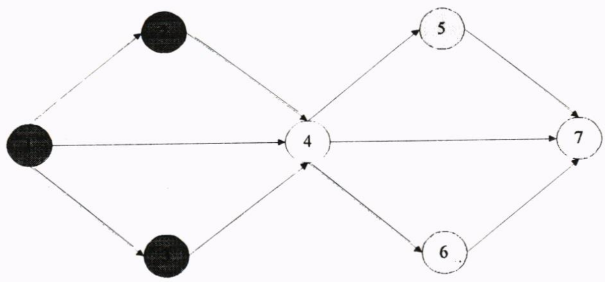

Figure 2: Test Network 2

Table 6: Productions and Attractions (Test Network 2)

	Origin			Destination		
	1	2	3	5	6	7
Trips	150	300	200	250	100	300

The third test network is essentially the U-town network (see Figure 3), which is taken from the FHWA publication (1986) with little modification. The network contains 130 links and 45 nodes, in which nodes 1 and 2 are the origins, nodes 3, 4 and 5 are destinations, the other nodes are intermediate. The associated productions and attractions are assumed in Table 7.

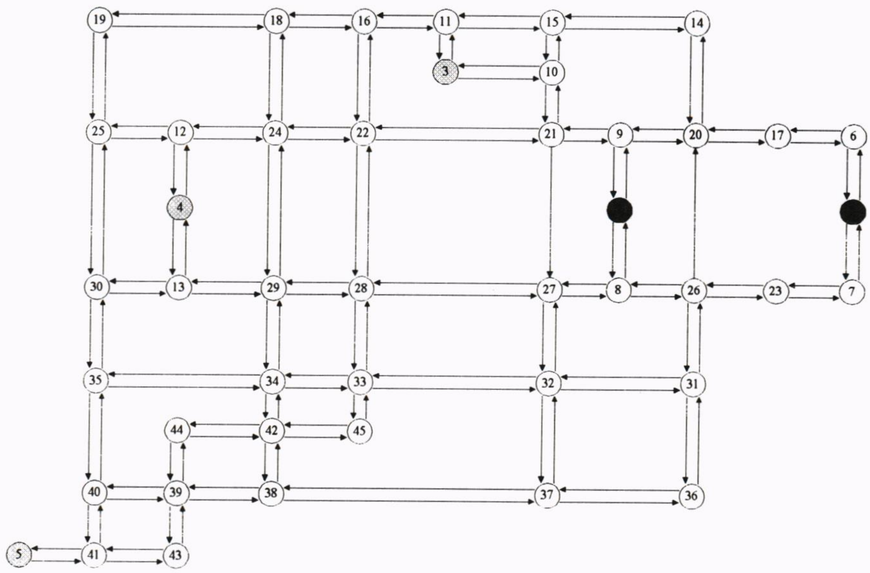

Figure 3: Test Network 3

Table 7: Productions and Attractions (Test Network 3)

	Origin		Destination		
	1	2	3	4	5
Trips	3000	4000	2000	3500	1500

The fourth network is a real network, which is constructed for Chungli-Pincheng urban area in Taoyuan county, Taiwan. Chungli-Pincheng is about 35 kilometers distant from the southeast boundary of Taipei. The area and population within the Chungli-Pincheng network are roughly 7 thousand acres and 300 thousand residents, respectively. The entire network is represented by 138 nodes and 449 links which are cast into 16 traffic zones. For demonstration purpose, we only use three origins (nodes 1, 2, and 3) and three destinations (nodes 137, 138 and 139). The associated productions and attractions are tabulated in Table 8.

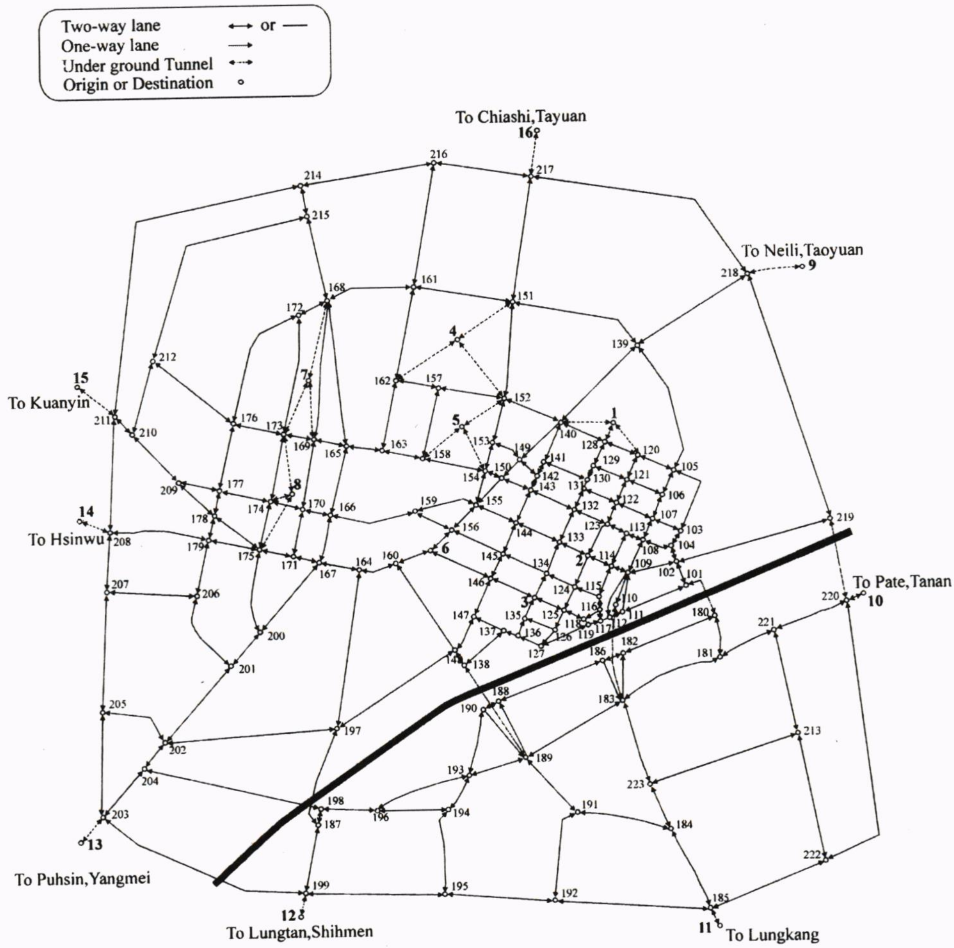

Figure 4: Test Network 4

Table 8: Productions and Attractions (Test Network 4)

	Origin			Destination		
	1	2	3	137	138	139
Trips	1550	2000	2450	960	3000	2040

The FW and Evans methods are then compared for these four test networks in terms of five performance measures, i.e., iterations for outer loops, objective value, total execution time, execution time for each outer loop, memory requirement. The results are summarized in Table 9.

Table 9: Comparisons of Performance Measures

Network Number	Performance Measure	FW Method	Evans Method
Test Network 1	Iterations for Outer Loops	143	145
	Objective Value	13705.184458	13705.184458
	Total Execution Time (sec)	0.250	0.205
	Execution Time for Each Outer Loop (sec)	0.0017	0.0014
	Memory Requirement (bytes)	3232	2020
Test Network 2	Iterations for Outer Loops	1116649	2
	Objective Value	53535.784855	53527.929181
	Total Execution Time (sec)	3600.002	0.004
	Execution Time for Each Outer Loop (sec)	0.0032	0.0020
	Memory Requirement (bytes)	5600	3464
Test Network 3	Iterations for Outer Loops	467071	41
	Objective Value	158337.603830	158334.863882
	Total Execution Time (sec)	3600.001	0.254
	Execution Time for Each Outer Loop (sec)	0.0077	0.0062
	Memory Requirement (bytes)	143300	81261
Test Network 4	Iterations for Outer Loops	21558	3
	Objective Value	920006.898934	919849.504949
	Total Execution Time (sec)	3600.142	0.264
	Execution Time for Each Outer Loop (sec)	0.166998	0.088
	Memory Requirement (bytes)	944876	590548

From Table 9, Evans method outperforms for all four test networks in terms of five performance measures. The fewer iterations for outer loops are due to closer approximation of the partial linear subproblem to the original main problem. For a network with N nodes, at each outer iteration, the number of O-D pairs assigned with positive values are $2N$-1 and $N(N$-1) for the FW and Evans methods, respectively. If we like to update each O-D pair exactly once, Evans method would need $N/2$ iterations. The shorter turn-around time is basically due to computational efficiency of RAS algorithm over Hitchcock algorithm. The lower memory requirement is probably because of simple adjustment operations for matrix column and row parameters, rather than a time-consuming tree spanning and flow augmentation process as in the network flow problem.

6. CONCLUSION AND SUGGESTIONS

In this paper, two methods, i.e., the FW and Evans algorithms are compared for the doubly constrained entropy user-optimal route choice problem. The latter one outperforms in terms of five performance measures. The shorter turn-around computation time is consistent with the results by Boyce (1984) but contradicts with the points made by Sheffi (1985). To consolidate the experiments, four issues should be further explored in the future.

1. Incorporating link capacity constraints into the model.
2. Exploring path based algorithms which have been proved more efficient somewhere else.
3. Incorporating temporal dimension into the model such that the dynamics of flow variations over time can be better represented.
4. Implementing tests with large real networks.

REFERENCES

1. Bachem A. and Korte B. 1979. On the RAS-algorithm, **Computing, 25**, 189-198.
2. Boyce D. E., 1984. Urban Transportation Network-Equilibrium and Design Models: Recent Achievement and Future Prospects, **Environment Planning, A16**, 1445-1474.
3. Boyce D.E. and Janson B.N. 1980. A discrete transportation network design problem with combined trip distribution and assignment. **Transportation Research, 14B**, 147-154.
4. Boyce D. E., LeBlanc, L. J. and Chon, K. S., 1988. Network equilibrium models of urban location and travel choices: a retrospective survey. **Journal of Regional Science, 28**, 159-183.
5. Chabini I. and Florian M. 1995. An Entropy Based Primal-Dual Algorithm for Convex and Linear Cost Transportation Problems. Report CRT-95-17, University of Montreal.
6. Chabini I., Drissi-Kaitouni O. and Florian M. 1994. Parallel Implementations of Primal and Dual Algorithms For Matrix Balancing, **Computational Techniques for Econometrics and Economic Analysis**, D. A. Belslev (ed.), Kluwer Academic Publishers, Netherlands, 173-185.
7. U.S. Department of Transportation. 1986. **Urban Transportation Planning System: UTOWN Case Study**, Washington, DC.
8. Nagurney A. 1993. **Network Economics : A Variational Inequality Approach,** Kluwer Academic Publishers, Massachusetts.
9. Schneider M.H. and Zenios S.A. 1990. A comparative study of algorithms for matrix balancing. **Operations Research, 38**, 439-453.
10. Sheffi Y. 1985. **Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods**, Prentice-Hall, INC., Englewood Cliffs, New Jersey.

NOTATIONS

Notations used in this paper are described as follows:

a : link designation
r : origin designation
s : destination designation
p : path designation
t_a : travel time on link a
u^{rs} : shortest path travel time between O-D pair rs
c_p^{rs} : travel time for path p from origin r to destination s
x_a : link flow for link a (main problem variable)
y_a : link flow for link a (auxiliary problem variable)
q^{rs} : traffic demand for O-D pair rs (main problem variable)
v^{rs} : traffic demand for O-D pair rs (auxiliary problem variable)
$\bar{q}^r$: fixed productions at origin r
$\bar{q}^s$: fixed attractions at destination s
α : step size