

Chung-Yung Wang and Jia-Wun Zhang

Department of Logistics Management, Management College of National Defense University, Taipei 11258, Taiwan, Republic of China

**Abstract:** This study focuses on estimating O-D (origin-destination) trip demand from link traffic flows. Equality relationship among link traffic flow, path flow, and O-D trip matrices are used to establish a linear equation system. Solution characteristics are analyzed based on the relationship between the rank of the link/path incidence matrix and column variables. And under the solution framework of conditional inverse matrices, a column exchange method and a path flow proportion method have been developed. Network testing results verify that the proposed methods yield good results.

Key words: O-D trip demand, conditional inverse matrix, column exchange method, path flow proportion method.

#### 1. Introduction

During the planning process of transportation, O-D (origin-destination) trip matrices are essential data. Their values as input enable route assignment procedures and promote understanding of the route selection behavior of road users in a road network. Further analysis and prediction regarding the traffic supply and demand on various road sections can be followed by planning performed then, and management tasks. Traditionally, trip O-D tables in a given vehicular network are obtained via highway users' surveys such as home survey, roadside interview, or license plate recording, which is very costly and may confront with problems of sampling bias or data recording errors. The increasing use and emplacement of VDs (vehicle detectors) in recent years enables direct observation of traffic flow on various road sections in the network. With proper methods, the traffic flow on different road sections will be able to reflect O-D transportation demands, enabling more active management of urban traffic planning. For this reason, the estimation of O-D trip demand from traffic flow has become a crucial research topic in recent years.

In general, the traffic flow observed in specific road sections equals the total path flow passing through the road section between all O-D pairs. As a result, O-D trip demand, path flow, and link traffic flow can be formed as a system of linear equations. With a good path generation tool for O-D pairs as well as matrix and inverse matrix computation, an appropriate analytical method can be developed to estimate trip demand using link traffic flow. Unfortunately, the inversed matrix does not exist if the target matrix is singular or non-squared. In this study, the O-D matrix estimation problem is formulated as a system of linear equations and solved by using the CIM (conditional inverse matrix) theory and its extended methods, column exchange method and path flow proportion method. The adopted matrix inverse method provides a generalized matrix inverse procedure even if the

**Corresponding author:** Chung-Yung Wang, Ph.D., associate professor, research fields: transportation planning, traffic assignment problems, network design problem, trip-chain network equilibrium model and solution algorithm. E-mail: stellar@ms35.url.com.tw.

target matrix is either singular or non-squared. Compared to the traditional network O-D demand estimation methods such as least squares method, bi-level programming method and PFE (path flow estimation method), the developed model is able to solve the network science problem under different flow distributions, whether the link flow is collected under a UE (user equilibrium) condition or not.

The remainder of this paper is organized as follows. In Section 2, we conduct a comprehensive investigation of related literature. Section 3 describes the models for the network O-D demand estimation problem and solution algorithms. Numerical analysis including experimental setup, solution methods and results are presented in Section 4. Finally, in Section 5, findings and conclusions of this research are summarized including suggestions for future research.

### 2. Literature Review and Problem Analysis

Since Robillard's [1] introduction of traffic flow data in the estimation of O-D trip matrices, the topic has become increasingly important in the field. Later, Cascetta [2] proposed a GLS (generalized least squares) method that minimized the squared error of O-D trips and link traffic flow and incorporated weight concepts. Based on this approach, Cascetta and Nguyen [3], Bell [4] and Doblas and Benitez [5] developed estimation models for transportation demand based on both GLS and link traffic flow.

Due to the fact that GLS-based estimation models for transportation demand do not consider the constraints of user equilibrium route choice behaviors, Yang [6], referred to the constrained GLS model presented by Bell [4], established a bi-level programming model that included these constraints. Maher et al. [7] and Lundgren and Peterson [8] worked on the development of similar models.

Sherali et al. [9] proposed the linear PFE model, which also estimates O-D trip matrices from available traffic flow data under the constraints of user equilibrium. However, due to incomplete data, the estimated traffic volume may be inconsistent with the observed link traffic flow; within an acceptable range of error, the model permits violations to the equilibrium conditions. To solve this problem, they proposed a column generation method. Subsequently, Bell and Shield [10], Bell et al. [11] and Chen et al. [12] all adopted the PFE to estimate transportation demand.

It can be noticed that previous approaches to estimate O-D transportation demand with link traffic flow can be divided into those that include user equilibrium constraints (PFE and bi-level programming models) and those that do not (GLS-based methods). With regard to the former group, if the equilibrium assignment of the inversed O-D transportation demand were to be performed anew, the flow patterns in the road network would have to conform to the user equilibrium results. However, the traffic flow in a network may not satisfy the condition of user equilibrium; if link traffic flow that does not conform to the user equilibrium principle is used to estimate O-D trip demand in bi-level programming models and the PFE, the resulting trip demand will display larger error. As for using GLS and link traffic flow to estimate O-D trip demand, additional information such as historical records on O-D trip demand and link traffic flow data obtained beforehand are crucial factors influencing the accuracy of the estimation. Nevertheless, this also implies that in the event of major changes in the O-D pairs of the study region or failure to obtain historical records on O-D trip demand, wider discrepancies may be shown between the O-D trip demand results estimated using GLS and the actual demand. Therefore, the estimated O-D trip demand is also worth discussing. Furthermore, even if the obtained historical records on O-D trip demand were extremely accurate, errors may still occur between the O-D trip matrices estimated using the model and the actual solution.

General forms of the linear equations used in

previous approaches are expressed in Eqs. (1) and (2).

$$\mathbf{q} = \mathbf{\Lambda}_1 \boldsymbol{h} \tag{1}$$

$$\boldsymbol{x} = \boldsymbol{\Lambda}_2 \boldsymbol{h} \tag{2}$$

where, **q** denotes the O-D trip demand matrix;  $\Lambda_1$  is the O-D trip demand/path incidence matrix; **h** is the path flow vector; **x** represents the link traffic flow vector;  $\Lambda_2$  signifies the link/path incidence matrix.

Bell and Shield [10], Bell et al. [11] and Chen et al. [12] all used forms of these equations. Gentili and Mirchandani [13] and Castillo et al. [14] further incorporated path indicators and partial path flows into the equations above to solve transportation demand, albeit with an extremely complex solution method.

In fact, Eqs. (1) and (2) can be used to derive O-D trip demand from link traffic flow through the inverse matrix computations given within Eqs. (3) and (4).

$$\boldsymbol{h} = \boldsymbol{\Lambda}_2^{-1} \boldsymbol{x} \tag{3}$$

$$\mathbf{q} = \mathbf{\Lambda}_1 \mathbf{\Lambda}_2^{-1} \mathbf{x} \,. \tag{4}$$

However, it is not guaranteed that link/path incidence matrix  $\Lambda_2$  will be a square matrix. In the event of a non-square matrix, the inverse does not exist, and in this case this method is inappropriate. Graybill [15] showed that conditional inverse matrices can be used to process inverse matrix problems with non-square matrices. According to the theory of conditional inverse matrices, Eqs. (3) and (4) can be inversed by Eqs. (1) and (2). That is, regardless of whether or not the link traffic flow conformed to user equilibrium, through the theory of conditional inverse matrices, the corresponding O-D trip matrices all should be obtained from link flows. It is a potential method to obtain the O-D trip matrices. In this study, we will explore the O-D trip matrices estimation problems, based on conditional inverse matrices theory.

# **3.** Conditional Inverse Matrix and Solution Algorithm

#### 3.1 Conditional Inverse Matrix

Graybill [15] showed that conditional inverse matrices can be used to process inverse matrix

problems with non-square matrices. During the computation of conditional inverse matrices, the full column rank condition does not have to be satisfied. As a result, unnecessary link traffic flow data are not eliminated due to rank annihilation when calculating the inverse of the link/path incidence matrix  $\Lambda_2$ . For relevant definitions and theorems for conditional inverse matrices, we refer to Graybill [15].

Definition 1: Let **A** be an  $m \times n$  matrix; **A**<sup>c</sup> is defined as the conditional inverse matrix of **A** if and only if **A**<sup>c</sup> satisfies **AA**<sup>c</sup> = **A**.

Definition 2: Let **H** be an  $n \times n$  matrix and be defined as of (upper) Hermite form if and only if it satisfies the four following conditions:

(1) **H** is an upper triangular matrix;

(2) The main diagonal elements are either 0 or 1;

(3) If a diagonal element is 0, then the elements of the entire row are 0;

(4) If a diagonal element is 1, then all of the non-diagonal elements in the column in which the 1 appears are 0.

• Theorem 1: The generalized inverse matrix of matrix **A** equals the conditional inverse matrix of matrix **A**; however, the conditional inverse matrix of matrix **A** does not necessarily equal the generalized inverse matrix of matrix **A**.

Note: The generalized inverse matrix  $A^g$  of matrix **A** should satisfy the four conditions below, we shall call  $A^g$  a generalized inverse of **A**, but the conditional inverse must only the Definition 1.

- (1)  $\mathbf{A}\mathbf{A}^{g}$  is symmetric;
- (2)  $A^{g}A$  is symmetric;
- (3)  $\mathbf{A}\mathbf{A}^{g}\mathbf{A} = \mathbf{A}$ ;
- (4)  $\mathbf{A}^{g}\mathbf{A}\mathbf{A}^{g} = \mathbf{A}^{g}$ .

• Theorem 2: A conditional inverse matrix exists for each matrix, but it is not necessarily the only conditional inverse matrix.

• Theorem 3: If **A** is an  $m \times n$  matrix, then the conditional inverse matrix is an  $n \times m$  matrix.

• Theorem 4: If **H** is in Hermite form, then  $\mathbf{H} = \mathbf{H}^2$ .

• Theorem 5: For all  $n \times n$  matrices **A**, a nonsingular matrix **B** exists. Furthermore, **BA** = **H**, where **H** satisfies the Hermite form.

• Theorem 6: Let **A** be an  $n \times n$  matrix, **B** is a nonsingular matrix, and **BA** = **H**, where **H** satisfies the Hermite form. Then **B** is the conditional inverse matrix of **A**.

• Theorem 7: For the conditional inverse matrix  $\mathbf{A}^{c}$  of any given  $m \times n$  matrix, matrices  $\mathbf{A}^{c}\mathbf{A}$  and  $\mathbf{A}\mathbf{A}^{c}$  are idempotent.

• Theorem 8: Let **A** be any given  $m \times n$  matrix. Then  $\mathbf{A}^{C} = (\mathbf{A}^{T}\mathbf{A})^{C}\mathbf{A}^{T}$ , where  $(\mathbf{A}^{T}\mathbf{A})^{C}$  is the conditional inverse matrix of any  $\mathbf{A}^{T}\mathbf{A}$ .

• Theorem 9: Let **A** be a known  $m \times n$  matrix,  $\mathbf{A}^c$  is the conditional inverse matrix of **A**. Suppose a solution exists for  $\mathbf{A}\boldsymbol{u} = \boldsymbol{g}$ , then for each  $n \times l$  vector  $\boldsymbol{k}$ , vector  $\boldsymbol{u}$  has the solution  $\boldsymbol{u} = \mathbf{A}^c \boldsymbol{g} + (\mathbf{I} - \mathbf{A}^c \mathbf{A})\boldsymbol{k}$ .

• Theorem 10: **A** is an  $m \times n$  matrix, for the rank of the conditional inverse matrix  $\mathbf{A}^{C}$  of any given **A**, the following are true: the rank of **A** equals the rank of  $\mathbf{A}^{c}\mathbf{A}$  as well as the rank of  $\mathbf{A}\mathbf{A}$  and is less than the rank of  $\mathbf{A}^{C}$ .

To obtain the conditional inverse matrix **B** of  $m \times n$ matrix A, we can extend Theorem 6 and add a 0 matrix to matrix A to form a square matrix  $A_0$  before establishing the augmented matrix of  $A_0$  and identity matrix I,  $[A_0|I]$ . Then, using the Gauss-Jordan elimination method for matrix computation, we can eliminate  $A_0$  on the left to form a matrix  $H_0$ satisfying the conditions of Hermite form. In the augmented matrix  $|\mathbf{H}_{0}|\mathbf{B}_{0}|$  resulting from row computation,  $\mathbf{B}_0$  is the conditional inverse matrix of  $A_0$ . Moreover, the Hermite form H and the conditional inverse matrix B of A are encompassed within  $H_0$  and  $B_0$ . Below, we describe the circumstances in which the number of rows is greater than the number of columns (m > n) in matrix A and also those in which the number of rows is less than the number of columns (m < n).

In the event that **A** is an  $m \times n$  matrix and m > n, an

 $m \times (m - n)$  **0** matrix is added to **A** to form  $m \times m$ square matrix **A**<sub>0</sub>, where **A**<sub>0</sub> = [**A 0**]. Using Gauss-Jordan elimination, augmented [**A 0**|**I**]

becomes 
$$\begin{bmatrix} \mathbf{H}_0 | \mathbf{B}_0 \end{bmatrix} = \begin{bmatrix} \mathbf{H} & \mathbf{0} | \mathbf{B} \\ \mathbf{0} & \mathbf{0} | \mathbf{B}_1 \end{bmatrix}$$
, where **H** is the

Hermite form of matrix **A** and is an  $n \times n$  matrix. Furthermore, **B** is the conditional inverse matrix of **A** and an  $n \times m$  matrix, and there exists a relationship between the matrices as expressed in Eq. (5).

$$\mathbf{H}_{0} = \mathbf{B}_{0}\mathbf{A}_{0} = \begin{bmatrix} \mathbf{B} \\ \mathbf{B}_{1} \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{B}\mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{H} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} (5)$$

In the event that **A** is an  $m \times n$  matrix and m < n, an  $(n - m) \times n$  **0** matrix is added to **A** to form  $n \times n$ square matrix **A**<sub>0</sub>, where  $\mathbf{A}_0 = \begin{bmatrix} \mathbf{A} \\ \mathbf{0} \end{bmatrix}$ . Using Gauss-Jordan elimination, augmented  $\begin{bmatrix} \mathbf{A} \\ \mathbf{0} \end{bmatrix} \mathbf{I}$ 

becomes  $[\mathbf{H}|\mathbf{BB}_1]$ , where **H** is the Hermite form of matrix **A** and is an  $n \times n$  matrix. Furthermore, **B** is the conditional inverse matrix of **A** and an  $n \times m$  matrix, and there exists the relationship expressed in Eq. (6).

$$\mathbf{H}_{0} = \mathbf{B}_{0}\mathbf{A}_{0} = \begin{bmatrix} \mathbf{B} & \mathbf{B}_{1} \end{bmatrix} \begin{bmatrix} \mathbf{A} \\ \mathbf{0} \end{bmatrix} = \mathbf{B}\mathbf{A} = \mathbf{H}.$$
 (6)

Regardless of whether or not matrix A is a square matrix, we can use the approach above to obtain the conditional inverse matrix  $A^{C}$ . Thus, when solving linear equations, the conditional inverse matrix approach is considerably more flexible than conventional inverse matrix methods.

# 3.2 Application of Conditional Inverse Matrix to Solve Origin-Destination Trip Matrix

To solve the linear equation system expressed by Eqs. (1) and (2), we can obtain the following inverse matrix conversion relationship based on conditional inverse matrix calculations:

$$\boldsymbol{h} = \boldsymbol{\Lambda}_2^C \boldsymbol{x} \tag{7}$$

$$\mathbf{q} = \mathbf{\Lambda}_1 \mathbf{h} = \mathbf{\Lambda}_1 \mathbf{\Lambda}_2^C \mathbf{x} \,. \tag{8}$$

Based on the conditional inverse matrix relationships in Eqs. (7) and (8), we can use the conditional inverse matrix of the link traffic flow and link/path incidence matrix to derive the O-D trip matrix. The solution procedure are as follows.

Step 1: Employ column generation to calculate the feasible paths between each O-D pair.

Step 2: Establish the O-D pair/path incidence matrix  $\Lambda_1$  and the link/path incidence matrix  $\Lambda_2$ .

Step 3: Calculate the conditional inverse matrix of  $\Lambda_2$ .

Step 3.1: Input link traffic flow, the link/path incidence matrix, and the O-D-pair/path incidence matrix.

Step 3.2: Based on Definition 2 and the four corresponding conditions, perform matrix computation and solve the Hermite form of  $[\Lambda_2 | \mathbf{I}]$  and conditional inverse matrix  $[\mathbf{H} | \mathbf{\Lambda}_2^C]$ , where  $\mathbf{\Lambda}_2^C$  is the conditional inverse matrix of  $\mathbf{\Lambda}_2$ .

Step 4: Substitute all the conditional inverse matrices obtained into Eq. (8) to derive the O-D trip matrix.

Using the approach described above, an accurate O-D trip matrix can be estimated from link traffic flow regardless of whether or not the flow patterns in each link of the road network conform to user equilibrium conditions. Nevertheless, in the event of a significant number of O-D pairs, inaccurate or infeasible solutions may be derived. To address this issue, we further prove the conditional inverse matrix relationships in Eqs. (7) and (8) as follows.

Proof: Let  $\mathbf{x} = \mathbf{\Lambda}_2 \mathbf{h}_c$  and  $\mathbf{\Lambda}_2 \neq \mathbf{0}$ 

$$\boldsymbol{\Lambda}_{2}\boldsymbol{\Lambda}_{2}^{C}\boldsymbol{x} = \boldsymbol{\Lambda}_{2}\boldsymbol{\Lambda}_{2}^{C}\boldsymbol{\Lambda}_{2}\boldsymbol{h}_{c}. \tag{9}$$

Based on Definition 1, we can derive the following using Eq. (8):

$$\boldsymbol{\Lambda}_2\boldsymbol{\Lambda}_2^C\boldsymbol{x} = \boldsymbol{\Lambda}_2\boldsymbol{h}_c.$$

Therefore,  $h_c$  is a solution of  $\Lambda_2^C x$ , and  $\Lambda_2 \Lambda_2^C x = x$ .

Let  $h_c$  be a solution of  $\Lambda_2^C x$ ; thus,  $h_c = \Lambda_2^C x$ .

By multiplying both sides of the equation by  $\Lambda_2^C$ , we can obtain

$$\boldsymbol{\Lambda}_{2}\boldsymbol{h}_{c} = \boldsymbol{\Lambda}_{2}\boldsymbol{\Lambda}_{2}^{C}\boldsymbol{x} = \boldsymbol{x}.$$

Therefore, Eq. (7)  $\boldsymbol{h}_c = \boldsymbol{\Lambda}_2^C \boldsymbol{x}$  is true, and we can derive Eq. (8)  $\boldsymbol{q} = \boldsymbol{\Lambda}_1 \boldsymbol{h}_c = \boldsymbol{\Lambda}_1 \boldsymbol{\Lambda}_2^C \boldsymbol{x}$ , proof completed.

From Theorem 2 and the proof above, we can see that there may be more than one solution for the conditional inverse matrix. For this reason,  $h_c$  is a solution of  $\Lambda_2^C x$  rather than the only solution. According to our tests, the Hermite form of link/path incidence matrix  $\Lambda_2$  is an identity matrix, an accurate solution can be estimated; however, when the Hermite form of  $\Lambda_2$  is not an identity matrix, a solution cannot be accurately estimated, and issues of negative path solutions may arise. Gentili and Mirchandani [13] and Castillo et al. [14] also indicated that in the relationship among O-D pairs, paths, and links, the number of links is generally smaller than the number of paths, thereby creating the problem of multiple solutions. In order to accurately estimate O-D transportation demand from link traffic flow, we analyzed the characteristics of the link/path incidence matrix  $\Lambda_2$  as follows.

When solving the linear equation Ax = b, the rank and number of variables in the equation influence the results, which may be only one solution, multiple solutions, or no solutions. During the solution process, Gauss-Jordan elimination is generally applied to simplify matrix A into a simpler matrix **R**, which is referred to as the row reduced echelon form of matrix A. If A is an  $m \times n$  matrix, then the corresponding **R** is likewise an  $m \times n$  matrix. Generally speaking, in the linear system Ax = b, only one solution exists when A is an  $m \times n$  matrix and the rank of row of matrix A(r) equals n; in contrast, the system has multiple solutions when r <*n*. Therefore, the rank of row for  $m \times n$  matrix A can be derived using **R**, the RREF (row reduced echelon form) of Ax = b.

Furthermore, the RREF must satisfy three

conditions [16]:

(1) All of the non-zero rows are above the all-zero rows. In other words, the all-zero rows are at the bottom of the matrix;

(2) The first non-zero term in each row is the only non-zero term in the entire row;

(3) The coefficient of the first non-zero term in each row is 1, and it is in a column further to the right than the coefficient of the first term in the row above it.

Consideration of Definition 3 of the conditional inverse matrices and the three conditions for the RREF reveal that Hermite form **H** possesses the characteristics of RREF **R**. As a result, there are two possible scenarios in this model:

(1) If the Hermite form obtained from  $\Lambda_2$  after the use of Gauss-Jordan elimination is identity matrix **I**, then the rank of row *r* for  $\Lambda_2$  equals the number of column variables *n*. Moreover, solution *h* of  $\Lambda_2 h = x$  is the only solution, and the solution **q** of  $\mathbf{q} = \Lambda_1 \Lambda_2^c \mathbf{x}$  is the only solution, too;

(2) If the Hermite form obtained from  $\Lambda_2$  after the use of Gauss-Jordan elimination is not identity matrix **I**, then the rank of row *r* for  $\Lambda_2$  is less than the number of column variables *n*. There are thus multiple solutions for solution *h* of  $\Lambda_2 h = x$ , therefore  $\mathbf{q} = \Lambda_1 \Lambda_2^c x$  possesses multiple solutions.

The inferences above indicate that only when rank r equals the path number n in Eq. (2) is there a unique path solution and thus an accurate reflection of O-D demand. The accuracy of applying conditional inverse matrices to estimate O-D demand depends on whether there is a unique path solution. If there is more than one path solution, infeasible solutions with negative path flows may be produced. To ensure that the path solution in this approach is feasible and unique, we propose a column exchange method and a path flow proportion method under the solution framework of the conditional inverse matrix method. In doing so, we aim to rectify the shortcomings of the conditional inverse matrix method.

#### 3.3 Column Exchange Method

Generating the paths between O-D pairs can be achieved using an approach of column generation method. Usually, the number of feasible paths generated increases with the number of O-D pairs. This causes the number of path variables to be larger than the link variables. Moreover, in the link/path incidence matrix  $\Lambda_2$ , the column (path) variables will be larger than the rank of row (link) variables, thereby allowing multiple solutions for  $\Lambda_2 h = x$  (Eq. (2)). In other words, a number of solutions will exist for O-D demand. In the event that additional network data cannot be obtained, at the least, the obtained solutions must be feasible. That is, the path flow solutions and the O-D trip demands obtained must be feasible solutions. Under the structure of conditional inverse matrices, we develop a column exchange method to obtain this answer to estimate a feasible O-D trips matrix.

Generally, in linear programming multiple solutions will occur when solving simultaneous equations if there are more variables than equations. In this case, the variables equal to the number of equations are set as the basic variables, and those that remain are referred to as non-basic variables. The non-basic variables are set as 0, and the solutions to the equations actually consist of the basic variables. Similarly, when the number of row rank is less than the column number in the link/path incidence matrix  $\Lambda_2$ , the structure of the column variables  $\Lambda_2$  can be divided into basic variables and non-basic variables. If the condition that all the basic variables are positive values can be satisfied, then it is referred to as a feasible solution; if there are negative values among the basic variables, then the solution is infeasible. In the event of infeasible basic variables, they are exchanged with the non-basic variables until a feasible solution is produced. This is the principle of the column exchange method.

So how do we determine that the path numbers generated are basic variables? This can be established

by observing the Hermite form. If the diagonal element is 1, then a value exists for the path number corresponding to the column in the Hermite form matrix, which is also the basic variable column. Furthermore, the rank of the matrix r equals the number of basic variables. In contrast, if the diagonal element is 0, then the path number corresponding to the column in the Hermite form matrix equals 0, and the column is the non-basic variable column. Observation of the Hermite form matrix also shows that during the process of Gauss-Jordan elimination, the majority of the diagonal elements at the rear of the matrix are eliminated, thereby becoming 0 and forming the non-basic variable column. When there are too many variables, multiple solutions will occur. To obtain different combinations of path solutions, we can exchange the basic variables with the non-basic variables and re-execute Gauss-Jordan elimination with the basic variables to be exchanged at the rear of the matrix.

The process of applying the column exchange method to an approach utilizing conditional inverse matrices is as follows.

Step 1: Generate feasible paths using column generation and construct O-D-pair/path incidence matrix  $\Lambda_1$  and link/path incidence matrix  $\Lambda_2$ .

Step 2: Apply Gauss-Jordan elimination to the augmented matrix,  $\begin{bmatrix} \Lambda_2 & 0 | I \end{bmatrix}$  or  $\begin{bmatrix} \Lambda_2 \\ 0 & I \end{bmatrix}$ , of link/path incidence matrix  $\Lambda_2$  to create  $\begin{bmatrix} H | \Lambda_2^c \end{bmatrix}$  and obtain the Hermite form **H** and conditional inverse matrix  $\Lambda_2^c$  of  $\Lambda_2$ .

Step 3: Multiple  $\Lambda_2^c$  by link traffic flow x to obtain path flow h.

Step 4: Determine whether negative flows exist in the path flow solutions: If a negative flow exists, execute Step 5. Otherwise, execute Step 6.

Step 5: Identify the basic variable columns and non-basic variable columns using **H**. Move the basic variable columns with negative path solutions to the rear of  $\Lambda_2$  to execute the column exchange.

Recalculate  $\Lambda_1$  and  $\Lambda_2$ , and then return to Step 2.

Step 6: Multiply  $\Lambda_1$  by path flow h, which now consists of positive path flows, and derive O-D trip demand  $\mathbf{q}$ .

Unlike mathematical programming, non-negative conditions can not be imposed in the constraints in the conditional inverse matrix approach. Consequently, negative values may appear in the solutions yielded. Nevertheless, this can be rectified through column exchange. This involves moving the infeasible basic variable columns obtained in the conditional inverse matrix method to the rear of matrix  $\Lambda_2$  before re-solving the problem. The purpose of this is to change the basic variables that were originally negative into non-basic variables (from negative values to 0) in the next round, thereby remedying the problem of negative flow in the overall path combination. Iterating the process will then produce a feasible solution.

#### 3.4 Path Flow Proportion Method

Despite being able to derive feasible path solutions, the column exchange approach still cannot guarantee to obtain a unique solution. In previous studies, to obtain a unique solution, many researchers were forced to include more road network data in their models to narrow the region of feasible solutions. The studies conducted by Gentili and Mirchandani [13] and Castillo et al. [14] both required path indices and partial path flow data to solve O-D transportation demand. However, the development of automated license plate scanning systems aided by intersection turning surveys and path usage sampling surveys have enabled researchers to obtain the proportions of path usage between O-D pairs. As a result, considering the data regarding path usage proportions between O-D pairs is theoretically reasonable.

Under the basic assumption that the path usage proportions between O-D pairs are known, we developed a path flow proportion method to increase the accuracy of O-D pair solutions. Path proportion

refers to the proportion of a given path flow between an O-D pair out of the total demand between said O-D pair and can be expressed using the path/O-D pair proportion matrix **P**. Table 1 presents path/O-D pair proportion matrix **P**, where in the rows, it is shown that Paths 1 and 2 account for proportions of 0.7 and 0.3 in O-D Pair 1, and that Paths 3 and 4 account for proportions of 0.4 and 0.6 in O-D Pair 2. In the columns, it is shown that when  $\Lambda_2$  is multiplied by **P**, Paths 1 and 2 are replaced by O-D Pair 1, and Paths 3 and 4 are replaced by O-D Pair 2.

Multiplying  $\Lambda_2$  by the known path/O-D pair proportion matrix **P** will produce the link/O-D pair proportion incidence matrix  $\Lambda_3$ . The proportional relationship between O-D demand and link traffic flow is presented in Eqs. (10) and (11).

$$\mathbf{\Lambda}_2 \mathbf{P} = \mathbf{\Lambda}_3 \tag{10}$$

$$\mathbf{\Lambda}_{3}\mathbf{q} = \mathbf{x} \tag{11}$$

The purpose of including link proportions in the solution process is to directly replace the path variables between an O-D pair with the O-D pair variable using the known proportions and replace the link/path incidence matrix  $\Lambda_2$  with the link/O-D pair proportion incidence matrix  $\Lambda_3$ . Such an approach will significantly reduce the number of variables in the matrices, enabling the rank to equal the number of variables and thus producing a unique solution. By solving Eq. (11) with the conditional inverse matrix method, we can obtain the desired O-D demand.

$$\mathbf{q} = \mathbf{\Lambda}_3^C \mathbf{x} \tag{12}$$

The steps to the solution method integrating the conditional inverse matrix and path flow proportion methods to obtain O-D demand are as follows.

Step 1: Use known paths to construct the link/path incidence matrix  $\Lambda_2$ .

Step 2: Multiply  $\Lambda_2$  by the path proportion matrix **P** to obtain the link/O-D pair incidence matrix  $\Lambda_3$ .

Step 3: Apply Gauss-Jordan elimination to the augmented matrix of link/O-D pair incidence matrix  $\Lambda_3$ ,  $[\Lambda_3 \ 0 | \mathbf{I}]$ , to obtain the Hermite form **H** and conditional inverse matrix  $\Lambda_3^C$  of  $\Lambda_3$ .

Step 4: Multiply  $\mathbf{\Lambda}_3^C$  by link traffic flow  $\mathbf{x}$  to derive O-D demand  $\mathbf{q}$ .

#### 4. Numerical Example

This study proposes two approaches to modify methods using conditional inverse matrices to estimate O-D demand from link traffic flow: The first utilizes column exchange and the second path flow proportion. Using the road network used by Yang [6] as a numerical example test (Fig. 1), in this section we demonstrate the procedures of the proposed methods and their effectiveness. The network comprises four O-D pairs with demands as set in Table 2, path flows as shown in Table 3 and link traffic flows as presented in Table 4.

#### 4.1 Column Exchange Method

In conditions where the path proportion data are unknown, suppose that the link traffic flows in the network have already been obtained via VDs but that path information between O-D pairs has not been derived from users. As a result, column generation must first be employed to create feasible paths before



Fig. 1 Test Network 1 [6].

Table 1 Path/O-D pair proportion matrix.

No	O-l	D Pair 1	O-D Pair 2						
INO.	Path 1	Path 2	Path 3	Path 4					
Path 1	0.7	0	0	0					
Path 2	0.3	0	0	0					
Path 3	0	0	0.4	0					
Path 4	0	0	0.6	0					

No.	O-D pair	Demand
1	1→3	200
2	1→4	150
3	2→3	140
4	2→4	185

 Table 2
 The O-D demands of Test Network 1.

Table 3 Set path flows of Test Netwo
--------------------------------------

No.	Path	Flow	No.	Path	Flow
1	1→5→3	200	9	$2 \rightarrow 7 \rightarrow 8 \rightarrow 5 \rightarrow 3$	87
2	$1 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 3$	0	10	$2 \rightarrow 6 \rightarrow 8 \rightarrow 5 \rightarrow 3$	40
3	$1 \rightarrow 7 \rightarrow 8 \rightarrow 5 \rightarrow 3$	0	11	$2 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 3$	0
4	$1 \rightarrow 5 \rightarrow 8 \rightarrow 9 \rightarrow 3$	0	12	$2 \rightarrow 6 \rightarrow 8 \rightarrow 9 \rightarrow 3$	13
5	$1 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 4$	100	13	2→6→4	126
6	$1 \rightarrow 5 \rightarrow 8 \rightarrow 6 \rightarrow 4$	30	14	$2 \rightarrow 6 \rightarrow 8 \rightarrow 9 \rightarrow 4$	59
7	$1 \rightarrow 7 \rightarrow 8 \rightarrow 6 \rightarrow 4$	20	15	$2 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 4$	0
8	$1 \rightarrow 5 \rightarrow 8 \rightarrow 9 \rightarrow 4$	0	16	$2 \rightarrow 7 \rightarrow 8 \rightarrow 6 \rightarrow 4$	0

 Table 4
 Set link traffic flows of Test Network 1.

No.	Link	Flow	No.	Link	Flow
1	1-5	230	8	6-8	112
2	1-7	120	9	7-8	207
3	2-6	238	10	8-5	127
4	2-7	87	11	8-6	50
5	5-3	327	12	8-9	172
6	5-8	30	13	9-3	13
7	6-4	176	14	9-4	159

the approach combining conditional inverse matrices with column exchange can be applied to yield a feasible solution for the O-D trip matrix.

Step 1: Using column generation, we generated feasible paths and constructed O-D-pair/path

incidence matrix  $\Lambda_1$  and link/path incidence matrix  $\Lambda_2$ , as shown in Tables 5 and 6. Using MATLAB, we calculated the rank of  $\Lambda_2$ , which was 9 and equal to the number of paths with flow.

Step 2: According to the method described in Section 3.2, we applied Gauss-Jordan elimination to the augmented matrix of link/path incidence matrix

$$\mathbf{\Lambda}_2$$
,  $\begin{bmatrix} \mathbf{\Lambda}_2 & \mathbf{0} | \mathbf{I} \end{bmatrix}$  or  $\begin{bmatrix} \mathbf{\Lambda}_2 \\ \mathbf{0} \end{bmatrix} \mathbf{I}$ , which became  $[\mathbf{H} | \mathbf{\Lambda}_2^C]$ ,

and derived the Hermite form **H** and conditional inverse matrix  $\mathbf{\Lambda}_2^C$  of  $\mathbf{\Lambda}_2$  as shown in Eqs. (13) and (14).

Step 3: We multiplied  $\Lambda_2^C$  by link traffic flow x to obtain path flow h.

Step 4: We determined whether negative flows existed in the path flow solutions, which they did (the 1st iteration in Table 7), and then executed Step 5.

Step 5: We identified the basic variable columns and non-basic variable columns using **H**. Then, we moved the basic variable columns with negative path solutions to the rear of  $\Lambda_2$  to execute column exchange. We recalculated  $\Lambda_1$  and  $\Lambda_2$ , as shown in Tables 8 and 9. The gray columns indicate the negative path flows that have been moved to the rear of the matrix.

Step 6: After iterating the process three times, consistently positive path flows were derived, as shown

		(1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 )
		0	1	0	0	0	0	1	-1	0	0	1	1	0	0	0	1
		0	0	1	0	0	0	0	0	0	0	$^{-1}$	$^{-1}$	0	-1	$^{-1}$	-1
		0	0	0	1	0	0	$^{-1}$	1	0	0	0	0	0	0	0	-1
		0	0	0	0	1	0	0	1	0	0	0	0	0	1	1	0
		0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Н	=	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	1
		0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	(	1	-1	0	-1	0	-1	0	0	1	0	0	0	0	0)
		0	0	0	0	0	0	0	1	1	-1	0	-1	1	0
		-1	6	2	5	1	3	-2	-1	-4	-2	0	-2	0	0
		0	2	1	2	0	1	-1	-1	-2	0	0	0	0	0
		1	-5	-2	-5	-1	-3	2	0	3	3	0	3	-1	0
		0	-1	-1	-1	0	0	1	1	1	0	0	0	0	0
		0	1	0	1	0	0	0	0	-1	0	0	0	0	0
A c	_	0	2	1	2	0	1	-1	0	-1	-1	0	-1	0	0
$\mathbf{n}_2$	_	0	-1	0	0	0	0	0	0	1	0	0	0	0	0
		0	-2	-1	-2	0	-1	1	1	1	1	0	1	0	0
		0	1	1	1	0	0	-1	-1	-1	0	1	0	0	0
		-1	3	1	3	1	2	-1	0	-2	-2	0	-1	0	0
		0	0	1	0	0	0	0	-1	0	0	0	0	0	0
		-1	5	2	5	1	3	-2	0	-3	-3	0	-3	1	1
		0	0	0	0	0	0	0	0	0	0	0	0	0	0
	l	0	0	0	0	0	0	0	0	0	0	0	0	0	0)

in Table 7. We then multiplied  $\Lambda_1$  by the consistently positive path flows *h* and derived O-D demand **q**, as presented in Table 10.

The results in Table 7 display negative path flows in the 1st and 2nd iterations, which indicate infeasible path solutions. However, during the process of column exchange, the number of negative path flows reduced; by the 3rd iteration, all of the path flows derived were positive and feasible solutions, thereby reaching convergence. At this point, the estimated demand values were identical to the preset values (shown in Table 10).

Table 5 O-D pair/path incidence matrix  $\Lambda_1$  of Test Network 1.

0 D main	No								Р	ath No.							
O-D pair	INO.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1-3	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
1-4	2	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
2-3	3	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
2-4	4	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

Table 6 Li	nk/path incidence	matrix $\Lambda_2$ of	Test Network 1.
------------	-------------------	-----------------------	-----------------

0 Davia	No.								Pa	ath No.							
O-D pair	INO.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1-5	1	1	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
1-7	2	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0
2-6	3	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0
2-7	4	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	1
5-3	5	1	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0
5-8	6	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
6-4	7	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	1
6-8	8	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0
7-8	9	0	1	1	0	1	0	1	0	1	0	1	0	0	0	1	1
8-5	10	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0
8-6	11	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1
8-9	12	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0
9-3	13	0	1	0	1	0	0	0	0	0	0	1	1	0	0	0	0
9-4	14	0	0	0	0	1	0	0	1	0	0	0	0	0	1	1	0

Path No.	Preset flow	1st iteration	2nd iteration	3rd iteration
1	200	200	200	200
2	0	33	-59	0
3	0	-72	0	0
4	0	-20	0	0
5	100	159	159	100
6	30	50	30	30
7	20	0	20	20
8	0	0	0	0
9	74	87	15	74
10	53	112	112	53
11	13	0	72	13
12	0	0	0	0
13	126	126	126	126
14	59	0	0	59
15	0	0	0	0
16	0	0	0	0

Table 7Iteration of path flows estimated by columnexchange method.

4.2 Path Flow Proportion Method

In conditions where the path proportion data are known, suppose that the link traffic flows in the

network are known and that path proportion data have been obtained via automated license plate scanning systems, intersection turning surveys, and inspection visits. We can then apply the approach combining conditional inverse matrices and the path flow proportion method to estimate O-D demand.

Step 1: We used known path information to construct the link/path incidence matrix  $\Lambda_2$ . The path proportion matrix **P** is presented in Table 11.

Step 2: By multiplying  $\Lambda_2$  by the path proportion matrix **P**, we obtained the link/O-D pair proportion incidence matrix  $\Lambda_3$ , as shown in Table 12. Using MATLAB, we calculated the rank of  $\Lambda_3$ , which was 4 and equaled to the number of O-D pairs.

Step 3: We applied Gauss-Jordan elimination, turning the augmented matrix of link/O-D pair incidence matrix  $\Lambda_3$ ,  $[\Lambda_2 \ 0 | \mathbf{I}]$ , into  $[\mathbf{H} | \mathbf{\Lambda}^c]$  and obtained the Hermite form **H** and conditional inverse matrix  $\Lambda_3^c$  of  $\Lambda_3$ .

	1	0	0	0		1.12	-0.28	0	0.18	-0.12	0	0	0	0	0	0	0	0	0]
н_	0	1	0	0	▲ C	-0.62	1.4	0	-0.91	0.62	0	0	0	0	0	0	0	0	0
- 11	0	0	1	0	$\Lambda_3 =$	-1.78	0.45	0	-1	1.78	0	0	0	0	0	0	0	0	0
	0	0	0	1		1.78	-0.45	1	2	-1.78	0	0	0	0	0	0	0	0	0

Table 8O-D pair/path incidence matrix  $\Lambda_1$  after column exchange.

O-D pair	No.	r No. Path No.																
		1	2	5	6	7	8	9	10	11	12	13	14	15	16	3	4	
1-3	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
1-4	2	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	
2-3	3	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	
2-4	4	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	

Table 9Link/path incidence matrix  $\Lambda_2$  after column exchange.

Link	No	_							Ра	th No.							
LIIIK	10.	1	2	5	6	7	8	9	10	11	12	13	14	15	16	3	4
1-5	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1
1-7	2	0	1	1	0	1	0	0	0	0	0	0	0	0	0	1	0
2-6	3	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0
2-7	4	0	0	0	0	0	0	1	0	1	0	0	0	1	1	0	0
5-3	5	1	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0
5-8	6	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1
6-4	7	0	0	0	1	1	0	0	0	0	0	1	0	0	1	0	0
6-8	8	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0

T int.	Na	Path No.															
LINK	INO.	1	2	5	6	7	8	9	10	11	12	13	14	15	16	3	4
7-8	9	0	1	1	0	1	0	1	0	1	0	0	0	1	1	1	0
8-5	10	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0
8-6	11	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0
8-9	12	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1
9-3	13	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	1
9-4	14	0	0	1	0	0	1	0	0	0	0	0	1	1	0	0	0

(Table 9 continued)

Table 10O-D demand estimates from column exchangemethod.

O-D pair No.	Preset demand	Demand estimated by column exchange method	Error
1	200	200	0.00%
2	150	150	0.00%
3	140	140	0.00%
4	185	185	0.00%

Step 4: We multiplied  $\Lambda_3^C$  by link traffic flow x to derive O-D demand q, as shown in Table 13.

The results in Table 13 show that because the rank of the link/O-D pair proportion incidence matrix  $\Lambda_3$ , which was derived by multiplying  $\Lambda_2$  by the path proportion matrix **P**, equals the number of O-D pairs and because the Hermite form **H** of  $\Lambda_3$  is an identity matrix, the estimated O-D trip demand is a unique solution which is identical to the preset values. **Table 11 Path proportion matrix P**.

# 4.3 Method Comparison

With preliminary testing, we established that the proposed methods can successfully and accurately estimate transportation demand from link traffic flow in different networks. To further understand the characteristics of these solution methods and differences among them, we tested and compared the three proposed methods which are conditional inverse matrix method, column exchange method and path flow proportion method on different networks and parameter conditions.

Three road networks were used in this section: Test Network 1, used by Yang [6] (Fig. 1), Test Network 2, which is the Nguyen-Dupuis network adopted by Castillo et al. [14] (Fig. 2), and Test Network 3, which

								0	-D pai	r No.								_
Path No.			1				2					3				4		_
	1		2	3 4	1		2	3	4	1		2	3 4	1	1	2	3 4	ŀ
1	1.00	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	
2	0.00	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	
3	0.00	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	
4	0.00	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	
5	0	0	0	0	0.67	0	0	0	0		0	0	0	0	0	0	0	
6	0	0	0	0	0.20	0	0	0	0		0	0	0	0	0	0	0	
7	0	0	0	0	0.13	0	0	0	0		0	0	0	0	0	0	0	
8	0	0	0	0	0.00	0	0	0	0		0	0	0	0	0	0	0	
9	0	0	0	0	0	0	0	0	0	.62	0	0	0	0	0	0	0	
10	0	0	0	0	0	0	0	0	0	.29	0	0	0	0	0	0	0	
11	0	0	0	0	0	0	0	0	0	.00	0	0	0	0	0	0	0	
12	0	0	0	0	0	0	0	0	0.	.09	0	0	0	0	0	0	0	
13	0	0	0	0	0	0	0	0	0		0	0	0	0.68	0	0	0	
14	0	0	0	0	0	0	0	0	0		0	0	0	0.32	0	0	0	
15	0	0	0	0	0	0	0	0	0		0	0	0	0.00	0	0	0	
16	0	0	0	0	0	0	0	0	0		0	0	0	0.00	0	0	0	

Lint	No		O-I	) pair	
LIIK	10.	1	2	3	4
1-5	1	1	0.2	0	0
1-7	2	0	0.8	0	0
2-6	3	0	0	0.38	1
2-7	4	0	0	0.62	0
5-3	5	1	0	0.91	0
5-8	6	0	0.2	0	0
6-4	7	0	0.33	0	0.68
6-8	8	0	0	0.38	0.32
7-8	9	0	0.8	0.62	0
8-5	10	0	0	0.91	0
8-6	11	0	0.33	0	0
8-9	12	0	0.67	0.09	0.32
9-3	13	0	0	0.09	0
9-4	14	0	0.67	0	0.32

Table 12Link/path proportion incidence matrix  $\Lambda_3$ .

Table 13O-D trip demand estimated by path flowproportion method.

O-D pair No.	Preset demand	Estimated demand	Error
1	200	200	0.00%
2	150	150	0.00%
3	140	140	0.00%
4	185	185	0.00%

was proposed by Yang and Zhou [17] (Fig. 3). Test Network 2 comprises of 13 nodes and 38 links, whereas Test Network 3 has 24 nodes and 76 links. As in the previous studies parameter settings differed in O-D pair numbers, path variables, numbers of paths with flow, and link/path incidence matrices, we re-applied the three algorithms to the three test networks and compared the results, which are displayed in Table 14. In this table, the estimated path solutions either are feasible solutions, which may or may not be equal to the preset values, or involve negative path flow values and therefore are infeasible.

The results indicate the following:

(1) If the conditional inverse matrix method is used directly, accurate O-D transportation demands can be derived when the variable equals the rank. However, in the event that the variable is larger than the rank, it is not guaranteed that the conditional inverse matrix method can yield an accurate solution;

(2) During the testing process for the column exchange method, discrepancies appeared between the estimated O-D demand and the preset value when the number of paths with flows was larger than the rank of the link/path incidence matrix  $\Lambda_2$ . Nevertheless, the estimated path flows were feasible solutions with positive flows. We speculate that this is because only the flows of basic variable paths are reflected when using the approach combining conditional inverse matrices and column exchange method, whereas the remaining non-basic variable flows are 0. As a result, when the number of paths actually used by users is greater than the ranks of link, it is not guaranteed that an accurate combination of path flows will be estimated, and therefore, the O-D demand derived from said path combinations may also differ from the actual trip demand;

(3) In the path flow proportion method, regardless of whether or not the number of paths with flow was greater than the rank of the link/path incidence matrix  $\Lambda_3$ , the known path proportions enabled us to convert the equation in which the variable was



Fig. 2 Test Network 2 [6].



Fig. 3 Test Network 3 [17].

 Table 14
 Comparison of results for three test networks.

Test network	No. of O-D pairs	Path variable <i>n</i>	No. of paths with flow <i>N</i>	Rank of $\Lambda_2$ , $r$	Variable and rank relationship	Algorithm	Estimated path solution	comparison between estimated value and preset value
		9	9	9	N = N = r	Conditional inverse matrix	Preset solution	No error present
1	4	15	0	9	n > N - n	Conditional inverse matrix	Infeasible solution	Error present
			9		n > n - r	Column exchange Path flow proportion	Preset solution	No error present
		26	26	26	n = N = r	Conditional inverse matrix	Preset solution	No error present
		10	26	26	n > N - n	Conditional inverse matrix	Infeasible solution	Error present
	12	48	26	26	n > N = r	Column exchange Path flow proportion	Preset solution Preset solution	No error present No error present
2		40	40		N7.	Conditional inverse matrix	Infeasible solution	Error present
		48	48	26	n = N > r	Column exchange Path flow proportion	Feasible solution Preset solution	Error present No error present
	10	70		28	375	Conditional inverse matrix	Infeasible solution	Error present
	18	79 79	/9		n = N > r	Column exchange Path flow proportion	Feasible solution Preset solution	Error present No error present

(Table 14 C	continuea)							
Test network	No. of O-D pairs	Path variable <i>n</i>	No. of paths with flow <i>N</i>	Rank of $\Lambda_2$ , $r$	Variable and rank relationship	Algorithm	Estimated path solution	Comparison between estimated value and preset value
		48	48	48	n = N = r	Conditional inverse matrix	Preset solution	No error present
		82	40	50		Conditional inverse matrix	Infeasible solution	Error present
	16		48	50	n > r > N	Column exchange	Preset solution	No error present
						Path flow proportion	Preset solution	No error present
3				-	37.	Conditional inverse matrix	Infeasible solution	Error present
		82	82	50	n = N > r	Column exchange	Feasible solution	Error present
						Path flow proportion	Preset solution	No error present
	20	154	154	58	37	Conditional inverse matrix	Infeasible solution	Error present
	30				n = N > r	Column exchange	Feasible solution	Error present
						Path flow proportion	Preset solution	No error present

Application of the Conditional Inverse Matrix Theory for Estimation of Origin-Destination Trip Matrices 33 from Link Traffic Flows

originally greater than the rank,  $\Lambda_2 h = x$ , into an equation in which the variable equals the rank,  $\Lambda_3 \mathbf{q} = \mathbf{x}$ . For this reason, the estimated O-D demand was the only solution and equaled the preset value. It is evident that when the path proportion data of a road network can be obtained, O-D trip demand can be estimated more accurately and closer to actual values.

### 5. Discussion and Conclusions

(T 11 14

..

In this study, we applied conditional inverse matrices approach to solve the linear equations formed by path flow and link traffic flow. Using a column exchange method, we obtained non-negative and thus feasible path flow solutions to estimate corresponding O-D trip demand. We also confirmed that the addition of path proportion data effectively narrows the range of feasible solutions when using conditional inverse matrices. With three test networks, we verified the applicability of the proposed methods combining conditional inverse matrices with column exchange method and path flow proportion method. In particular, we formulate the following conclusions.

(1) Under circumstances where there are no errors in the observed link traffic flows, a linear equation relationship exists among the O-D demand, path flow, and link traffic flow in the network. Thus, when estimating O-D demand from link traffic flow, conditional inverse matrices can be used in the solution process;

(2) The results of this study indicate that when the rank of the link/path incidence matrix equals the path variable, only a unique conditional inverse matrix exists. In contrast, when the rank is less than the path variable, multiple solutions will occur;

(3) Even if multiple solutions are generated, using the column exchange method will guarantee feasible solutions. While the O-D trip matrix derived may not be accurate, feasible solutions for the matrix can be estimated from link traffic flow without inputting additional information;

(4) During the testing process, we saw that in the application of the column exchange method, the estimated O-D demand will equal the preset value as long as the rank of the link/path incidence matrix  $\Lambda_2$  equals the path number. When the path number is greater than the rank, errors will occur in the estimated O-D trip demand. Nevertheless, the path flow values derived are consistently positive; therefore, the O-D demand obtained will still be a feasible solution suitable for practical application;

(5) When there are multiple solutions, additional path proportion data will enable the path flow proportion method to obtain correct O-D trip demand in the event that the number of O-D pairs is less than the number of links, which is generally the case in

road networks;

(6) Despite the fact that multiple solutions exist when the path number is larger than the rank, known path proportions can convert the equation  $\Lambda_2 h = x$  in which the variable was originally greater than the rank into the equation  $\Lambda_3 \mathbf{q} = \mathbf{x}$  in which the variable equals the rank using the link/O-D pair proportion incidence matrix  $\Lambda_3$ , which was derived by multiplying the link/path incidence matrix  $\Lambda_2$  by the path proportion matrix. Solving the equation  $\Lambda_3 \mathbf{q} = \mathbf{x}$  will then yield the correct O-D trip demand solution.

We have shown that the path flow proportion method is an effective approach for solving networks in which multiple solutions exist. Furthermore, our results demonstrate that it is extremely accurate with the requisite data, i.e., path flow proportion between each O-D pair. The aim of this study was to illustrate that a unique solution for the O-D estimation problem can be theoretically derived, and desirable network O-D trip demand estimates obtained without any further assumptions. In our method, path variables can be replaced by O-D pair variables, thereby significantly reducing the number of columns in the link proportion/O-D incidence matrix  $\Lambda_3$ . If the number of links is larger than the number of O-D pairs in a given network, then rank  $(\Lambda_3) = r =$ column *n*. For the linear system  $\Lambda_3 \mathbf{q} = \mathbf{x}$ , a unique solution can be obtained by the conditional inverse  $\mathbf{q} = \mathbf{\Lambda}_3^C \mathbf{x}$ . To obtain the requisite data in the case of multiple solutions, there are two approaches which could be used. In an urban network where most road users are frequent network users, the path flow proportions between each O-D pair could be estimated by a sampling survey. Users' route choice probabilities can also be obtained using advanced sensor technologies (e.g., AVI (automatic vehicle identification) or license plate recognition technologies). Since path flow proportions are the most important element in the estimation of O-D matrices from link traffic flows, the effect of sampling error rates could be significant. Future studies may consider this as well as the construction of a sampling survey mechanism.

# Acknowledgments

The authors sincerely thank the National Science Council (Project No: NSC 101-2410-H-606-010) for funding this study.

#### References

- Robillard, P. 1973. "Estimating the O-D Matrix and Network Characteristic from Observed Link Volumn." In Proceedings of the International Conference on Transportation Research, First Conference, 736-40.
- [2] Cascetta, E. 1984. "Estimation of Trip Matrices from Traffic Count and Survey Data: A Generalized Least Squares Estimation." *Transportation Research Part B: Methodological* 18 (4-5): 289-99.
- [3] Cascetta, E., and Nguyen, S. 1988. "A Unified Framework for Estimating or Updating Origin/Destination Matrices from Traffic Counts." *Transportation Research Part B: Methodological* 22 (6): 437-55.
- [4] Bell, M. G. H. 1991. "The Estimation of Origin-Destination Matrix by Constrained Generalized Least Square." *Transportation Research Part B: Methodological* 25 (1): 13-22.
- [5] Doblas, J., and Benitez, F. G. 2005. "An Approach to Estimating and Updating Origin-Destination Matrices Based upon Traffic Counts Preserving the Prior Structure of a Survey Matrix." *Transportation Research Part B: Methodological* 39 (7): 565-91.
- [6] Yang, H. 1995. "Heuristic Algorithms for the Bilevel Origin-Destination Matrix Estimation Problem." *Transportation Research Part B: Methodological* 29 (4): 231-42.
- [7] Maher, M. J., Zhang, X., and van Vliet, D. 2001. "A Bi-level Programming Approach for Trip Matrix Estimation and Traffic Control Problems with Stochastic User Equilibrium Link Flows," *Transportation Research Part B: Methodological* 35 (1): 23-40.
- [8] Lundgren, J. T., and Peterson, A. 2008. "A Heuristic for the Bilevel Origin-Destination-Matrix Estimation Problem." *Transportation Research Part B: Methodological* 42 (4): 339-54.
- [9] Sherali, H. D., Sivanandan, R., and Hobeika, A. G. 1994.
   "A Linear Programming Approach for Synthesizing Origin-Destination Trip Tables from Link Traffic Volumes." *Transportation Research Part B:*

Methodological 28 (3): 213-33.

- [10] Bell, M. G. H., and Shield, C. M. 1995. "A Log-linear Model for Path Flow Estimator." In Proceedings of 4th International Conference on the Applications of Advanced Technologies in Transportation Engineering, 695-99.
- [11] Bell, M. G. H., Shield, C. M., Busch, F., and Kruse, G. 1997. "A Stochastic User Equilibrium Path Flow Estimator." Transportation Research Part C: Emerging Technologies 5 (3/4): 197-210.
- [12] Chen, A., Chootinan, P., and Recker, W. 2009. "Norm Approximation Method for Handling Traffic Count Inconsistencies in Path Flow Estimator." *Transportation Research Part B: Methodological* 43 (8-9): 852-72.
- [13] Gentili, M., and Mirchandani, P. B. 2005. "Locating

Active Sensors on Traffic Networks." *Annals of Operations Research* 136: 229-57.

- [14] Castillo, E., Menéndez, J. M., and Jiménez, P. 2008.
   "Trip Matrix and Path Flow Reconstruction and Estimation Based on Plate Scanning and Link Observations." *Transportation Research Part B: Methodological* 42 (5): 289-99.
- [15] Graybill, F. A. 1983. Matrices with Applications in Statistics, an Introduction to Linear Statistical Models. Belmont, California: Wadsworth, Inc.
- [16] Strang, G. 2003. *Introduction to Linear Algebra*. USA: Wellesley-Cambridge Press.
- [17] Yang, H., and Zhou, J. 1998. "Optimal Traffic Counting Locations for Origin-Destination Matrix Estimation." *Transportation Research Part B: Methodological* 32 (2): 109-26.