Katholieke Universiteit Leuven

Departement Elektrotechniek

ESAT-SISTA/TR 98-82

DYNAMIC TOTAL LEAST SQUARES ESTIMATION OF INTERSECTION ORIGIN-DESTINATION MATRICES

Baibing Li and Bart De Moor[2]

August 1998

[1]This paper is also available by anonymous ftp from ftp.esat.kuleuven.ac.be as file rep9882.ps.Z in the directory pub/SISTA/baibing/reports.

[2]ESAT/SISTA, K.U.Leuven, Kardinaal Mercierlaan 94, B-3001 Heverlee (Leuven), Belgium, tel.+32-16-32.18.55, fax +32-16-32.19.70, email: baibing.li@esat.kuleuven.ac.be.

DYNAMIC TOTAL LEAST SQUARES ESTIMATION OF INTERSECTION ORIGIN-DESTINATION MATRICES

Baibing Li and Bart De Moor

ESAT-SISTA, Dept. of Electrical Engineering, Katholieke Universiteit Leuven
Kardinaal Mercierlaan 94, 3001 Leuven, Belgium

Abstract—The total least squares techniques are introduced to estimate intersection origin-destination matrices when both entering and exiting counts have random measurement errors. Compared with the recursive estimation methods proposed by Creamer & Keller (1987) and Nihan & Davis (1987), its accuracy can be improved substantially in numerical results, especially when the measurement errors are relatively large. Hence, the total least squares based estimation provides a competitive alternatives to identifications of intersection origin-destination matrices with measurement errors.

1. INTRODUCTION

In recent two decades, total least squares (TLS) techniques have been paid more and more attention in theory and methodology. Meanwhile, they have been applied with success to some practical fields. It is shown that in typical situations, for instance, when both dependent and independent variables have measurement errors, substantial gains in accuracy can be obtained using TLS with respect to the ordinary least squares (OLS) methods (Van Huffel & Vandewalle, 1991). For recent development, see, Van Huffel (1997). In this paper, we will consider their application in estimation of intersection origin-destination matrices.

For management and control for intersections, it is a fundamental requirement knowing the information on turning movements and through volumes, or called intersection origin-destination (O-D) patterns. Generally speaking, based on the entering and exiting automatic counts in each legs of an intersection, the O-D pattern can be identified by using appropriate estimation methods. Among all of the approaches of estimating the O-D matrices for intersections, it is most noticeable that Creamer & Keller (1987) and Nihan & Davis (1987) proposed recursive (OLS) estimations. It is especially appropriate for real time identification and can be used to embedded into an adaptive controller for signals timing in intersections.

However, according to the TLS theory and application cases, when both entering and exiting counts have measurement errors, an OLS based estimator does not have satisfactory behavior. Hence, it is natural to use the TLS, instead of the OLS, to increase accuracy of estimation of intersection O-D matrices in such cases.

In this paper, we investigate the dynamic implementation of TLS estimation of intersection O-D matrices and its performance. We show by simulation experiments that, under some conditions, its performance is substantially better than the one of recursive OLS estimation proposed by Cremer & Keller (1987) and Nihan & Davis (1987). Therefore, it provides a competitive alternative for estimation of intersection O-D matrices.

2. PROBLEM FORMULATION

We consider an intersection with m entering legs and n exiting legs. It is assumed that traffic counts taken at N constant intervals of time are available for each entering and exiting legs. Let

$y_j(t)$ (j=1,...,n) denote the exiting count at leg j during time interval t;

$q_i(t)$ (i=1,...,m) denote the entering count at leg i during time interval t;

b_{ij} (i=1,...,m; j=1,...,n) denote the probability a vehicle entering at i exiting at j.

$y(t)=[y_1(t),......,y_n(t)]^T$; $q(t)=[q_1(t),......,q_m(t)]^T$

$B=[b_{ij}]_{m\times n}$ $b^{(i)}=[b_{i1},......,b_{in}]^T$ $b_j=[b_{1j},......,b_{mj}]^T$, $b=[b_{1,}^T......,b_n^T]^T$.

where z^T denotes the transpose of the vector z.

The elements of B satisfy

$$\sum_{j=1}^{n} b_{ij} = 1, \qquad i = 1, \cdots, m \,. \tag{1a}$$

$$b_{ij} \geq 0, \qquad i=1,\ldots,m, \quad j=1,\ldots,n. \tag{1b}$$

$$b_{ij}=0, \qquad (i,j) \in Z. \tag{1c}$$

where the set Z will differ from problem to problem but contain those O-D pairs which are known a priori to have zero flow. Usually, the constraints (1c) can be satisfied by eliminating the elements whose indexes belong to the set Z. Particularly, for a typical four-way intersection prohibiting U-turns, $Z=\{(1,1),(2,2),(3,3),(4,4)\}$.

Following Creamer & Keller (1987) and Nihan & Davis (1987), we consider linear model

$$y_j(t) = q^T(t) b_j + e_j(t), \qquad j = 1, \cdots, n. \tag{2}$$

where $e_j(t)$ is a random variable with zero mean.

For the observation $y(t)$ and $q(t)$ available at each time interval t in practice, the problem of dynamic estimation of $B=[b_{ij}]$ is to find $\hat{B}(t) = [\hat{b}_{ij}(t)]$ at each time interval t such that it can (approximately) solve following optimization problem

$$J = \min_{B} \sum_{s=1}^{t} \sum_{j=1}^{n} \{y_j(s) - q^T(s) b_j\}^2 \qquad t=1,......,N \tag{3}$$

subject to (1).

3. THE TLS BASED ESTIMATION

In this section, we consider the dynamic TLS based estimation of intersection O-D matrices.

The recursive OLS estimation, considered by Creamer & Keller (1987) and Nihan & Davis (1987), has a special algorithm such that the demands on computation and memory not increase as time when the observances are obtained sequentially.

Similarly, appropriate algorithms of recursive TLS estimation are needed for intersection O-D matrices. As we know, however, most of dynamic TLS algorithms require $O(n^3)$ flops. Procedures with $O(n^2)$ flops, for instance, by Gu & Eisenstat (1995), are not practical at present, or by Barlow and Yoon (1997), are approximate approaches. Moreover, if the equality constraints (1a) are considered to obtain a simultaneous estimation, more complicated situations occur due to the structured zeros. Several algorithms termed structured TLS algorithms, only for non-recursive one, were investigated by Lemmerling, De Moor and Van Huffel (1996), Rosen, Park, and Glick (1996) (see, also, Van Huffel, 1997). No recursive structured TLS algorithm has been proposed. Finally, the estimation problem based on TLS will be much more difficult if the inequality constraints (1b) are also considered. De Moor (1990) investigated the TLS problems with both linear equality and inequality constraints based on Karush-Kuhn-Tucher conditions. However, similar to the OLS, this approach is hard to be introduced into recursive version.

In this paper, we consider a practical compromise approach to deal with this problem. Firstly, we adopt a two-step estimation proposed by Creamer & Keller (1987) and Nihan & Davis (1987), that is, in first step, obtaining an unconstrained estimation, and then enforcing the constraints (1) in second step. Moreover, since the estimation problem in first step is decomposed into several independent ones, each with small dimensions, the TLS estimations are obtained by direct computation rather than use (approximate) recursive version. Finally, for computation of TLS estimation in each time interval, we use eigenvalue decomposition rather than singular decomposition due to the former has recursive property for memory.

More specifically, let $h_j(t)$ and $\hat{b}_j(t)$ (j=1,...,n) be the vectors of $q(t)$ in (2) and estimator of b_j, crossing out those elements whose indexes belong to the set Z at time interval t. Then this algorithm is summarized as follows:

(a)For the measurement information $g_j(t)=[h_j^T(t), y_j(t)]$ at each time interval t, let

$$C_j(t)=\begin{bmatrix} g_j(1) \\ \cdots \\ g_j(t) \end{bmatrix} \quad \text{and} \quad G_j(t)=C_j^T(t)C_j(t)$$

Obtaining the eigenvector $v_j(t)$ of $G_j(t)$ corresponding to the minimum eigenvalue. Let $\tilde{b}_j(t)$ be the scaled $v_j(t)$, *i.e.*, its last component is -1.

(b)Using the projection proposed by Nihan & Davis (1987) to enforce the constraints (1) satisfied, that is,

Truncation for the constraints (1b)

$$T(\tilde{b}_{ij}(t))=\begin{cases} \tilde{b}_{ij}(t) & \text{if } 0 \le \tilde{b}_{ij}(t) \le 1 \\ 0 & \text{if } \tilde{b}_{ij}(t) < 0 \\ 1 & \text{if } \tilde{b}_{ij}(t) > 1 \end{cases}$$

Normalization for the constraints (1a)

$$\hat{b}_j(t)=N(\tilde{b}^{(i)}(t))=\tilde{b}^{(i)}(t)/[(\tilde{b}^{(i)}(t))^T \boldsymbol{1}]$$

where $\boldsymbol{1}$ denotes a vector of ones of appropriate dimension.

(c)If $t<N$, let $G_j(t+1)=C_j^T(t+1)C_j(t+1)=G_j(t)+g_j^T(t+1)g_j(t+1)$. Go to (i).

Remarks: (i)Noting that the dimension of $G_j(t+1)$ is less than $n\times n$ (for a typical four-way intersection prohibiting U-turn, its dimension is 3×3), using non-recursive approach rather than recursive one to solve eigenvalue problem dose not increase too much computational amount.

(ii)From (c), $G_j(t+1)$ can be updated by $g_j(t+1)$ in each time step t. Hence, storage for $G_j(t)$ rather than $C_j(t)$ means that the demand for memory do not increase as time. That is the reason we use eigenvalue decomposition rather than use more usual singular value decomposition in the above algorithm.

(iii)Generally, using recursive estimation has two main advantages in algorithm: first, decreasing computational amount; second, memory not increasing as time. From this point of view, the above algorithm is feasible in practice.

4.NUMERICAL COMPARISON

In this section we investigate whether the TLS based estimation has significant improvement in accuracy compared with the OLS based estimation proposed by Creamer & Keller (1987) and Nihan & Davis (1987).

4.1. Simulation Setting

(i) Parameters and criteria for comparison

Consider a typical four-way intersection prohibiting U-turn, n=m=4. The “true” parameter matrix B is given by (see Nihan & Davis 1987):

$$B = \begin{bmatrix} 0 & 0.2 & 0.7 & 0.1 \\ 0.05 & 0 & 0.8 & 0.15 \\ 0.3 & 0.2 & 0 & 0.5 \\ 0.1 & 0.8 & 0.1 & 0 \end{bmatrix}$$

The length of simulated period of time is set as N=100. To decrease the random effect for results, S=100 separate simulation experiments are conducted. Moreover, to eliminate the results’ dependence on initial value at t=0 in the recursive algorithm, we start each simulation at time t_0=13 and use the values at t=12 as the initial values.

For each simulation experiment, define the distance $R(t) = \left\| \hat{B}(t) - B \right\|_2$ at each time interval t. Consider the total time average (TTA) of $R(t)$ and the average over the last ten time intervals (LTA) of $R(t)$ to measure the average and final performances of each estimation method in each simulation experiment as follows*:*

$$TTA(N) = \sum_{t=13}^{N} R(t) / (N-12) \qquad \text{and} \qquad LTA(N) = \sum_{t=N-9}^{N} R(t) / 10$$

Finally, denote the means of TTA and LTA over 100 simulation experiments as $MTTA$ and $MLTA$ respectively

$$MLTA = (1/100)\sum_{i=1}^{100} [LTA \text{ of the } i\text{th simulation experiment}]$$

$$MTTA = (1/100)\sum_{i=1}^{100}[TTA \text{ of the } i\text{th simulation experiment}]$$

(ii)Generation of random variables

Similar to Nihan & Davis (1987), the "true" entering counts $q_i(t)$ are generated as the outcome of a Poisson random variable with mean 100. The observed entering counts, which are the entering counts with measurement errors, are denoted as $p_i(t)$ and generated as

$$p_i(t)= q_i(t)+\mu\, q_i(t)\varepsilon_i(t)$$

where $\varepsilon_i(t)$ is standard normal random variable. μ is a proportional constant and taken from 0.01 to 0.05 in simulation.

The exiting counts $y_j(t)$ are generated as:

$$y_j(t) = q^T(t)b_j + \nu[q^T(t)b_j]\zeta_j(t)\,, \qquad t=1,...N.$$

where $\zeta_j(t)$ is the standard normal random variable. ν is a proportional constant and will be taken as 0.04 and 0.05 in simulation. $\varepsilon_i(t)$ and $\zeta_j(t)$ are mutually independent ($i=1,...m;$ $j=1,...n$).

4.2. Simulation Experiments

The summary results of the 100 simulation experiments are listed in Table 1.

Table 1. The average performances over 100 simulation experiments with time period length $N=100$ for two estimation methods

	TLS based	estimation	OLS based	estimation
	MLTA	MTTA	MLTA	MTTA
μ=0.05 ν=0.05	0.1685	0.2254	0.2557	0.3231
μ=0.04 ν=0.05	0.1621	0.2093	0.2237	0.2888
μ=0.03 ν=0.05	0.1763	0.2146	0.1989	0.2686
μ=0.02 ν=0.05	0.1829	0.2177	0.1944	0.2616
μ=0.01 ν=0.05	0.1923	0.2209	0.1728	0.2443
μ=0.00 ν=0.05	0.1950	0.2235	0.1744	0.2461
μ=0.04 ν=0.04	0.1232	0.1685	0.1852	0.2512
μ=0.03 ν=0.04	0.1350	0.1730	0.1664	0.2339
μ=0.02 ν=0.04	0.1386	0.1712	0.1439	0.2037
μ=0.01 ν=0.04	0.1466	0.1738	0.1556	0.2125
μ=0.00 ν=0.04	0.1517	0.1771	0.1443	0.2089

It can be seen from Table 1 that

(i) generally speaking, the results of TLS based estimation are substantially better than the ones of the OLS based estimation.

(ii) when the measurement errors in entering counts are relatively large (μ=0.04 and 0.05), the superiority of the TLS based estimation over the OLS based estimation is more significant.

However, it should be noted that the simulation experiments show that the TLS based estimations are too sensitive to the relatively large variance of exiting counts. This is partly because that the TLS gives magnifying, rather than shrinking, estimators. Moreover, it also may be due to the fact that the generated data for simulation violate the assumptions of the equal variance in both dependent and independent variables. Therefore, a careful investigation is needed in practice to choose an appropriate estimation approach.

5.CONCLUSIONS

In this paper, we introduce TLS based estimation to identify intersection O-D matrices. It is shown that although it is not a recursive one, the demand for memory will not increase as time. Moreover, since the dimensions are small for decomposed intersection O-D matrices, the increase in computation is quite limited. Simulation experiments show that when both entering and exiting counts have measurement errors, the TLS based estimation can improve the accuracy substantially compared with the OLS based estimation. Therefore, it provides a good alternative for estimating O-D matrices for different situations.

Acknowledgement—The second author of this paper is a senior Research Associate with the FWO-Flanders (Fund for Scientific Research-Flanders). This work was supported by Concerted Research Action GOA-MIPS, the FWO project G.0256.97: Numerical Algorithms for Subspace System

Identification, the FWO Research Communities: ICCOS and Advanced Numerical Methods for Mathematical Modelling, and the Belgian Government: Interuniversity Attraction Poles (IUAP P4-02 and P4-24).

REFERENCES

Barlow, J.L. and Yoon, P. A. (1997) Solving resursive TLS problems using the rank-revealing ULV decomposition, In: Recent Advances in Total Least Squares Techniques and Errors-in-Variables Modeling, Van Huffel, S. ed., SIAM, Philadephia, PA.

Cremer, M. and Keller, H. (1987) A new class of dynamic methods for the identification of origin-destination flows, *Transpn Res.*, Series B21, 117-132.

De Moor, B. (1990) Total linear least squares with inequality constraints, ESAT/SISTA report TR 90-02, Dept. Of Electr. ENG., Katholieke Universiteit Leuven, Belgium.

Gu, M. And Eisenstat, S. (1995) Downdating the singular value decomposition, SIAM, J. Matrix Anal. Appl., 16, 793-810.

Lemmerling, P. De Moor, B. and Van Huffel, S. (1996) On the equivalence of constrained total least squares and structured total least squares, IEEE Trans. On Signal Processing, SP-44,

Nihan, N. and Davis, G. (1987) Recursive estimation of origin-destination matrices from input/output counts, *Transpn Res.*, Series B21, 149-163.

Rosen, J.B., Park, H. and Glick, (1996) Total least norm formulation and solution for structured problems, SIAM J. Matrix Anal. Appl. 17, 110-126.

Van Huffel, S. (1997) Recent Advances in Total Least Squares Techniques and Errors-in-Variables Modeling, SIAM, Philadephia, PA.

Van Huffel, S. and Vandewalle, J. (1991) The Total Least Squares Problem: Computational Aspects and Analysis, SIAM, Philadephia, PA.