IMPROVING THE RELIABILITY OF A TWO-STEP DYNAMIC DEMAND ESTIMATION APPROACH BY SEQUENTIALLY ADJUSTING GENERATIONS AND DISTRIBUTIONS

Guido Cantelmo
Faculty of Science, Technology and Communication
Research Unit in Engineering Science
University of Luxembourg
Rue Coudenhove-Kalergi 6, L1359 Luxembourg city, Luxembourg
guido.cantelmo@uni.lu

Francesco Viti
Faculty of Science, Technology and Communication
Research Unit in Engineering Science
University of Luxembourg
Rue Coudenhove-Kalergi 6, L1359 Luxembourg city, Luxembourg
francesco.viti@uni.lu

Ernesto Cipriani
Department of Engineering
Roma Tre University
Via Vito Volterra 62, 00146, Rome, Italy
ernesto.cipriani@uniroma3.it

Marialisa Nigro
Department of Engineering
Roma Tre University
Via Vito Volterra 62, 00146, Rome, Italy
marialisa.nigro@uniroma3.it

Word Count

No. of words: 5494

No. of figures: 6*250=1500 Figures4, tables2

Total: 6994 + 32 references

ABSTRACT

This work proposes a procedure to simplify the demand estimation problem in the dynamic case while guaranteeing reliable solutions and without increasing problem complexity. The procedure does not claim to be an alternative to existing theoretical estimation approaches, but focuses on extending and testing practical solution algorithms based on previous models developed by the authors, in order to improve their applicability in terms of networks sizes and in cases where a reliable a priori estimate of the demand is not available. In fact, the assumption often made by researchers about its availability, or its reliability is not always true. Thus, for dealing with this occurrence a two-step approach is proposed in this paper: it aims at estimating the proper level of demand generated by any traffic zone in the first step, and accurate demand distributions between the different OD in the second one, while preserving the correct traffic regime. Tests carried out show that a more reliable estimation of the demand over time and space is achieved.

Main contributions of the new approach are a) a considerable reduction of the variance in calibration parameters, implying that robust and accurate solutions have been obtained; b) its suitability to large-scale networks. The latter point derives from a new variant of the widely adopted Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm proposed in the second step, where an opportune subset of the OD flows is perturbed at once. Performed experiments show that the proposed procedure is able to obtain results as accurate as those of the conventional SPSA reducing the number of variables to be used in each iteration up to 50%.

INTRODUCTION

Simulation of traffic conditions requires as main input the knowledge of travel demand. When dealing with transportation networks, traffic conditions are usually not stationary, hence it is recommended to adopt time-dependent profiles of the travel demand to best represent congestion and its propagation. If this information is not available or incorrect, the simulation output performances are compromised.

The problem of estimating travel demand in case of non-stationary conditions is well known in literature as Dynamic Demand Estimation Problem (DDEP). DDEP searches for temporal Origin-Destination (OD) matrices able to best-fit measured data. DDEP can be applied for both within day (intra-period) and day-to-day (inter-period) dynamic frameworks, as well as for offline (medium-long term planning and design) and on-line (real-time management) contexts.

DDEP is commonly classified between sequential or simultaneous approaches [1], where usually the first is adopted for on-line applications, while the second for offline applications. Another classification can be done according to the type of observed data adopted for the estimation: usually traffic counts are adopted, but recently also other measures such as speeds and occupancies are introduced to take into account the congestion state of the network [2]. These data, as the traffic counts, are commonly link-based, while also other path-based data can be added as probe data from vehicle equipped by AVI tags ([3-7]). When only traffic counts are adopted for the estimation, the link between dynamic travel demand and measurements is usually captured by the assignment matrices (explicitly, as in [8], or by a linear approximation of the assignment matrices [9-10]). While the assignment matrix fully establishes direct relationships between OD flows and link flows, the interaction between other measurements, whether link or path based, is not directly represented and a simulation approach is preferred.

Online solution algorithms, based on different state-space representations of traffic flow propagation, and tuned with advanced regression methods such as Kalman filtering [11], are very popular for capturing within-day dynamics and calibrating traffic models [12] using real-time data [13]; however, studies on Kalman filtering are also proposed for the offline context [11-14]. In offline applications DDEP is generally formulated as a bi-level optimization problem, where in the upper level demand matrices are corrected using measured data while in the lower level DTA simulation is performed to obtain the synthetic data [15-16]. Generally, the upper level problem is solved using stochastic or deterministic path search approaches [17]. Recently, stochastic solution approaches were proposed along this direction, as in Antoniou et al. [18] and Cipriani et al. [19]. Other approaches work on the solution space dimension. Djukic et al. [20] applies Principle Component Analysis to study the matrices high-dimensional data structure while Flötteröd and Bierlaire [21] propose to improve DDEP using a new linearization of the network loading map in order to overcome the inadequacy of a proportional assignment in congested conditions.

To reduce the solution space and the set of possible solutions, classical methods – called “single level” in this paper - often just include information about a reference OD demand matrix (usually known as seed matrix) whose solutions have demand levels similar to the starting one. Therefore if the seed matrix is different from the real one, this localism can lead to significant errors [22]. The need for methods dealing with the correction of the seed matrix in such applications was pointed out recently by Cantelmo et al. [23], who proposed a two-step approach where the first step was focused on correcting the seed matrix by focusing on the OD flows having largest impact on the measured link counts. This two-steps procedure demonstrated its ability in correcting the starting demand value without introducing new traffic measures, apart from traffic counts, or developing new models, and effectively improved the results on congested networks by correcting the seed matrix in the first step and directing it towards the real demand values. Though effective, the developed method can hardly be applied on large-sized networks, where the number of OD pairs to be selected in the first step may become significantly large.

The contributions of this paper are twofold. In the next section, we propose an enhancement of the previously developed two-steps approach by exploiting information on aggregated demand data such as generation data by zones, which are adopted in the first step of the proposed procedure. Specifically, the first step searches for generation values that best represent the measurements (traffic counts); hence, in the first step the variables are no more the dynamic OD trips, but the total production values, thus reducing the dimension of the problem considerably. In the second step, the classical DDEP procedure is performed improving temporal and spatial matrix distributions. Breaking the problem as such, one benefits of the right demand level identified in the first phase, avoiding single-step localism problems. Since the proposed two step approach allows the reduction of the number of variables used in DDEP with respect to the one step case, it becomes less sensible to the network size. Further, in the current paper a method to reduce the number of variables is introduced in the second step, reducing the significantly the problem size towards the application to large-sized networks.

The proposed approach has been later applied on a real network case, resulting more robust in terms of goal function trends, link flows and traffic state representations. Conclusions and future research directions conclude this paper.

METHODOLOGY

The DDEP is generally solved as an optimization problem. Its formulation requires the specification of the objective function, also known as goal function, its variables, elements of the dynamic OD demand matrix to be estimated, and its constraints related to feasibility and routing conditions. Considering different types of measures and a simultaneous approach the problem can be formulated as:

$$(\boldsymbol{d}_1^*, \dots, \boldsymbol{d}_n^*) = argmin \begin{bmatrix} z_1(\boldsymbol{l}_1, \dots, \boldsymbol{l}_n, \widehat{\boldsymbol{l}_1}, \dots, \widehat{\boldsymbol{l}_n}) + \\ +z_2(\boldsymbol{n}_1, \dots, \boldsymbol{n}_n, \widehat{\boldsymbol{n}_1}, \dots, \widehat{\boldsymbol{n}_n}) + \\ +z_3(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n, \widehat{\boldsymbol{x}_1}, \dots, \widehat{\boldsymbol{x}_n}) + \\ +z_4(\boldsymbol{r}_1, \dots, \boldsymbol{r}_n, \widehat{\boldsymbol{r}_1}, \dots, \widehat{\boldsymbol{r}_n}) + \end{bmatrix} \quad (1)$$

Where

- $\mathbf{l}/\hat{\mathbf{l}}$ are respectively simulated values and measurements on the links;
- $\mathbf{n}/\hat{\mathbf{n}}$ are respectively simulated values and measurements on the nodes;
- $\mathbf{x}/\hat{\mathbf{x}}$ are respectively estimated value and previous information on dynamic demand (seed matrix);
- $\mathbf{r}/\hat{\mathbf{r}}$ are respectively simulated values and measurements on routes;
- $\boldsymbol{d}_{\boldsymbol{n}}^*$ estimated demand matrix for time interval n;
- $z{:}\{z_1, z_2, z_3, z_4\}$ is the estimator represented by the deviations between simulated/estimated and measured/a priori values.

The dependence between simulated information in (1) and the estimated demand is obtained directly by simulation performing a dynamic traffic assignment (DTA), so that:

$$\boldsymbol{l}_1, \dots, \boldsymbol{l}_n = \mathbf{F}(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$$
$$\boldsymbol{n}_1, \dots, \boldsymbol{n}_n = \mathbf{F}(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$$
$$\boldsymbol{r}_1, \dots, \boldsymbol{r}_n = \mathbf{F}(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$$

With **F** = Dynamic Traffic Assignment (DTA) function.

Generation-distribution adjustment process

In the proposed two-steps procedure, the first step aims at optimizing the generation values of each zone in each time interval, while maintaining constant the dynamic trip distributions derived by the seed matrix. The objective function in (1) can be generally rewritten for the first step as:

$$(\boldsymbol{E}_1^*, \dots, \boldsymbol{E}_n^*) = argmin \begin{bmatrix} z_1'(\boldsymbol{l}_1, \dots, \boldsymbol{l}_n, \widehat{\boldsymbol{l}_1}, \dots, \widehat{\boldsymbol{l}_n}) + \\ +z_2'(\boldsymbol{n}_1, \dots, \boldsymbol{n}_n, \widehat{\boldsymbol{n}_1}, \dots, \widehat{\boldsymbol{n}_n}) + \\ +z_3'(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n, \widehat{\boldsymbol{x}_1}, \dots, \widehat{\boldsymbol{x}_n}) + \\ +z_4'(\boldsymbol{r}_1, \dots, \boldsymbol{r}_n, \widehat{\boldsymbol{r}_1}, \dots, \widehat{\boldsymbol{r}_n}) + \end{bmatrix} \quad (2)$$

Where $x_n^{OD} = E_n^O d_{D|O}^{Seed,n} \quad \forall O, \forall D, \forall n$

Where:

$E_n{}^O$ = generation of origin zone O and time interval n;
$\boldsymbol{E}_n^*$ = generation vector containing generation from all origins in time interval n.
x_n^{OD} = trips flow from origin zone O to destination zone D in time interval n.
$d_{D|O}^{Seed,n}$ = seed matrix probability distribution to move in traffic zone D from traffic zone O in time interval n.

The idea of working on production values in the first step, rather than on dynamic OD trips, derives by the increasing attention received by this type of aggregated information in the literature. Already Iannò and Postorino [24] proposed a generation-constrained approach for the static demand estimation problem where the objective function contains a specific term in order to prevent the emission from each origin zone to be greater than the actual one. Then, also Cipriani et al. [25] and Cipriani et al. [19] proposed to introduce a generation constraint in the dynamic demand estimation. Cascetta et.al [8] proposed a quasi-dynamic approach where the main assumption is that the demand generation changes much faster than the distributions. The OD shares are then considered constant across reference period, while total flows leaving each origin varies for each sub-period within the reference period. Finally, in Cantelmo et al. [26] some remarks are reported about the possible adoption of the generation values as a constraint in the DDEP.

The high significance given in literature to this aggregated information derives mainly by the following considerations:

- Total generated trips can act by limiting a demand overestimation during the DDEP; the overestimation can usually occur when dealing with traffic measurements collected on congested networks;
- Total generated trips are more easily available than OD trips, and generation models, from which these data are obtained, are considered the most reliable models in transport engineering applications;
- Adopting the generation values inside the DDEP, as in (2), reduces the number of variables (from O×D×n to O×n): The expected result of this phase is the correct level of generated demand for each time interval.

Accordingly, the goal of the first step is to act on the seed matrix in order to obtain a “right level of demand”, then moving to the second step in order to optimize the dynamic distributions OD trips as in (1).

The present approach has analogies with the quasi-dynamic approach reported in [8]. In the latter, distributions are explicitly considered in terms of probabilities and approximated as an average over a time period greater of the time slice itself; in this approach we assume them constant and equal to the one of the seed matrix, in the first step, while they are considered as unconstrained variables in the second, so removing any assumption on them. Hence the model uses generations to move on the right demand level, using constants seed distributions as an indirect constrain to the original demand matrix.

Solution algorithm

For the solution of the first step (2) a Finite Difference Stochastic Approach (FDSA, [27]) has been adopted to find the descent direction. FDSA is a method usually adopted when there is stochasticity in the measurements. At the first step we are mostly interested in investigating the effectiveness of our assumption about the ability of generation values to move the optimization towards the "right level of demand". Hence the choice of using FDSA is done as it permits to obtain at each iteration *i* an exact gradient $\boldsymbol{G}^i$ from a finite-difference computation. Specifically each variable $\boldsymbol{\theta}$ is perturbed as follows:

$$\boldsymbol{G}^i(\boldsymbol{\theta}^i) = \begin{bmatrix} \frac{z(\boldsymbol{\theta}^i + c^i \xi^1) - z(\boldsymbol{\theta}^i)}{c^i} \\ \vdots \\ \frac{z(\boldsymbol{\theta}^i + c^i \xi^r) - z(\boldsymbol{\theta}^i)}{c^i} \end{bmatrix} \tag{3}$$

where ξ is a vector with all zero, except for the variable to be perturbed, c^i is the step and z the adopted objective function. In this method each variable is perturbed independently, so the number of simulations required for computing the gradient in any iteration is equal to the number of variables (in the first step variables are equal to the generated trips from each origin zone O and time interval n) plus the value of z in the starting point.

Once computed $\boldsymbol{G}^i$, the solution is then updated at each iteration by:

$$\boldsymbol{\theta}^{i+1} = \boldsymbol{\theta}^i - \alpha^i \boldsymbol{G}^i \tag{4}$$

with α the step length for the update.

At the second step, given the estimated total generated demand of the first step, the optimization works on dynamic OD trips in a more traditional manner. For the second step, the Simultaneous Perturbation Stochastic Approximation (SPSA, [27-29]) has been adopted. SPSA is a path search optimization method, where an approximation of the gradient is computed based on a simultaneous perturbation of all the variables. In the SPSA, the equation to update the solution is the standard formulation reported in (4), while the approximated gradient at each iteration *i* is obtained as follows:

$$\widehat{\boldsymbol{g}}_k(\boldsymbol{\theta}^i) = \frac{z(\boldsymbol{\theta}^i + c^i \Delta^k) - z(\boldsymbol{\theta}^i)}{c^i} \begin{bmatrix} (\Delta_1^k) \\ \vdots \\ (\Delta_r^k) \end{bmatrix} \tag{5}$$

$$\boldsymbol{G}^i = \overline{\boldsymbol{g}}(\boldsymbol{\theta}^i) = \frac{\sum_{\boldsymbol{k=1}}^{\boldsymbol{Grad\_rep}} \widehat{\boldsymbol{g}}_k(\boldsymbol{\theta}^i)}{Grad\_rep} \tag{6}$$

with c^i the perturbation step, *Grad_rep* is the number of the replications to compute the average gradient and Δ is a vector with elements {-1,1}.

With respect to the FDSA, the gradient has a stochastic component, but the computational time to obtain the descent direction is smaller being the variables perturbed simultaneously. It is possible, and recommended, to repeat the perturbation to obtain a good approximation, given the stochasticity of the gradient approximation method. In the equation above (5), the formulation of the SPSA model is presented with the asymmetric design (SPSA-AD, [25, 30]). The advantage of using this formulation is that the number of simulations needed to compute the gradient is halved with respect to the basic SPSA with symmetric design (SD).

P-SPSA

Since the approach aims to be applicable to real-sized networks, SPSA is appropriate for solving the second step problem. However, although the gradient computation is not dependent on the number of variables, approximation increases with the number of variables *N*:

$$\sum_{\boldsymbol{e=1}}^{\boldsymbol{M}} \frac{\partial z_e(\theta^k)}{\partial \theta^k} \widehat{\boldsymbol{\theta}} = \boldsymbol{G}^{\boldsymbol{SPSA}}(\theta^k) + \boldsymbol{\varepsilon_1}(N) \tag{7}$$

Where *M* is the number of terms in the goal function and $\boldsymbol{\varepsilon_1}(N)$ is the error related to perturbing all the *N* variables simultaneously. A new variant, here called P-SPSA (Partial SPSA) is proposed in the second approach to reduce the approximation of the SPSA with respect to this problem. In every iteration only a percentage *P* of the matrix is perturbed and updated. Elements of the Δ vector are now {-1,0,1}. Therefore by fixing the value of *P*, we regulate the share of non-zero in the Δ vector. The variables to be perturbed are randomly selected in every iteration, so any of them is selected throughout the whole optimization process:

$$\sum_{\boldsymbol{e=1}}^{\boldsymbol{M}} \frac{\partial z_e(\theta^k)}{\partial \theta^k} \widehat{\boldsymbol{\theta}} = \boldsymbol{G}^{\boldsymbol{P-SPSA}}(\theta^k) + \boldsymbol{\varepsilon_1}^{\boldsymbol{k}}(\boldsymbol{N_p}) \tag{8}$$

While it is easy to observe that the error $\boldsymbol{\varepsilon_1}(\boldsymbol{N_p}) < \boldsymbol{\varepsilon_1}(\boldsymbol{N})$ where $N_p < N$ **,** we have to consider that the procedure could converge more slowly with respect to the SPSA since only a part of the variables are updated in an iteration. On the other hand, we know that $0 \leq P \leq 1$, and specifically the computational time is going to increase more and more the closer P gets to 0, while with P=1 is going to become the same of the SPSA. We can consider this problem inserting a second error in (8) i.e.:

$$\begin{cases} \boldsymbol{\varepsilon}(\theta^k) = \boldsymbol{\varepsilon_1}^{\boldsymbol{k}}(N_p) & for\ \theta^k \in N_p \\ \boldsymbol{\varepsilon}(\theta^k) = \boldsymbol{f}(\theta^i - \theta^{i+i}_{spsa}) = \boldsymbol{\varepsilon}^{\boldsymbol{k}}(N - N_p) & for\ \theta^k \notin N_p \end{cases} \tag{9}$$

where N_p is the ensemble of perturbed variables, $\boldsymbol{\varepsilon}^{\boldsymbol{k}}(N - N_p)$ is the error related to not updated variables, θ^{i+1}_{spsa} is the value that the variable θ^k, not updated in the current iteration, assumed in the next iteration when a full SPSA is performed.

If *i* is the number of iterations, we can now assume that if

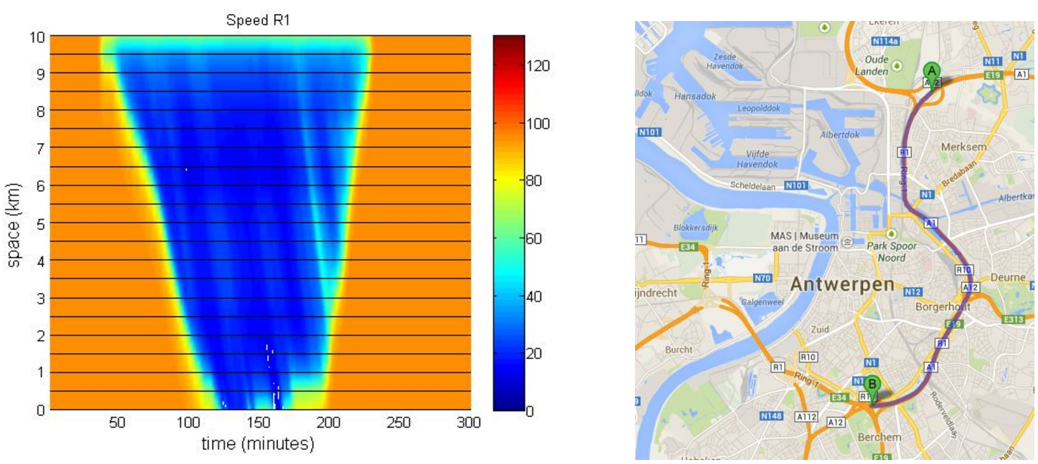
$$\sum_i \sum_{N_p} \boldsymbol{\varepsilon_1}(\boldsymbol{N_p}) + \sum_i \sum_{\boldsymbol{N-N_p}} \boldsymbol{\varepsilon}(\boldsymbol{N} - \boldsymbol{N_p}) \leq \sum_i \sum_{\boldsymbol{N}} \boldsymbol{\varepsilon_1}(N) \tag{10}$$

then the computational time of the P-SPSA is smaller or equal to the time of the SPSA. Since equality in (10) is satisfied for *P*=1 our preliminary assumptions are that very low values of *P* (i.e. 0.25) the term $\boldsymbol{\varepsilon}(\boldsymbol{N} - \boldsymbol{N_p})$ increases much more than the reduction in $\boldsymbol{\varepsilon_1}(\boldsymbol{N_p})$. It is reasonable to assume that opposite holds for high *P* values (i.e. 0.75).

Preliminary results in this paper confirm our hypothesis on P-SPSA, which are relevant on big size networks where solving the problem presented in (7) is well known to be cumbersome. This is shown in a test network in the next section.

CASE STUDY

The test case study is the same presented in Frederix et. al [17], and used in [23], related to the inner ring-way around Antwerp, Belgium. The network includes 56 links, 39 nodes, with 46 OD pairs, all mainly connecting the different entry and exit points of this stretch of motorway, making rerouting options not likely. The considered morning peak period occurs between 05:30 and 10:30. The field data – speeds and flows – were available every 5 minutes. The detectors are located at the on and off-ramps and on some intermediate sections. The OD flows have been estimated for 15-minutes departure intervals, so the dynamic matrix contains 966 OD pairs; the seed matrix that amounts to 202,200 trips is derived from an existing static OD matrix by superimposing a time profile. Flows of a selection of OD pairs have been increased obtaining a congestion pattern similar to the actual one. As a consequence, the seed matrix captures the correct traffic regimes.



(a) (b)

Figure 1: (a) x,t plot of the measured speeds on the network, which is indicated by a blue curve in (b) illustrating the Ring of Antwerp

In order to start with the application of the two-step procedure, with respect to (1) and (2), the objective function to be minimized contains only the z_1 term, where the link measurements are the traffic counts. Specifically:

$$z_1 = \sum_{l \in D} (\boldsymbol{y}_s - \boldsymbol{y}_r)^2 \qquad (11)$$

With $\boldsymbol{y}_s$ and $\boldsymbol{y}_r$ respectively being the simulated and measured flows on each link and D the subset of network links with sensors. The speed measurements have been used only for validation, since it is expected that if the initial traffic regime is accurately represented on any link, then the new estimated matrix reduces the link errors related to flows while preserving the correct traffic regimes. The simulations required to compute simulated flows on each link have been conducted adopting the Link Transmission Model described in e.g. [31-32].

First step application

In the first step, the generation values for each zone and each time interval have been optimized using FDSA. Firstly, an analysis has been conducted on the step α to be adopted in (4). Since the step value of 0.5 well performed the problem when variables were the OD pairs and every generation value is the sum of six OD pairs, the following set of steps values is considered: 0.5, 1, 3. Speeds measurements are not included in the goal function, so the general stop criterion on speeds is set on values implying acceptable traffic regime: RMSE≤ 20; RMSN ≤ 35. For preliminary tests on step size stronger requirements are used setting thresholds values to 15 for RMSE and 23 for RMSN. Results are reported in Table 1-A and Figure 2.

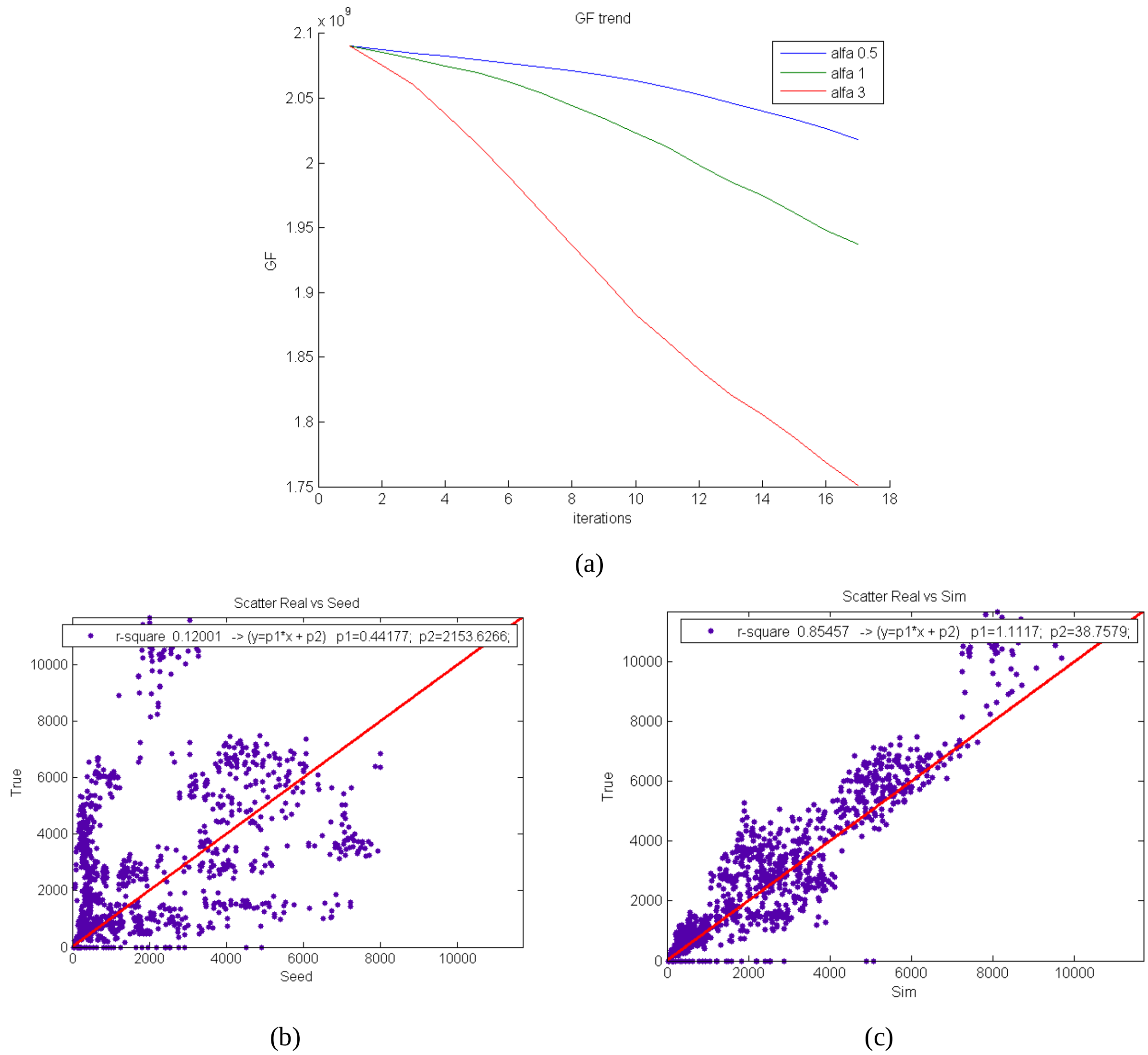
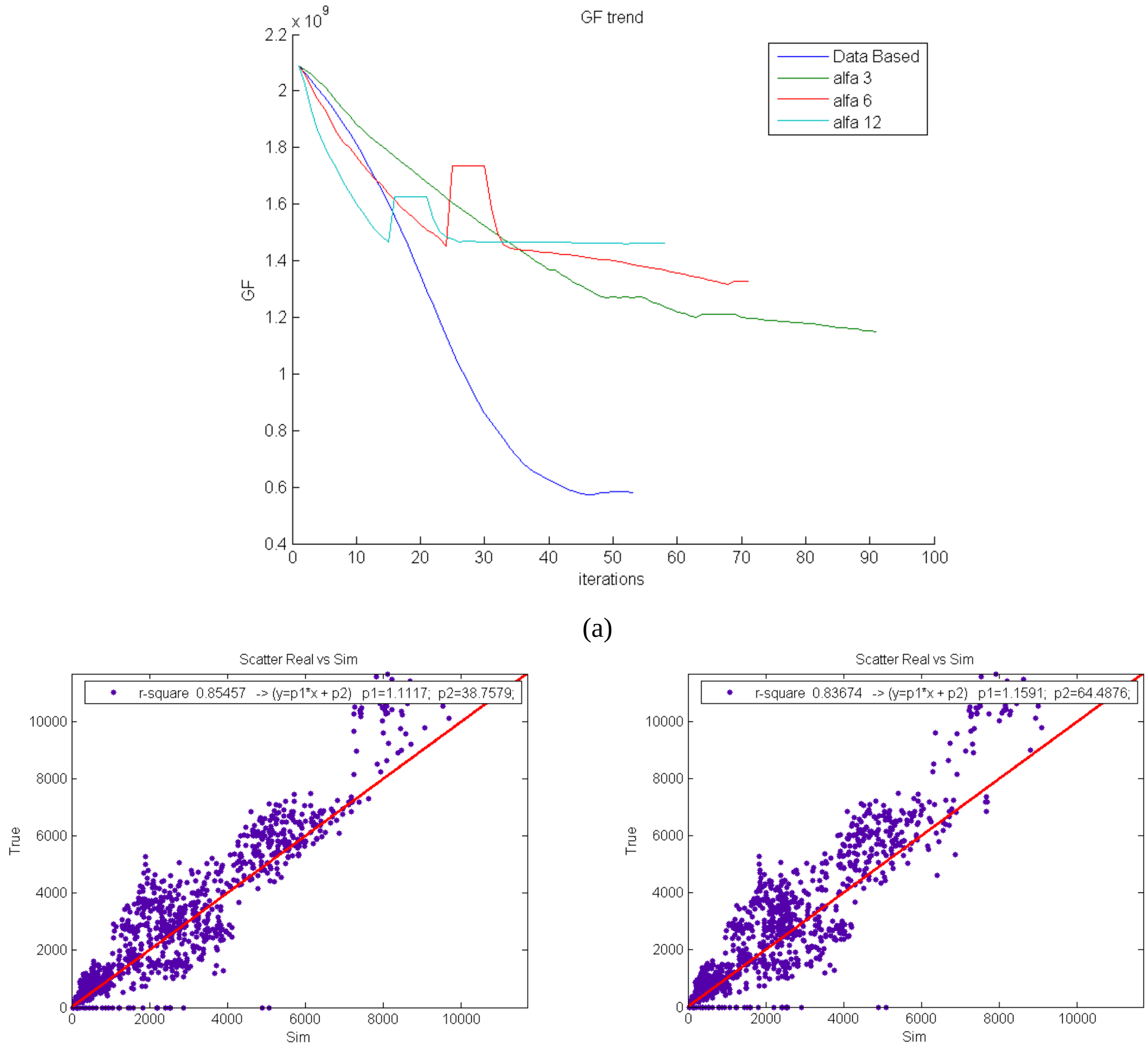


Figure 2: (a) Goal function trend for α=[0.5 1 3], (b) Scatter plot observed vs. simulated speeds for the seed matrix, (c) Scatter plot observed vs. simulated speeds for the solution matrix adopting α=3

As expected, higher step sizes result in an acceleration of the optimization (see Figure 2a). Figure 2b and 2c report respectively the scatter plot between the traffic counts and the simulated flows derived by the assignment of the seed matrix and by the assignment of the matrix obtained using the step α=3.Taking into account that both matrices (seed and estimated one from the first step with α=3) present the right congestion pattern, it is more reasonable to start from the estimated matrix of the first step and then perform the global DDEP.

Then, the full experiment is carried out: generations are corrected in the first step performing a full optimization using increasing values for α: [3 6 12]. Results have been compared with those obtained by the most performing approach to correct the seed matrix reported in Cantelmo et al. [23] (Data based values, Table1-B).



(b) (c)

Figure 3: (a) Goal function trend for α=[3 6 12], (b) Scatter plot observed speed simulated speed for solution matrix using α=3 after 91 iterations, (c) Scatter plot observed speed simulated speed for the solution matrix using α=12

Scatter plots on flows are very similar for different values of α, even if the value of the goal function and the number of iterations is different. In Figure 3c and 3b we can observe scatters for α=3 and α=12: the scatter plot for α=3, after 91 iterations, is quite similar to the one obtained for α=3, after 31 iterations; the r-square changes from 0.838 to 0.854 and p1 from 1.167 to 1.111. Further, all the scatters present the same aggregate characteristics (i.e. the higher errors are on the higher flows). This suggests that the higher contribution of the first step is independent from the starting value of α. Finally, all the obtained matrices present similar scatter plots and the right congestion pattern.

The reference model (Data Based, Table 1-B) implies the best improvement, but as reported in [23] it works on a subset of variables (the OD pairs that generated the highest error with respect to traffic counts).

TABLE 1 – Experiments results

1A – Preliminary Tests	α=0.5	α=1	α=3
Final O.F. value	1.68E+09	1.75E+09	1.51E+09
O.F. improvement [%]	19.75	16.14	27.9
Link flows RMSE	1264.43	1274.1	1181.41
Link flows RMSN [%]	38.56	39.42	36.55
Link speeds RMSE	14.48	14.94	15.43
Link speeds RMSN [%]	22.46	23.19	23.99
# iterations	50	31	31

1B – Full Experiment	α=3	α=6	α=12	Data based
Final O.F. value	1.68E+09	1.33E+09	1.43E+09	5.8E+08
O.F. improvement [%]	45.02	36.48	30.04	72.26
Link flows RMSE	1031.5	1108.9	1163.7	730.3
Link flows RMSN [%]	31.91	34.31	36.55	22.6
Link speeds RMSE	19.00	15.75	17.48	18.47
Link speeds RMSN [%]	29.49	24.44	27.13	28.67
# iterations	91	71	58	53

This sub-set of variables is not easy to capture in all the networks and changing the subset of variables, the quality of results and the computational time can present high variance [23]. In the first step here proposed, there is not this type of problem and the solutions are reliable with respect to the inputs.

Moreover, in this case, in the first step we are working on all the variables, using distributions derived from the seed matrix as a constraint. Thus, it is reasonable to obtain a higher value for the goal function. Final considerations derive from the fact that in this two-step approach it is expected to have the greatest contribution in the second step of the model, where the estimation is done on the disaggregated OD flows

Some remarks can be done about the computational time of the first step. Here we have adopted the FDSA, which is a computationally expensive method. The Sensitivity-Based OD Estimation (SBODE) method of Frederix et al. [9] could represent an alternative solution to reduce the computational times. Using that algorithm it is possible to reach the convergence in two iterations, obtaining a 48.5% improvement of the goal function. This is possible since the model utilizes a line search to obtain the best step size. However, when line search method is used to reduce computational time, it is necessary to add speeds/densities measurements in the goal function, in order to avoid a wrong traffic regime identification.

Second step application

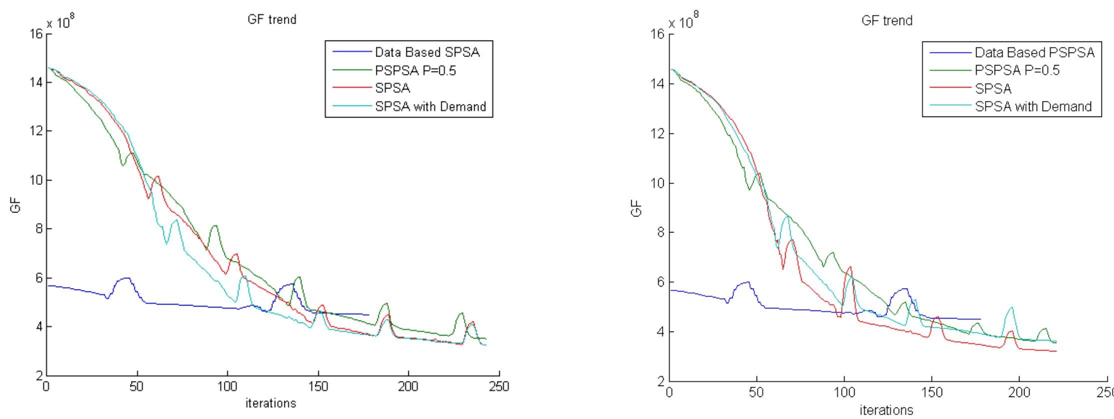
In this second step, the correction is mainly focused on distributions. The experiments are performed adopting as starting matrix the solution obtained using a step size of α=12, tested in the previous stage. Such solution is considered the most interesting case for two main reasons. First of all, it was the configuration for which the convergence has been reached earlier, finding the solution after 30 iterations (Fig. 3a). Since the matrix presents the highest value of the goal function, and since results are robust with respect to both link flows and speeds data, if matrices obtained with α value 3 or 6 are used, then the result should not be worse. Before performing the second step optimization, results from the single step are shown. In table 2-A it is possible to observe results obtained applying SPSA and P-SPSA in a single-step classical DDEP. While for the SPSA the stop criterion is the convergence, P-SPSA is stopped after approximately 190 iterations. Since several single-step SPSA optimizations were performed, in table 2-A “best” represents the best value for each parameter obtaining during the statistical analysis, “worst” the lower value while “avg” is the average solution.

Results suggest that the hypothesis done in (10) about computational time is reasonable: when the perturbation P≥0.5 computational time is not going to increase. Furthermore it is recommend to never use P<0.5: when P is small, the probability to work on all the variables during the optimization largely decreases. P-SPSA results in Table 2-A are experimental, since just one optimization is performed. P-SPSA allows to reduce the number of variables of the problem up to 50%, which is a fundamental property for big-sized networks, without affecting the quality of the results. Since the interest is to apply the two-steps approach to all the networks, both SPSA and P-SPSA are tested. The most interesting goal for the P-SPSA is to reach the same result of SPSA without increasing the computational time, so the case with P=0.5 is considered to perform the second step.

Both models use the same goal function presented in (11). Furthermore, SPSA algorithm is also tested using the demand matrix in the goal function. So equation becomes:

$$z = z_1 + z_3 = \sum_{l \in D} (\boldsymbol{y}_s - \boldsymbol{y}_r)^2 + \sum_{N} (\boldsymbol{x}_i - \boldsymbol{\hat{x}_i})^2 \qquad (12)$$

Where *N* is the number of OD pairs and $\boldsymbol{\hat{x}_i}$ is the target matrix, in this case the solution of the first step. This experiment is called “SPSA with Demand” in the rest of the paper. In figures 4a and 4b goal functions trend are proposed for two independent optimizations. The trend shows again the robustness of the model. Results are compared with the old data-based two steps approach.



(a) (b)

Figure 4: Goal function trend for two different tests (a-b).

Stop criterion is the convergence or an RMSE on the speed lower than 20. Once more, the results highlight the robustness of the process with respect to the Data-Based approach. In fact in the Data Based the main contribution was in the first step, where only 126 OD pairs out of 966 were used. In the second step the model just added local adjustments on the matrix. The main positive results of the Data Based approach are the lower error in the speeds, with respect to the original seed matrix, and the lower number of iterations (Fig. 4). Unfortunately these results are not easily generalizable since the subset is not uniquely defined and if another subset is chosen, results are completely different. In [23] another approach- called “Network Analysis Based Approach - was proposed where a different subset of variables was used. The results were completely different from the Data Based one. If the goal function improvement was greater (89.9%), the error on the speed increased (RMSE=18) as well as the distance from the seed matrix (equal to 6.26 E+04 in Data Based and 1.15E+05 in the Network Analysis Based). The strong difference between results was the initial input to generate the current approach.

TABLE 2 Final DDEP results

2A – Single Step Results	P-SPSA P=0.25	P-SPSA P=0.50	P-SPSA P=0.75	SPSA BEST	SPSA AVG	SPSA WORST
Final O.F. value	6.67E+08	3.86E+08	3.93E+08	3.28E+08	3.96 E+08	5.01 E+08
O.F. improvement [%]	68.07	81.51	81.18	84.29	81.40	76.04
Link flows RMSE	786.29	601.35	602.75	552.03	598.44	681.79
Link flows RMSN [%]	24.32	18.60	18.60	16.58	18.39	21
Link speeds RMSE	18.94	18.42	18.63	17.59	19.30	21.01
Link speeds RMSN [%]	29.29	28.59	28.91	27.47	30.52	34.47
# of iterations	187	195	195	90	160	273

2B – Two Steps Results	SPSA	SPSA demand	P-SPSA P=0.5	Data based SPSA	Statistics results BEST	Statistics results AVG	Statistics results WORST
Final O.F. value	3.29E+08	3.18E+08	3.08E+08	4.49E+08	3.08E+08	3.23E+08	3.5E+08
O.F. improvement [%]	84.25	84.77	85.24	78.52	85.24	84.54	84.51
Improvement in 2th [%]	77.25	78.18	78.85	20.77	78.85	77.85	75.72
Link flows RMSE	547.42	538.20	534.75	644.82	534.34	545.76	571.92
Link flows RMSN [%]	16.93	16.65	16.53	19.95	16.53	16.88	17.69
Link speeds RMSE	19.98	18.44	17.29	13.67	16.22	18.69	20.7
Link speeds RMSN [%]	34.71	28.44	26.83	21.16	25.41	29.64	34.71
Regression coefficients					*Regression coefficients*		
r2	0.936	0.937	0.939	0.920	0.939	0.936	0.936
Angular coefficient p1	0.99	1.00	0.99	0.97	1	1	0.997
Intercept coefficient p2	49.02	43.00	43.21	103.11	34.74	44.25	50.25

In this approach we can observe the advantages having a uniquely defined subset of variables in the first step. Results for each method are very close to each other. Moreover scatter plots of the results are very similar to each other: the parameters of the regression (r2, p1, p2) are very similar. About P-SPSA it is possible now to make some remarks. The main goal of P-SPSA it is to reach the same solution of SPSA without increasing computational time whilst reducing the number of variables. In (10) we assume that if the number of variables perturbed in every iteration is at least the 50% computational time is not going to increase. Tests show that P=0.5 is, as expected, the limit case using P-SPSA. If the computational time is higher than the one of SPSA, such increase is limited. Setting as stop criterion the number of iterations, the goal function value at iteration 243 is 3.50E+08, while at iteration 269 is 3.32E+08. Results show that the approximation is not going to reduce the quality of the result. This conclusion is important in real networks, where the number of variables is too high to use in an efficient way SPSA. P-SPSA is an appropriate alternative to manage problems two times bigger with respect to classic SPSA without compromising significantly the quality of the solution and the computational time.

Finally, some considerations have to be done with respect to the comparison with single step approach. Observing table 2, differences in results are significant. In 2-B the procedure better fits measured data than the single step approach, as calibration parameters confirm. Furthermore, a strong reduction in variance results is observed. In the first case difference between the best and worst goal function value is almost 10%, while in the second case is approximately 1%. If variance in some parameters, like iterations number, seems to be good, these parameters are generally related to the worst cases. The number of iterations of the “best” case in table 2-A is lower than those reported in figure 4. However, when convergence is reached too fast, model results in high goal functions values and not satisfactory solutions. Further, regression coefficients are worst with respect to the two steps approach (r2=0.934, p1=0.98, p2=71 for the best solution). About Euclidean distance from the seed matrix, the average value is similar in both cases while the distance between each solution matrix is different. The average distance between solutions matrices found using two step approach is 3.42E+04, while is 4.35E+04 in the single step. Further the variance of this value is higher in the single step with respect to the proposed approach, confirming robustness of our method with respect to the single step.

CONCLUSIONS AND FUTURE RESEARCH

In this paper a two steps approach is proposed to improve performances of existing DDEP algorithms. Since the reliability of the results in dynamics problem is one of the most critical aspects in using dynamics methods for real problems, the main contribution of this approach is finding robust results with respect to both the single-step approach and the previous version of the Two-Steps approach. In the paper a combination of deterministic and stochastic algorithms is used to perform offline estimation on the inner ring of Antwerp, Belgium. Speeds are used to validate quality of the solution and as stop criterion.

The main motivation in developing the proposed approach is obtaining accurate and reliable results by operating an adequate solution space reduction. Since the number of possible solutions generally increases with the size and the complexity of the network, it is relevant introducing general procedures to reduce step by step the solution space without increasing the problem complexity. The two steps approach is based on the correlation between the aggregate demand data – named generation data -, the disaggregate demand data – i.e. the OD flows – and supply data as link speeds and flows. Since, generally, aggregate data from statics models are more reliable with respect to the disaggregate one, it is natural to fix them in an aggregate level.

Following a two-steps procedure, as initially proposed in previous studies by the authors, in the first step the total flow generated for each traffic zone is corrected. The demand at aggregate level can be used to catch the right demand level keeping constant the distributions. In this first phase, distributions are used as an indirect constraint for the demand, reducing the possible solutions for the problem without introducing new measurements or data. Vice versa, since aggregate data works as an indirect constraint, it is possible to eliminate the demand term from the goal function. In this way it is possible to strongly reduce the localism of the DDEP. Results show the reliability of the approach with respect to the most important parameter, the step size. Is it so possible to increase the speed of the problem without having significant errors in the solution of the first step.

In the second step, correction of the demand is performed using SPSA algorithm obtaining good results. The used method is generally adopted to solve problem on big sized networks, since it is not dependent on the number of variables. On the other hand the stochasticity of the model increases with the size of the problem. In the specific case study, SPSA obtains stable results. A variant of that model, called P-SPSA, is presented in this paper. It should be pointed out that results are experimental and preliminary, since no test on other networks are still available; the model was tested together with the SPSA in the second step. P-SPSA reaches, in the case study, the same result of the SPSA, while working on no more than 50% of the OD pairs simultaneously. So in the current case study we are able to perform a full satisfactory OD estimation reducing the number of the variables to the only generation in the first step and to the 50% of the OD pairs in the second. The possibility to reduce number of variable is one of the most relevant aspects in DDEP, since often in real practice is not possible to work on all of them. Results highlight the robustness of the proposed approach with respect to the classical single step.

Future research will still focus on small networks where however route choice is more significant than the network used in this paper. If results are confirmed the last step is to apply it on medium/large sized networks.

REFERENCES

[1] Cascetta E., Inaudi D. and G. Marquis (1993). Dynamic Estimators of Origin-Destination Matrices using Traffic Counts. *Transportation Science* ***27***, 363-373.

[2] Balakrishna, R. (2006). Off-line calibration of dynamic traffic assignment models. PHD thesis. Massachusetts Institute of Technology.

[3] Dixon, M. and L.R. Rilett (2002). Real-Time OD Estimation Using Automatic Vehicle Identification and Traffic Count Data. *Computer-Aided Civil and Infrastructure Engineering* ***17*** *(2002),* 7–21

[4] Eisenman, S.M., List, G.F. (2004). Using probe data to estimate OD Matrices. *Intelligent Transportation Systems Conference. Washington DC October 3-6*: 291-296.

[5] Caceres, N., Wideberg, J.P. and F.G. Benitez (2007). Deriving origin–destination data from a mobile phone network. *IET Intell Transp. Syst., 2007,* ***1****(1)*, pp. 15–26.

[6] Barceló, J., Montero, L., Bullejos, M., Serch, O., Carmona, C. (2012). Dynamic OD Matrix Estimation Exploiting Bluetooth Data in Urban Networks, Recent Researches in Automatic Control and Electronics, *ISBN*: 978-1-61804-080-0.

[7] Mitsakis, E., Salanova, J.M., Chrysohoou, E., Aifadopoulou, G.(2013). A robust method for real-time estimation of travel times for dense urban road networks using point-to-point detectors. Proceedings of the 92nd Annual Meeting in Transportation Research Board, TRB 2013.

[8] Cascetta, E., Papola, A., Marzano, V., Simonelli, F. and I. Vitiello (2013). Quasi-dynamic estimation of o-d flows from traffic counts: formulation, statistical validation and performance analysis on real data. *Transportation Research Part B*, doi 10.1016/j.trb.2013.06.007.

[9] Frederix, R., Viti, F., Corthout, R., Tampère, C. M. J. (2011). New Gradient Approximation Method for Dynamic Origin-Destination Matrix Estimation on Congested Networks. *Transportation Research Record, 2263*:19-25.

[10] Toledo, T., Kolechkina, T. (2012). Estimation of Dynamic Origin–Destination Matrices Using Linear Assignment Matrix Approximations. *IEEE Transactions On Intelligent Transportation Systems*. Digital Object Identifier 10.1109/TITS.2012.2226211.

[11] Ashok, K. and Ben-Akiva, M.E. (2002). Estimation and prediction of time-dependent origin-destination flows with a stochastic mapping to path flows and link flows. *Transportation Science*, **36**, 184–198.

[12] Zhou X and H. Mahmassani (2007). A structural state space model for real-time traffic origin–destination demand estimation and prediction in a day-to-day learning framework. *Transportation Research B*, **41**, 823-840

[13] Ashok, K. and M.Ben-Akiva (2000). Alternative approaches for real-time estimation and prediction of time-dependent origin-destination flows. *Transportation Science* ***34***, 21-36.

[14] Balakrishna, R., Koutsopoulos, H. N. and M. Ben-Akiva (2005). Calibration and validation of dynamic traffic assignment systems. Proceedings of 16th ISTTT, 407-426.

[15] Yang, H., 1995. Heuristic algorithms for the bilevel origin–destination matrix estimation problem. *Transportation Research B*, **29**, 231–242.

[16] Tavana, H., 2001. Internally-consistent estimation of dynamic network origin–destination flows from intelligent transportation systems data using bi-level optimization. *Ph.D. thesis, Department of Civil Engineering, University of Texas at Austin*, Austin, Texas.

[17] Frederix, R. (2012). Dynamic Origin-Destination Matrix Estimation in Large-Scale congested networks. PHD thesis. Katholieke Universiteit Leuven

[18] Antoniou C., Balakrishna R., Koutsopoulos H.N. and M. Ben-Akiva (2009). Off-Line and On-Line Calibration of Dynamic Traffic Assignment Systems. Presented at the 12th IFAC Symposium on Control in Transportation Systems.

[19] Cipriani E., Florian M., Mahut M. And M. Nigro (2011). A gradient approximation approach for adjusting temporal origin–destination matrices. *Transportation Research C*, **19**(3), 270-282

[20] Djukic, T., van Lint, J. W. C. and Hoogendoorn, S. P. (2012). Application of principal component analysis to predict dynamic origin-destination matrices, *Transportation Research Record* **2283**(81-89).
[21] Flötteröd, G. and Bierlaire, M. (2009). Improved estimation of travel demand from traffic counts by a new linearization of the network loading map. *Proceedings of the European Transport Conference*.
[22] Frederix, R. Tampère, C. Viti, F. (2013). Dynamic Origin-Destination Matrix Estimation in congested networks: Theoretical findings and implication in practice. *Transportmetrica* **9**(6), pp. 494-513.
[23] Cantelmo G., Viti F. Cipriani E., Nigro M.L. (2014) Two-Step Approach for Correction of Seed Matrix in Dynamic Demand Estimation. Proceedings of the 93nd Annual Meeting in Transportation Research Board, TRB 2014.
[24] Iannò, D., Postorino, M.N. (2002). A generation constrained approach for the estimation of O/D trip matrices from traffic counts. Proceeding in conference: Association for European Transport 2002,
[25] Cipriani E, Florian M, Mahut M, Nigro M (2010). Investigating the Efficiency of a Gradient Approximation Approach for Solution of Dynamic Demand Estimation Problem. *New Developments in Transport Planning: Advances in Dynamic Transport Assignment* (Transport Economics, Management and Policy Series). Edward Elgar Publishing, ISBN: 1848449631
[26] Cantelmo G, Cipriani E, Gemma A, Nigro M (2014). An adaptive bi-level gradient procedure for the estimation of dynamic traffic demand. *IEEE transaction on intelligent transport systems*, ISSN: 1524-9050, doi: 10.1109/TITS.2014.2299734
[27] Spall, J. (2012). Stochastic Approximation. *Handbook of Computational Statistic: Concepts and Methods* (2nded.) (J.Gentle, W. Hardle, Y. Mori, eds.), Springer-Verlang, Heidenberg, Chapter 7, pp. 173-201.
[28] Spall, J. (2000). Adaptive Stochastic Approximation by the Simultaneous Perturbation Method. *IEEE transactions on automatic control*, **45**, No. 10
[29] Spall, J. (1998). An Overview of the Simultaneous Perturbation Method for Efficient Optimization. *Johns Hopkins Apl Technical Digest*, **19**(4), pp 482-492
[30] Nigro, M.(2009). Correzione della domanda di trasporto in dinamica intraperiodale con l'ausilio di differenti fonti di dati. PHD thesis. Università degli studi Roma Tre.
[31] Corthout, R. (2012). Intersection modelling and marginal simulation in macroscopic dynamic network loading. PHD thesis. Katholieke Universiteit Leuven
[32] Yperman, I. (2007). The Link Transmission Model for Dynamic Network Loading. PHD thesis, Katholieke Universiteit Leuven.