Transport and Telecommunication, 2013, Volume 14, No 1, 1–12
Transport and Telecommunication Institute, Lomonosova 1, Riga, LV-1019, Latvia
DOI 10.2478/ttj-2013-0001

EXPLORATORY ASSESSMENT OF THE LIMITING EXTENDED KALMAN FILTER PROPERTIES

Constantinos Antoniou, Alexandra Kondyli, Georgia-Maria Lykogianni and Elias Gyftodimos

National Technical University of Athens, Greece, School of Rural and Surveying Engineering
Laboratory of Transportation Engineering
Zografou, GR-15780, Greece
Ph.: +30 210 7722783. Fax: +30 210 7722629
E-mail: antoniou@central.ntua.gr, akondyli@gmail.com, gmlikog@gmail.com
elias.gyftodimos@gmail.com

Most of the methodologies for the solution of state-space models are based on the Kalman Filter algorithm (Kalman, 1960), developed for the solution of linear, dynamic state-space models. The most straightforward extension to nonlinear systems is the Extended Kalman Filter (EKF). The Limiting EKF (LimEKF) is a new algorithm that obviates the need to compute the Kalman gain matrix on-line, as it can be calculated off-line from pre-computed gain matrices. In this research, several different strategies for the construction of the gain matrices are presented: e.g. average of previously computed matrices per interval per demand level and average of previously computed matrices per interval independent of demand level.

Two case studies are presented to investigate the performance of the LimEKF under the different assumptions. In *the first case study*, a detailed experimental design was developed and a large number of simulation runs was performed in a synthetic network. The results suggest that indeed the LimEKF algorithm is robust and – while not requiring the explicit computation of the Kalman gain matrix, and thus having vastly superior computational properties – its accuracy is close to that of the "exact" EKF. In *the second case study*, a smaller number of scenarios is evaluated using a real-world, large-scale network in Stockholm, Sweden, with similarly encouraging results. Taking the average of various pre-computed Kalman Gain matrices possibly reduces the noise that creeps into the computation of the individual Kalman gain matrices, and this may be one of the key reasons for the good performance of the LimEKF (i.e. increased robustness).

Keywords: Origin-destination flows' estimation and prediction, state-space models, Extended Kalman Filter (EKF), Limiting Extended Kalman Filter (LimEKF)

Introduction

Effective management of large-scale transportation systems requires the support of detailed traffic simulation models. Such traffic simulation models involve a large number of parameters and input data that must be calibrated to match surveillance data. Depending on the scope, calibration can be either performed off-line or on-line. The role of off-line calibration is to develop a historical database that supports the simulation model's ability to replicate average or expected traffic conditions as captured by archived surveillance data. The output of this process is expected to perform adequately in off-line, planning-level studies, such as the evaluation of alternative traffic management strategies. Traffic conditions are affected by a number of exogenous factors, including incidents and weather conditions. In order to capture these fluctuations, therefore, the off-line, "average", parameters must therefore be updated dynamically [1]. The real-time performance of the traffic simulation models could therefore be significantly enhanced by re-calibrating the a priori parameter estimates using the available surveillance data.

Off-line and on-line calibration has similar objectives, and a general formulation that encompasses both has been proposed by Antoniou et al. [2]. Off-line calibration is a global optimisation step aiming to determine the mean values of the parameters and inputs, while on-line calibration adjusts the off-line values to reflect prevailing conditions. Data from prior days may be available for off-line calibration, while on-line calibration will generally use smaller amounts of recent, real-time data. Further, off-line calibration has no explicit computational constraints, while on-line calibration has to be faster than real-time. Finally, off-line calibration must estimate all model inputs and parameters, while on-line calibration could focus on a subset of critical parameters. The remainder of this paper focuses on on-line calibration.

Several approaches for on-line calibration of dynamic traffic simulation models have been proposed. Tavana and Mahmassani [3] use transfer functions (bivariate time series models) to estimate dynamic speed-density relations from detector data, while Huynh et al. [4] extend this work by using transfer functions in a simulation-based DTA framework. Qin and Mahmassani [5] evaluate the same model with sensor data from several links in Irvine, CA. Antoniou et al. [6] formulate the problem of on-line calibration of the speed-density relationship as a flexible state-space model and present solution approaches, validating and comparing three of the presented solutions [Extended Kalman Filter (EKF), Iterated EKF, and Unscented Kalman Filter (UKF)]. Wang and Papageorgiou [7] and Wang et al. [8, 9] present a general approach for real-time freeway traffic state estimation, with detailed case studies. Ashok and Ben-Akiva [10, 11] formulate the real-time OD estimation and prediction problem as a state-space model and solve it using Kalman Filters. The approach has been implemented in DynaMIT [12–14]. Bierlaire and Crittin [15] present an efficient solution algorithm for OD estimation, while Zhou and Mahmassani [16] develop similar Kalman-filter based adaptive OD estimation and prediction procedures using a polynomial trend filter to recursively capture demand deviations from a priori demand estimates. Antoniou [17] develops an approach that jointly formulates the on-line calibration problem as a state-space model comprising transition and measurement equations. Applicable solution algorithms are presented and compared in Antoniou et al. [18].

The remainder of this paper is structured as follows. The next section provides some background to the problem, introducing the Origin-Destination estimation problem, while the following section outlines the methodology, leading to and presenting the optimisation algorithms that are used to solve the state-space formulation of the OD estimation and prediction problem. Two case studies are presented in the following section. The first one includes several experiments undertaken in a synthetic network, followed by a case study using a real-world large-scale network. The concluding section provides an overview and discussion of the results obtained, as well as future research directions.

Background

The proposed methodology builds upon the OD estimation and prediction framework presented by Ashok and Ben-Akiva [10, 19] and formulated as a state-space model. A state-space model is formulated as a set of:

- Transition equations; and
- Measurement equations.

Denoting by $\mathbf{X}_h$ the vector representing the number of vehicles between each OD pair departing their origins during time interval h, the transition equation can be expressed in matrix form as:

$$\hat{\mathbf{X}}_{h+1} = \sum_{p=h-q}^{h} \mathbf{f}_h^p \mathbf{X}_p + \mathbf{w}_h \,, \tag{1}$$

where $\hat{\mathbf{X}}_{h+1}$ is an estimate of $\mathbf{X}_{h+1}$, $\mathbf{f}_h^p$ is the matrix of effects of $\mathbf{X}_p$ on $\hat{\mathbf{X}}_{h+1}$, $\mathbf{w}_h$ is a vector of Gaussian, zero-mean, uncorrelated errors, and q is the degree of the autoregressive process.
The measurement equation, which relates unknown OD flows to observed link counts, can thus be stated in matrix form as follows:

$$\mathbf{Y}_h = \mathbf{A}_h \mathbf{X}_h + \mathbf{v}_h \,, \tag{2}$$

where $\mathbf{A}_h = \mathbf{a}_h^h$ is the assignment matrix mapping the drivers that departed in interval h to the link counts observed in interval h and $\mathbf{v}_h$ is the vector of measurement errors, assumed to be zero-mean, uncorrelated as well as uncorrelated with the transition errors $\mathbf{w}_h$.

Methodology

Most of the methodologies for the solution of state-space models are based on the Kalman Filter algorithm [20], developed for the solution of linear, dynamic state-space models. The most straightforward extension to nonlinear systems is the Extended Kalman Filter (EKF, shown in Algorithm 1), which employs a linearization of the non-linear relationship [21]. This algorithm can be applied to simulation-based systems (which do not have an analytical representation) by performing the linearization step using numerical derivatives. The use of numerical techniques requires a large number of function evaluations (each implying a run of the simulator) thus making this technique potentially computationally expensive.

Initialisation

$X_{0|0} = X_0$

$P_{0|0} = P_0$

For h = 1 to N do

Time update

$X_{h|h-1} = F_{h-1} X_{h-1|h-1}$

$P_{h|h-1} = F_{h-1} P_{h-1|h-1} F_{h-1}^T + Q_h$

Linearization

$$H_h = \left. \frac{\partial h(x^*)}{\partial x^*} \right|_{x^* = X_{h|h-1}}$$

Measurement update

$G_h = P_{h|h-1} H_h^T \left(H_h P_{h|h-1} H_h^T + R_h \right)^{-1}$

$X_{h|h} = X_{h|h-1} + G_h \left[Y_h - h\left(X_{h|h-1} \right) \right]$

$P_{h|h} = P_{h|h-1} - G_h H_h P_{h|h-1}$

End for

Algorithm 1. Extended Kalman Filter (EKF)

In Antoniou [17] and Antoniou et al. [18] the Limiting EKF (LimEKF, shown in Algorithm 2), a new algorithm that obviates the need to compute the Kalman gain matrix on-line was proposed.

Generation of *limiting* Kalman gain matrix G and H

Initialisation

$X_{0|0} = X_0$

$P_{0|0} = P_0$

For h = 1 to N do

Time update

$X_{h|h-1} = F_{h-1} X_{h-1|h-1}$

$P_{h|h-1} = F_{h-1} P_{h-1|h-1} F_{h-1}^T + Q_h$

Measurement update

$X_{h|h} = X_{h|h-1} + G \left[Y_h - h\left(X_{h|h-1} \right) \right]$

$P_{h|h} = P_{h|h-1} - GHP_{h|h-1}$

End for

Algorithm 2. Limiting Extended Kalman Filter (EKF)

The performance of that approximate algorithm proved very close to the "exact" EKF and raised interest for the approach. In that work, the off-line computed Kalman gain matrix was the average of all available pre-computed such matrices. In this research, several different strategies for the construction of the gain matrices are presented: e.g. average of previously computed matrices per interval per demand level and average of previously computed matrices per interval independent of demand level.

Platform Implementation

The experimental framework was implemented as a standalone Java application, consisting of three main components: a) An implementation of the parameter estimation algorithm; b) a wrapper of the traffic simulation; c) the main module which initialises the experimental set-up and orchestrates the execution of simulation and estimation for the appropriate iterations.

The parameter estimation currently includes implementations of the EKF and Limiting EKF algorithms, and is designed to be modular and easily extensible. The input to the EKF algorithm includes the initial values of the $\boldsymbol{X}$ vector, which correspond to the demand values in the traffic network. Also given are the matrices $\boldsymbol{F}$, $\boldsymbol{P}$, $\boldsymbol{Q}$ and $\boldsymbol{R}$, and the vector $\boldsymbol{Y}$ corresponding to the network flow measurements. The output of the algorithm implementation is mainly the estimated vector $\boldsymbol{X^*}$ and the simulated $\boldsymbol{Y^*} = f(\boldsymbol{X^*})$. The implementation of the Limiting EKF is similar, and takes the additional input of the pre-calculated $\boldsymbol{H}$ matrix.

The simulation wrapper implementation abstracts the execution of the traffic simulation software as a function $\boldsymbol{Y} = f(\boldsymbol{X})$. The orchestration module implements the main method, which reads the configuration of the experiment (specifying which estimation algorithm is used, providing the estimation parameters, defining the parameters of the mezzo simulation). The estimation algorithm is executed for the specified time intervals, and the estimation output is persisted to the file system for further processing.

Mezzo [22, 23] is mesoscopic traffic simulation software developed by the Centre for Traffic Research of the Stockholm Royal Institute of Technology (KTH). It allows the definition of a road network over which aggregated traffic is simulated on the level of network links, according to specified values of traffic demand. The software facilitates the execution of simulations in sequential time slices with the possibility of varying the demand for each of the time intervals. Mezzo supports modelling of traffic features such as traffic signals and road incidents. The software can be executed either with a graphical user interface allowing user interaction and step-by-step monitoring of the simulation process, or in batch mode as a standalone application. The simulation results are logged in appropriately formatted text files containing relevant information such as the (average) link travel times, origin/destination outputs, speeds and densities. Mezzo is developed in C++ and is free and open source under Gnu Public License v. 3.

Case Study 1: Synthetic Network

A detailed experimental design was developed and a large number of simulation runs was performed. In order to be able to explore the specific properties of the Gain matrices in a practical way, a small synthetic network is considered, with varying levels of demand and number of OD pairs. A simulation framework is developed, that implements the various algorithms and uses the Mezzo mesoscopic traffic simulator.

Synthetic Network Description

For the purposes of this research, a simple synthetic network was developed to explore the specific properties of the EKF and Limiting EKF algorithms. The synthetic network which was modeled in Mezzo simulator is comprised of ten links and three OD pairs (one origin and three destinations). The simulation would run for two hours (eight consecutive 15-min intervals), where the first hour was considered to be the warm-up period. To avoid congestion and spillbacks into the network links, the input demand was set at medium levels. The demand during the warm-up period remained constant, and it varied slightly at each subsequent interval. In this synthetic network it was assumed that detector data (flows) were available on two of its links.

Experimental Design

Several scenarios were developed in this research to evaluate the EKF and the different Limiting EKF variants using the synthetic network described earlier. More specifically, one of the parameters that varied throughout the simulations was the OD demand. Five levels of the demand were evaluated; correct knowledge of the OD demand, and its variation by ±10% and ±20%. Additionally, to account for the variability in real life data that is typically induced due to detector errors or misreading, we further assumed that the traffic counts on the two links would vary by 0%, 10%, and -10%. Table 1 summarizes the different scenarios that were evaluated through the use of the synthetic network.

Table 1. Different scenarios and parameter levels for the EKF and Limiting EKL algorithm evaluation

	OD demand variation	Traffic counts variation
Scenario 1	0%	0%
Scenario 2	0%	+10%
Scenario 3	0%	-10%
Scenario 4	+10%	0%
Scenario 5	+10%	+10%
Scenario 6	+10%	-10%
Scenario 7	-10%	0%
Scenario 8	-10%	+10%
Scenario 9	-10%	-10%
Scenario 10	+20%	0%
Scenario 11	+20%	+10%
Scenario 12	+20%	-10%
Scenario 13	-20%	0%
Scenario 14	-20%	+10%
Scenario 15	-20%	-10%

Imperative in the estimation of the Kalman Gain matrix ($\boldsymbol{G_h}$) is the measurement matrix $\boldsymbol{H_h}$, which is obtained through an intermediate linearization step (see Algorithm 1). For the EKF algorithm, a 2% of the linear delta factor was assumed, indicating the percentage of the demand to be considered during the linearization step.

To overcome the computationally intensive step of the linearization in the EKF algorithm, the Kalman gain matrix $\boldsymbol{G_h}$ is replaced with its limit **G** (Chui and Chen, 1999), for the Limiting EKF algorithm. Thus, the main component of the gain matrix (i.e. the time-dependent matrix $\boldsymbol{H_h}$) is replaced herein with an average matrix $\boldsymbol{H}$ of a number of available matrices. We distinguish three cases for computing the average matrix $\boldsymbol{H}$: (a) it is the average matrix for each time interval for the scenarios with varying OD demand ±10%, (b) it is the average matrix for each time interval for the scenarios with varying OD demand ±20%, and (c) it is the average matrix across all time intervals for the scenarios with varying OD demand ±10%.

Furthermore, for the implementation of the Limiting EKF it was further assumed that the covariance matrices $\boldsymbol{Q_h}$ and $\boldsymbol{R_h}$ are identity matrices.

Results

The results suggest that indeed the Limiting EKF algorithm is robust and – while not requiring the explicit computation of the Kalman gain matrix, and thus having vastly superior computational properties – its accuracy is close to that of the "exact" EKF. Taking the average of various pre-computed Kalman Gain matrices possibly reduces the noise that creeps into the computation of the individual Kalman gain matrices, and this may be one of the key reasons for the good performance of the Limiting EKF (i.e. increased robustness).

This section demonstrates the ability of the two algorithms to estimate traffic flow conditions in a synthetic network. To evaluate the fit of the estimated and the simulated counts, two measures of effectiveness were considered, namely the normalized root mean square error (RMSN) and the mean percentage error (MPE).

$$RMSN = \frac{\sqrt{N\sum_{n=1}^{N}\left(Y_n^s - Y_n^o\right)}}{\sum_{n=1}^{N} Y_n^o}, \tag{3}$$

$$MPE = \frac{1}{N}\sum_{n=1}^{N}\left[\frac{Y_n^s - Y_n^o}{Y_n^o}\right], \tag{4}$$

where N is the number of observations, Y_n^s is the estimated observation and Y_n^o is the corresponding simulated traffic count.

Figure 1 summarizes the RMSN results for all four intervals of the 15 scenarios tested in this case study, considering also the three alternative options used for the calculation of the Kalman Gain matrix of the Limiting EKF algorithm. Figure 2 presents the corresponding MPE results. The "no-calibration" case is presented to indicate the starting point, while the EKF version presents a reference performance of the computationally intensive "exact" algorithm. The three Limiting EKF variants (depending on the way that the ***H*** matrix was pre-computed) are also presented. While the various scenarios are not really connected – and therefore a bar plot would be a more appropriate graphical representation – lines are used in this figure to provide a visually clearer presentation of the various cases.

The results suggest that, as expected, the EKF algorithm performs well, and it manages to provide a satisfactory correction of the initial erroneous values of the parameters. These trials demonstrate the utility of the EKF algorithm in the OD estimation and prediction problems. It is also interesting to note that, although the Limiting EKF algorithm eliminates the computationally intensive steps of the EKF algorithm, it does not lack significantly in terms of accuracy in its estimation. The results of the Limiting EKF algorithm for all scenarios presented on Figures 1 and 2 show that in some cases the simpler Limiting EKF algorithm provides accuracy comparable to that of the EKF algorithm. In some scenarios (such as Scenario 12, in which the true demand has been perturbed by +20% while the true surveillance measurements have been perturbed by -10%, thus "misleading" the calibration process) the EKF does not provide an improvement (and even provides a deterioration from the starting point). The Limiting EKF, on the other hand, which uses an average of formerly calibrated ***H*** matrices, manages to clearly outperform the EKF in this case. Overall, it is noted that the computation of the ***H*** matrix used in the Limiting EKF algorithm plays an important role in the accuracy of the estimation results. More specifically, the results indicate that the computation of the ***H*** matrix used in LimEKFa (average matrix for each time interval for the scenarios with varying OD demand ±10%) and LimEKFc (average matrix across all time intervals for the scenarios with varying OD demand ±10%) performed better than in the case of the LimEKFb (average matrix for each time interval for the scenarios with varying OD demand ±20%). Actually, the latter case seemed to perform rather poorly in a few situations. The approach of the average values that was introduced in this research manages to eliminate the noise in the simulations, without producing adverse estimation results. The exact values of the results are presented in a table in Appendix A.

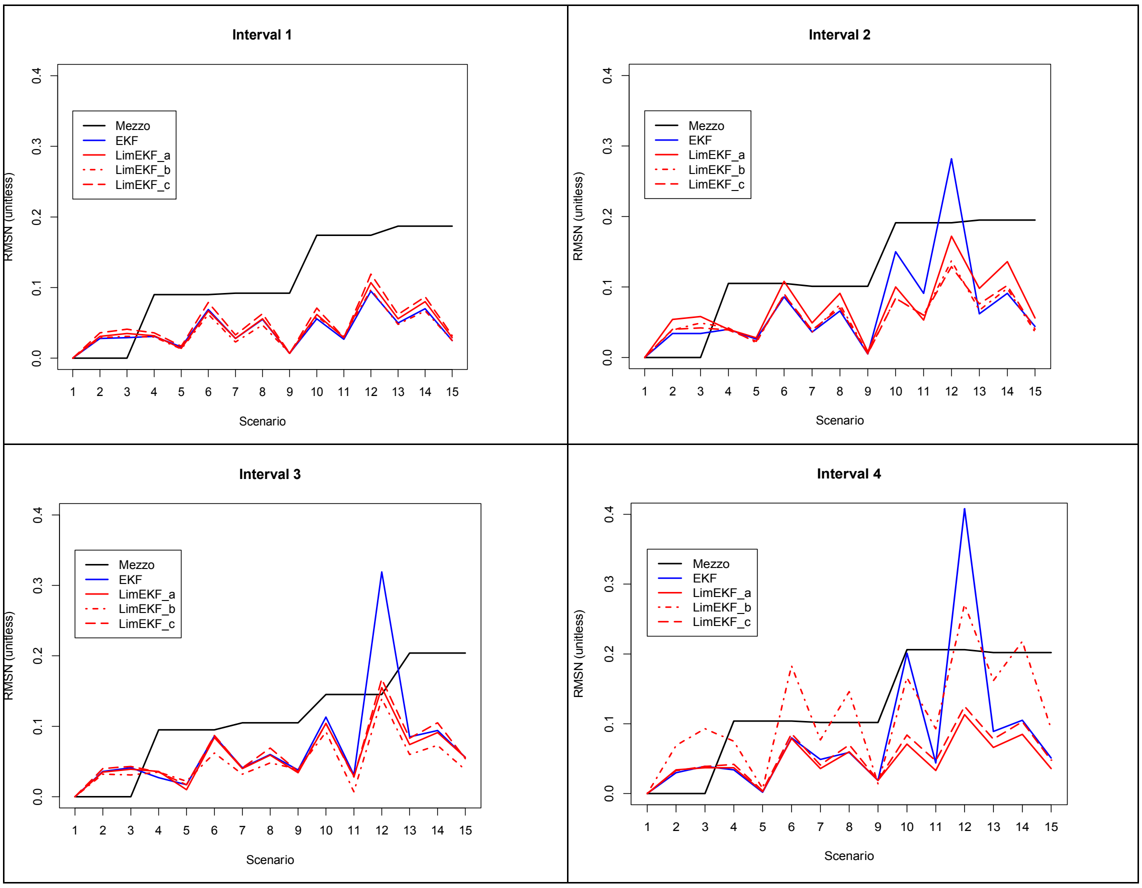

Figure 1. Overview of the calibration results of the synthetic network based on the RMSN statistic

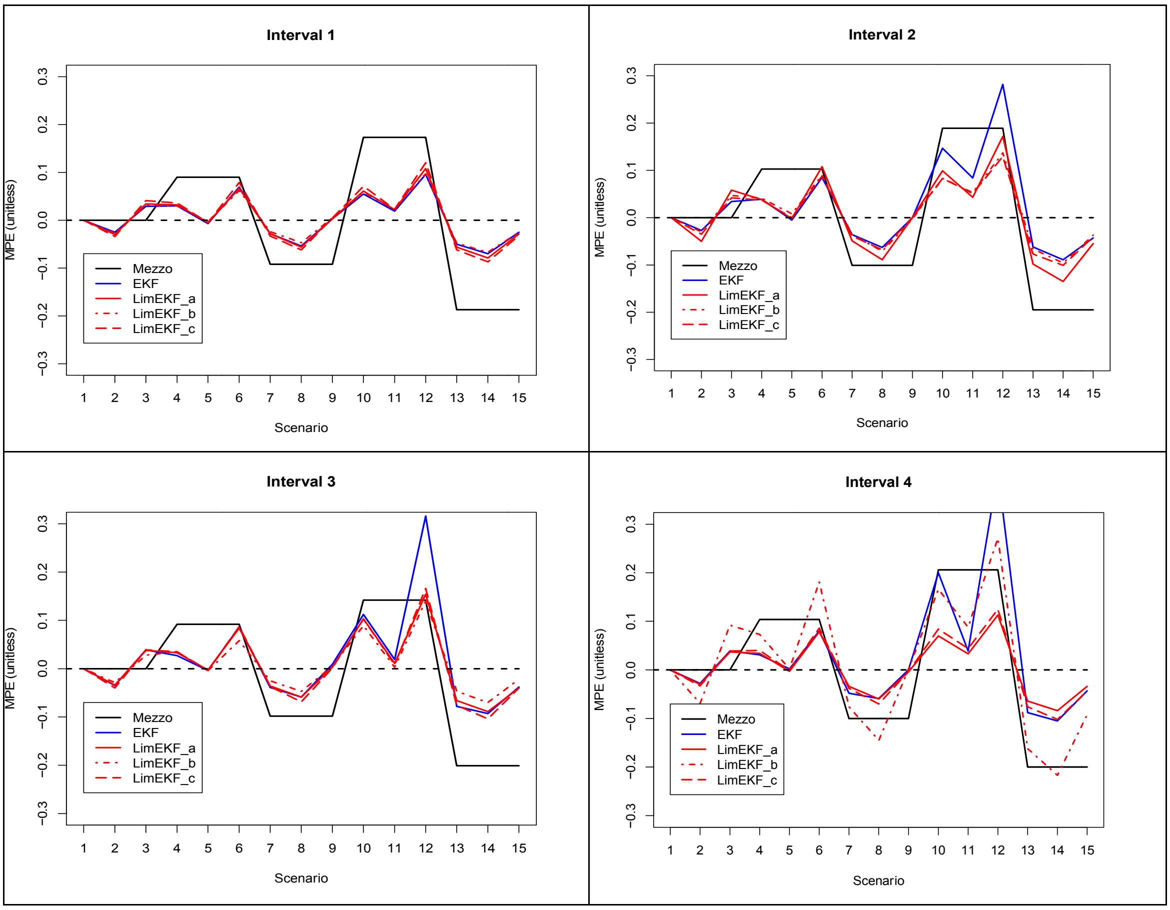

Figure 2. Overview of the calibration results of the synthetic network based on the MPE statistic

Case Study 2: Real-World, Large-Scale Network

Having established that the Limiting EKF can provide favourably results (often comparable to the EKF) in a small, synthetic network, the next step of this research was to evaluate the scalability of the experiments and to assess the transferability and applicability of the Limiting EKF algorithm to a real-world, large-scale network. For this task, the already validated network of the central district (Södermalm area; for a description of the network and the calibration process c.f. Burghout et al. [24]) of Stockholm city was used. This area is one of the most highly populated areas in Scandinavia. The Södermalm network consists of 462 OD pairs, 1101 links, and 577 nodes, and includes the urban arterial network of the area as well as the Södra Länken motorway, located at the south end of Stockholm. The total number of possible routes in the network is 4,491. Both Södra Länken motorway and its connection to the west (E4 motorway), experience congestion during the morning and afternoon peak hours, since these are the major commuter routes to and from the central district. A schematic of the Södermalm network as modelled within Mezzo is presented on Figure 3.

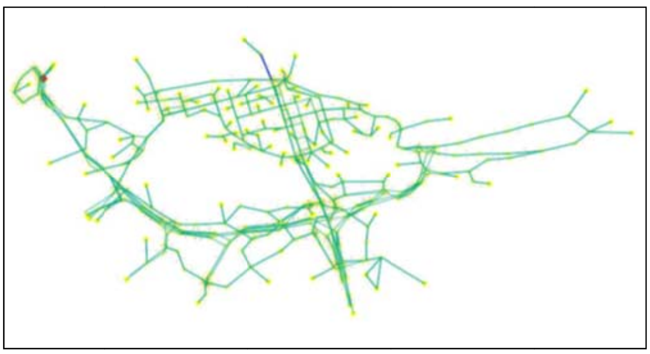

Figure 3. Södermalm network modelled in Mezzo

This scenario was also simulated for eight 15-min intervals (i.e. a total duration of two hours), considering the first hour as the warm-up period. In this set of experiments two levels of the OD demand were evaluated (set at -20% and -10% of the original demand), and perfect knowledge of traffic counts was available across a small sample of the network links. For determining which traffic count measurements to use in the limiting EKF algorithm, a simple sampling technique of selecting one every ten (non-zero) links was undertaken. This resulted to a total of 85 measurements. Similar to the synthetic network, it was further assumed that the covariance matrices $\boldsymbol{Q_h}$ and $\mathbf{R_h}$ are identity matrices. In addition, a 5% value was assumed for the linear delta factor, indicating the percentage of the demand to be considered during the linearization step.

As an intermediate step, the EKF algorithm was run for each scenario and the computed $\boldsymbol{H}$ matrices for the four intervals were used to calculate the average $\boldsymbol{H}$ matrix to be used in the Kalman gain matrix $\boldsymbol{G_h}$ of the Limiting EKF algorithm. The algorithm was evaluated in terms of its ability to estimate the traffic flow conditions at the measurement locations, through the RMSN and MPE statistics (as in case study 1). The results of the evaluation of the two scenarios are presented in Table 2.

Table 2. Summary results of the Limiting EKL algorithm (Södermalm network)

		Mezzo		LimEKF		Improvement	
		RMSN	**MPE**	**RMSN**	**MPE**	**RMSN**	**MPE**
S. 1 (-20%)	**Int. 1**	**24.1%**	**-13.6%**	**12.7%**	**-4.4%**	**47.3%**	**67.6%**
	Int. 2	**27.7%**	**-19.8%**	**14.3%**	**-6.2%**	**48.4%**	**68.7%**
	Int. 3	**29.4%**	**-17.8%**	**13.5%**	**-3.6%**	**54.1%**	**79.8%**
	Int. 4	**26.4%**	**-13.9%**	**16.0%**	**-3.4%**	**39.4%**	**75.5%**
S. 2 (-10%)	**Int. 1**	**13.1%**	**-7.1%**	**11.8%**	**1.0%**	**9.9%**	**85.9%**
	Int. 2	**15.0%**	**-9.7%**	**13.5%**	**-8.5%**	**10.0%**	**12.4%**
	Int. 3	**16.7%**	**-6.6%**	**15.7%**	**-5.5%**	**6.0%**	**16.7%**
	Int. 4	**16.2%**	**-8.1%**	**15.4%**	**-5.7%**	**4.9%**	**29.6%**

Based on the results shown in Table 2, the Limiting EKF algorithm provides significantly lower (i.e. better; between 40% and 80%) RMSN and MPE statistics compared to the base scenario, when the OD demand is -20%. For smaller variation of the OD demand the algorithm still appears to perform as expected, although it is not as efficient in minimizing the difference between the simulated and observed conditions. Still, the improvement is between 5% and 30%, with a single value exceeding 85% improvement). In conjunction with the results of the synthetic network, it can be inferred that the LimEKF algorithm performs well with regards to correcting the erroneous values of the OD demands, thus providing more accurate estimation of the prevailing conditions, especially when the discrepancies are significant. Scatterplots of observed vs. simulated traffic counts at the available network links using the results from the Limiting EKF for OD estimation for the two scenarios discussed above, are presented in Figure 4. Besides a slight underestimation of larger traffic counts, the results are fairly consistent with the "45 degree" straight line that represents the ideal results. It is noted that such an underestimation is expected, as the starting demand is significantly lower than the "true" demand.

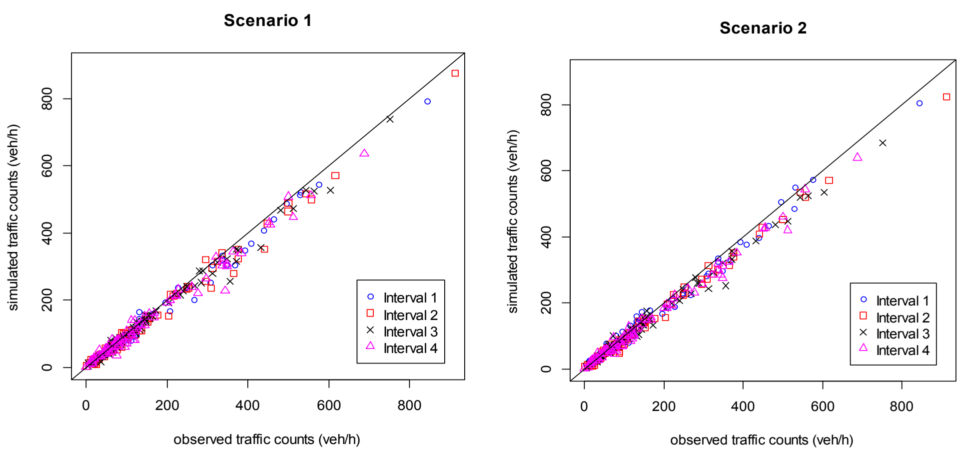

Figure 4. Scatterplots of observed and simulated traffic counts at the Södermalm network for the Limiting EKF algorithm

Conclusions

The results of the EKF algorithm (Algorithm 1) using a synthetic network showed that the algorithm provides acceptable corrections and minimizes the discrepancy between the simulated and observed traffic conditions. To overcome the computational burden of the linearization step during the on-line calibration, a variation of the EKF algorithm, the Limiting EKF algorithm, was also evaluated. The Limiting EKF algorithm computes the Kalman gain matrix off-line, taking into account pre-computed gain matrices. Experiments in the synthetic network showed that, the Limiting EKF algorithm provides accuracy comparable to that of the best algorithm (EKF), while providing substantial improvement in computational efficiency. As it was shown in Figures 1 and 2, in most cases these results were consistent among the three different approaches for computing the Kalman gain matrix. The increased robustness of the proposed algorithm can be explained by the fact that the average of various pre-computed Kalman gain matrices reduces the noise that is introduced in the computation of the individual Kalman gain matrices. Additional experiments undertaken in an existing large-scale network further validate these findings. The efficiency of the Limiting EKF algorithm increases with increased variation of the OD demands, whereas it's estimating and predictive power diminishes for low OD demand variations.

Future (ongoing) research includes trying additional combinations of Kalman gain matrices for the computation of the Limiting Kalman gain matrix, as well as testing the generalization and scalability of the obtained results; for example, whether the results would hold when using other traffic simulators and when applying the approach on larger, more complex networks. In addition, ongoing research performed using the Södermalm network evaluates the robustness of the LimEKF algorithm and the properties of the Kalman gain matrix, considering various structures of the error covariance matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$, as well as the impact of the number of available traffic counts in the network.

Acknowledgments

This work was supported in part by NTUA Basic Research Grant 65/1840 and benefited from the participation of the first two authors in the EU COST Action TU0903 – MULTITUDE. The authors would like to thank Dr. Wilco Burghout for help and support relating to the interfacing of the mezzo simulator within the developed framework.

References

1. Balakrishna, R., Ben-Akiva, M. and H. N. Koutsopoulos (2007). Off-line Calibration of Dynamic Traffic Assignment: Simultaneous Demand and Supply Estimation. *Transportation Research Record*, No. 2003, 50–58.
2. Antoniou, C., Balakrishna, R., Koutsopoulos, H. N. and M. Ben-Akiva (2011). Calibration methods for simulation-based dynamic traffic assignment systems. *International Journal of Modeling and Simulation,* 31(3), 227–233.
3. Tavana, H. and H. Mahmassani (2000). Estimation and application of dynamic speed-density relations by using transfer function models. *Transportation Research Record*, No. 1710, 47–57.
4. Huynh, N., Mahmassani, H. and H. Tavana (2002). Adaptive speed estimation using transfer function models for real-time dynamic traffic assignment operation. In Proceedings of the 81st Annual Meeting of the Transportation Research Board, January 13–17, 2002 (cd-rom). Washington, D.C.: Transportation Research Board.
5. Qin, X. and H. Mahmassani (2004). Adaptive calibration of dynamic speed-density relations for online network traffic estimation and prediction applications. In Proceedings of the 83rd Annual Meeting of the Transportation Research Board, January 11–15, 2004 (cd-rom). Washington, D.C.: Transportation Research Board.
6. Antoniou, C., Ben-Akiva, M. and H. N. Koutsopoulos (2005). On–line calibration of traffic prediction models. *Transportation Research Record: Journal of the Transportation Research Board*, No. 1934, 235–245.
7. Wang, Y. and M. Papageorgiou (2005). Real–time freeway traffic state estimation based on Extended Kalman Filter: A general approach. *Transportation Research Part B*, 39, 141–167.
8. Wang, Y., Papageorgiou, M. and A. Messmer (May 2007). Real–time freeway traffic state estimation based on Extended Kalman Filter: A case study. *Transportation Science*, 41, 167–181.

9. Wang, Y., Papageorgiou, M., Messmer, A., Coppola, P., Tzimitsi, A. and A. Nuzzolo (2009). An adaptive freeway traffic state estimator. *Automatica*, 45(1), 10–24.
10. Ashok, K. and M. Ben-Akiva (1993). Dynamic O-D matrix estimation and prediction for real-time traffic management systems. In C. Daganzo (Ed.), *Transportation and Traffic Theory*, (pp. 465–484). Amsterdam, The Netherlands: Elsevier Science Publishing.
11. Ashok, K. and M. Ben-Akiva (2000). Alternative approaches for real-time estimation and prediction of time-dependent origin-destination flows. *Transportation Science*, 34(1), 21–36.
12. Antoniou, C. (1997). *Demand simulation for dynamic traffic assignment*. Master's thesis, Massachusetts Institute of Technology.
13. Ben-Akiva, M., Bierlaire, M., Koutsopoulos, H. N. and R. Mishalani (2002). Gendreau, M. and Marcotte, P. (Eds), *Transportation and network analysis: current trends, chapter Real-time simulation of traffic demand-supply interactions within DynaMIT*, (pp. 19–36). Dordrecht, the Netherlands: Kluwer Academic Publishers. Miscellenea in honor of Michael Florian.
14. Ben-Akiva, M., Koutsopoulos, H.N., Antoniou, C. and R. Balakrishna (2010). Traffic Simulation with DynaMIT. In J. Barcelo (Ed.), *Fundamentals of traffic simulation*, (pp. 363–398). New York: Springer.
15. Bierlaire, M. and F. Crittin (2004). An efficient algorithm for real–time estimation and prediction of dynamic OD tables. *Operations Research*, 52(1), 116–127.
16. Zhou, X. and H.S. Mahmassani (2004). A structural state space model for real–time origin– destination demand estimation and prediction in a day–to–day updating framework. In Proceedings of the 83rd Annual Meeting of the Transportation Research Board, January 11–15, 2004 (cd-rom). Washington, D.C.: Transportation Research Board.
17. Antoniou, C. (September 2004) *On–line Calibration for Dynamic Traffic Assignment*. PhD dissertation, Massachusetts Institute of Technology, Cambridge.
18. Antoniou, C., Ben-Akiva, M., and H. N. Koutsopoulos (2007). Non–linear Kalman filtering algorithms for on–line calibration of dynamic traffic assignment models. *IEEE Transactions on Intelligent Transportation Systems*, 8(4), 661–670.
19. Ashok, K. (1996) *Estimation and Prediction of time-dependent Origin-Destination Flows*. PhD dissertation, Massachusetts Institute of Technology.
20. Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering (ASME)*, 82D, 35–45.
21. Chui, C. K. and G. Chen (1999). *Kalman Filtering with Real-Time Applications*. Heidelberg: Springer-Verlag.
22. Burghout, W. (2004) *Hybrid Mesoscopic-Microscopic Traffic Simulation*. PhD Thesis, Royal Institute of Technology (KTH).
23. Burghout, W., Koutsopoulos, H., Andreasson, I. (2005). Hybrid mesoscopic-microscopic traffic simulation. *Transportation Research Record*, No.1934, 218–225.
24. Burghout, W, Koutsopoulos, H. N. and I. Andreasson, (2010). Incident Management and Traffic Information Tools and Methods for Simulation-Based Traffic Prediction. *Transportation Research Record*, No. 2161, 20–28.

Appendix A

Table A.1. Summary results of the EKF and Limiting EKF algorithm (1/2)

		Mezzo		EKF		LimEKF (a)		LimEKF (b)		LimEKF (c)	
		RMSN	MPE	RMSN	MPE	RMSN	MPE	RMSN	MPE	RMSN	MPE
S. 1	Int. 1	0. 0%	0. 0%	0%	0%	0. 0%	0. 0%	0. 0%	0. 0%	0. 0%	0. 0%
	Int. 2	0. 0%	0. 0%	0%	0%	0. 0%	0. 0%	0. 0%	0. 0%	0. 0%	0. 0%
	Int. 3	0. 0%	0. 0%	0%	0%	0. 0%	0. 0%	0. 0%	0. 0%	0. 0%	0. 0%
	Int. 4	0. 0%	0. 0%	0%	0%	0. 0%	0. 0%	0. 0%	0. 0%	0. 0%	0. 0%
S. 2	Int. 1	0. 0%	0. 0%	2. 8%	-2. 5%	3. 1%	-3. 0%	2. 9%	-2. 7%	3. 6%	-3. 4%
	Int. 2	0. 0%	0. 0%	3. 4%	-2. 7%	5. 4%	-5. 0%	3. 9%	-3. 0%	4. 0%	-3. 5%
	Int. 3	0. 0%	0. 0%	3. 6%	-3. 5%	3. 5%	-3. 4%	3. 2%	-2. 9%	4. 0%	-4. 0%
	Int. 4	0. 0%	0. 0%	3. 0%	-2. 8%	3. 4%	-3. 0%	6. 9%	-6. 9%	3. 3%	-3. 3%
S. 3	Int. 1	0. 0%	0. 0%	2. 9%	2. 9%	3. 5%	3. 4%	3. 1%	3. 1%	4. 1%	4. 1%
	Int. 2	0. 0%	0. 0%	3. 4%	3. 4%	5. 8%	5. 8%	4. 9%	4. 7%	4. 2%	4. 2%
	Int. 3	0. 0%	0. 0%	4. 1%	3. 9%	3. 9%	3. 8%	3. 1%	2. 8%	4. 3%	3. 9%
	Int. 4	0. 0%	0. 0%	3. 9%	3. 9%	3. 7%	3. 7%	9. 3%	9. 2%	3. 9%	3. 9%
S. 4	Int. 1	9. 0%	9. 0%	3. 1%	3. 0%	3. 2%	3. 2%	3. 0%	3. 0%	3. 6%	3. 6%
	Int. 2	10. 5%	10. 3%	4. 0%	3. 9%	4. 0%	3. 8%	4. 2%	4. 1%	3. 9%	3. 7%
	Int. 3	9. 5%	9. 2%	2. 7%	2. 7%	3. 6%	3. 3%	3. 5%	3. 5%	3. 5%	3. 5%
	Int. 4	10. 4%	10. 4%	3. 4%	3. 1%	3. 7%	3. 4%	7. 5%	7. 3%	4. 2%	4. 0%
S. 5	Int. 1	9. 0%	9. 0%	1. 6%	-0. 7%	1. 4%	-0. 5%	1. 3%	-0. 3%	1. 7%	-0. 5%
	Int. 2	10. 5%	10. 3%	2. 6%	-0. 5%	2. 8%	-0. 2%	2. 1%	0. 8%	2. 3%	0. 0%
	Int. 3	9. 5%	9. 2%	1. 7%	-0. 4%	1. 0%	-0. 3%	2. 2%	-0. 4%	1. 7%	-0. 4%
	Int. 4	10. 4%	10. 4%	0. 2%	0. 2%	0. 3%	-0. 3%	0. 7%	0. 2%	0. 4%	-0. 1%
S. 6	Int. 1	9. 0%	9. 0%	6. 9%	6. 9%	6. 7%	6. 7%	6. 2%	6. 2%	7. 9%	7. 9%
	Int. 2	10. 5%	10. 3%	8. 6%	8. 5%	10. 8%	10. 8%	9. 0%	9. 0%	8. 9%	8. 8%
	Int. 3	9. 5%	9. 2%	8. 6%	8. 6%	8. 4%	8. 4%	6. 2%	5. 8%	8. 7%	8. 7%
	Int. 4	10. 4%	10. 4%	8. 0%	8. 0%	7. 9%	7. 8%	18. 3%	18. 1%	8. 6%	8. 6%
S. 7	Int. 1	9. 2%	-9. 2%	2. 8%	-2. 8%	2. 8%	-2. 8%	2. 3%	-2. 4%	3. 3%	-3. 3%
	Int. 2	10. 1%	-10. 1%	3. 6%	-3. 6%	4. 9%	-4. 9%	3. 6%	-3. 6%	3. 9%	-3. 9%
	Int. 3	10. 5%	-9. 8%	4. 1%	-3. 9%	4. 0%	-3. 6%	3. 2%	-2. 5%	4. 1%	-3. 7%
	Int. 4	10. 2%	-10. 0%	4. 9%	-4. 8%	3. 6%	-3. 4%	7. 7%	-7. 6%	4. 1%	-3. 8%

Table A.2. Summary results of the EKF and Limiting EKF algorithm (2/2)

		Mezzo		EKF		LimEKF (a)		LimEKF (b)		LimEKF (c)	
		RMSN	MPE	RMSN	MPE	RMSN	MPE	RMSN	MPE	RMSN	MPE
S. 8	Int. 1	9. 2%	-9. 2%	5. 5%	-5. 4%	5. 6%	-5. 6%	4. 7%	-4. 7%	6. 3%	-6. 2%
	Int. 2	10. 1%	-10. 1%	6. 6%	-6. 3%	9. 1%	-8. 9%	7. 5%	-7. 1%	7. 0%	-6. 8%
	Int. 3	10. 5%	-9. 8%	6. 0%	-5. 9%	5. 9%	-5. 9%	4. 8%	-4. 7%	6. 9%	-6. 9%
	Int. 4	10. 2%	-10. 0%	5. 9%	-5. 9%	6. 0%	-6. 0%	14. 6%	-14. 5%	7. 0%	-7. 0%
S. 9	Int. 1	9. 2%	-9. 2%	0. 7%	0. 3%	0. 7%	0. 3%	0. 7%	0. 5%	0. 68%	0. 46%
	Int. 2	10. 1%	-10. 1%	0. 5%	0. 0%	0. 8%	0. 0%	0. 6%	-0. 3%	0. 52%	0. 02%
	Int. 3	10. 5%	-9. 8%	3. 8%	0. 9%	3. 4%	0. 3%	3. 9%	0. 5%	3. 58%	0. 18%
	Int. 4	10. 2%	-10. 0%	1. 9%	0. 0%	1. 9%	-0. 4%	1. 4%	0. 1%	1. 93%	-0. 36%
S. 10	Int. 1	17.4%	17.3%	5.6%	5.5%	6.2%	6.1%	5.9%	5.8%	7.1%	7.1%
	Int. 2	19.1%	18.9%	15.0%	14.7%	10.0%	9.9%	8.4%	8.3%	8.4%	8.3%
	Int. 3	14.5%	14.2%	11.3%	11.2%	10.4%	10.3%	9.2%	8.9%	10.4%	10.3%
	Int. 4	20.6%	20.6%	20.1%	20.0%	7.1%	7.0%	16.6%	16.6%	8.4%	8.4%
S. 11	Int. 1	17.4%	17.3%	2.7%	1.9%	2.9%	2.2%	3.1%	2.3%	2.9%	2.3%
	Int. 2	19.1%	18.9%	9.1%	8.4%	5.3%	4.3%	6.0%	5.3%	6.0%	5.1%
	Int. 3	14.5%	14.2%	3.1%	1.9%	2.9%	1.1%	0.7%	0.3%	2.8%	0.8%
	Int. 4	20.6%	20.6%	4.4%	3.9%	3.3%	3.3%	9.3%	8.8%	4.7%	4.4%
S. 12	Int. 1	17.4%	17.3%	9.5%	9.6%	10.7%	10.8%	9.7%	9.7%	11.9%	12.0%
	Int. 2	19.1%	18.9%	28.2%	28.2%	17.2%	17.2%	13.7%	13.7%	12.9%	12.9%
	Int. 3	14.5%	14.2%	31.9%	31.6%	15.5%	15.4%	13.9%	14.0%	16.7%	16.6%
	Int. 4	20.6%	20.6%	40.8%	40.8%	11.3%	11.3%	27.1%	27.1%	12.5%	12.5%
S. 13	Int. 1	18.7%	-18.7%	5.0%	-5.0%	5.6%	-5.6%	4.8%	-4.8%	6.2%	-6.2%
	Int. 2	19.5%	-19.5%	6.2%	-6.2%	9.8%	-9.8%	6.7%	-6.6%	7.6%	-7.7%
	Int. 3	20.4%	-20.1%	8.5%	-7.8%	7.4%	-6.6%	6.0%	-4.7%	8.3%	-7.6%
	Int. 4	20.2%	-20.0%	8.9%	-8.8%	6.6%	-6.4%	16.2%	-16.2%	7.8%	-7.6%
S. 14	Int. 1	18.7%	-18.7%	7.0%	-7.0%	8.0%	-7.9%	6.7%	-6.7%	8.7%	-8.7%
	Int. 2	19.5%	-19.5%	9.1%	-8.9%	13.6%	-13.5%	9.8%	-9.6%	10.2%	-10.1%
	Int. 3	20.4%	-20.1%	9.4%	-9.3%	9.1%	-8.9%	7.3%	-7.0%	10.5%	-10.4%
	Int. 4	20.2%	-20.0%	10.5%	-10.5%	8.5%	-8.4%	21.8%	-21.7%	10.3%	-10.2%
S. 15	Int. 1	18.7%	-18.7%	2.5%	-2.5%	2.9%	-2.9%	2.5%	-2.5%	3.2%	-3.2%
	Int. 2	19.5%	-19.5%	4.4%	-4.3%	5.6%	-5.5%	3.6%	-3.7%	4.0%	-4.0%
	Int. 3	20.4%	-20.1%	5.6%	-3.8%	5.6%	-4.0%	3.8%	-2.2%	5.4%	-4.0%
	Int. 4	20.2%	-20.0%	5.1%	-4.3%	3.6%	-3.4%	9.3%	-9.2%	4.8%	-4.3%