ABSTRACT

This paper presents a method for estimating origin-destination tables and route choice from Bluetooth detector data which explicitly accounts for missed detections. The paper also explores the performance of the method using simulation experiments. The method works by writing the expected value of observed detector sequence counts as a function of the route flows to be estimated, and then by minimizing the difference between this function and the actually observed counts. For route flows of 100 vehicles or less, the method performs poorly with respect to relative estimation error. But for route flows greater than 300 vehicles the method shows little bias; has planning-level precision or better; and performs better than a naive method which ignores missed detections. In addition, the method is relatively robust to uncertainty and variation across normal ranges of detection probability and penetration rate.

INTRODUCTION

Origin-Destination (OD) tables are a critical part of many transportation analyses, but OD patterns are difficult to observe directly. Several methods have been proposed for estimating accurate OD tables, but each has its drawbacks. License plate matching techniques are costly because they are labor intensive; travel diaries and surveys are also costly, and don't directly measure actual OD patterns. OD table estimation from link counts requires introducing information by making important assumptions that may not reflect actual travel patterns.

Recently Bluetooth detectors have emerged as viable option for cheaply collecting travel pattern data suitable for OD estimation. These detectors are deployed at road-side locations throughout an area of interest and detect the globally-unique identification numbers of Bluetooth-enabled devices that pass near, thus enabling the tracking of vehicles as they travel between detectors (*1*). With this data, estimating OD tables seems feasible, with the level of detail limited only by the number and locations of detectors. In fact, Bluetooth data has already been used to estimate OD tables for several important studies in Florida, including studies to understand the demand for future managed lanes (see, for example, (*2*)).

However, estimating OD tables from Bluetooth detectors is not without challenges. Often vehicles are missed even though they travel within range of the detector (*3*); and typically only a minority of all vehicles is detectable at all. In addition, the probability of a missed detection and the penetration rate of detectable vehicles are most often not known with great precision. So when approaching the problem of estimating OD tables from Bluetooth data several questions arise. How can Bluetooth data be used to produce reliable estimates? Can Bluetooth data be used to estimate route choice? How important is the effect of missed detections? Is it small enough to ignore? How does the penetration rate of detectable vehicles affect the reliability of estimates? Does uncertainty about detection probability and penetration rate prevent accurate estimations?

Some of these questions have been addressed in previous research, but only partially. Kwon and Varaiya proposed a method to estimate OD tables from electronic toll tag data that explicitly addressed missed detections, but their method only applies to networks where each OD pair has only one route. In addition, their investigation of the method's performance was limited to a single test case and so could not answer the questions regarding importance of detection probability, penetration rate and uncertainty (*4*). More recent research proposed a hidden Markov model for reconstructing a vehicle trajectory from Bluetooth data considering missed detections (*5*). However, it was not clear how or if the method could be extended to estimate OD tables and the evaluation was limited to a single test case. Other research has not explicitly addressed the possibility of missed detections (*6, 7*).

This paper seeks to answer the research questions by proposing an estimation method that explicitly accounts for missed detections and by exploring the performance of the method across a comprehensive set of test scenarios. The paper also compares the method to a naive method that ignores missed detections, and illustrates an application of the method with an example.

ESTIMATION METHOD

Consider a highway system with Bluetooth detectors deployed near highway segments. Model the system as a network with links and nodes, and model each detector by inserting a node that splits the corresponding highway segment at the point nearest the detector. Vehicle trips on this network begin at origin nodes and end at destination nodes. Every OD pair has at least one and possibly several practical routes connecting them, and every route has some non-negative path flow. Each route is defined by a sequence of nodes, and may include one or more detector nodes. Extracting the detector nodes from a route (while preserving their order) produces the detector sequence (DS) associated with the route.

Each vehicle in the network is either equipped with an active Bluetooth-enabled device, and is detectable; or is not equipped and so is not detectable. As a detectable vehicle travels along a route it has an opportunity to be detected at each detector in the route's DS. However, at each detector there is some probability that the detection fails even though the detectable vehicle passes near. If the detection is successful, then the vehicle's unique ID is recorded, along with the time stamp of the detection and the name of the detector. If the detection fails, then no data is recorded.

At the end of the data collection period, the data from all detectors is combined, then grouped by vehicle ID and sorted by time stamp. Once the data are grouped and sorted, the vehicle IDs and time stamps are of no further use. They are dropped them from the data set, leaving just sequences of detector names - one for each vehicle trip. Each of these sequences only includes those detectors where the detection was successful. If a detection was not successful, it appears nowhere in the dataset. If none of the detections for a particular vehicle trip were successful, then that vehicle trip is not represented in the dataset at all. Non-detectable vehicles never appear in the dataset.

Since each sequence in the data set represents successful detections, each is called a successful detection sequence (SDS). The entire data set is described concisely and without loss of information by tabulating the number of times each unique SDS occurred.

The goal is to use the dataset (SDS frequencies) to estimate traffic patterns on the highway network. In the ideal case one could estimate the path flow for each individual route on the network. However, note that several routes could have identical DSs, and routes that have the same DS are not distinguishable from each other in the data. Therefore only the sum of paths flows for each set of routes sharing the same DS can be estimated. Call each of these sums a detector sequence flow (DS flow), and the goal becomes to estimate each DS flow. With DS flows in hand, an OD table can be computed by aggregating flows according to first and last detector in the DS.

To define the problem more formally, order the DSs arbitrarily so as to collect the (to be estimated) DS flows into a vector θ. Also let k index the set of detectors, let p_k denote the probability of a successful detection at detector k, and let P denote the vector of all probabilities. Finally, denote the portion of vehicles that are detectable with w, and assume that w and P are known. Then the problem to be solved is:

Estimate θ given known P and w, and a data set consisting of SDS counts.

To make the estimation, begin by collecting the SDS counts into a vector Y. However, note that the SDS dataset may not include every possible SDS, simply because of the random nature of the SDS counts and the fact that some SDSs may be extremely unlikely. The SDS counts dataset needs to be augmented to include the SDSs that were observed with zero count. The complete set of SDSs can be determined by enumerating all the combinations of length at least one from every DS, where every combination is without replacement and preserves ordering. Then those SDSs from the complete set that did not appear in the original data set are entered with a frequency of zero, and this augmented dataset is then organized into the vector Y.

The counts in Y are random variables that are related to the parameters θ. In particular, if q_{ij} denotes the probability that an instance of the j-th DS produces the i-th SDS, then the expected value of the i th SDS count is

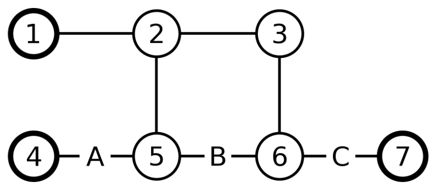

FIGURE 1 A simple network illustrates the concepts of detector sequence (DS) and successful detection sequence (SDS). The lettered nodes A, B and C are detector nodes. Nodes 1 and 4 are origin nodes, and node 7 is a destination node, so that there is a total of two OD pairs: (1, 7) and (4, 7). Nodes 1 and 7 have two practical routes connecting them: (1, 2, 3, 6, C, 7) and (1, 2, 5, B, 6, C, 7); and nodes 4 and 7 have one practical route connecting them: (4, A, 5, B, 6, C, 7), making a total of three different routes. Each of the three routes has a distinct DS, for a total of three DSs: (C), (B, C), and (A, B, C). The DS (C) has only one possible SDS, which is (C). The DS (B, C) has three possible SDSs: (B), (C), and (B, C). The DS (A, B, C) has seven possible SDSs: (A), (B), (C), (A, B), (A, C), (B, C) and (A, B, C). Note that some SDSs are shared by multiple DSs: the SDS (C) is shared by all three DSs; and the SDSs (B) and (B, C) are shared by two DSs.

$$E[y_i] = w \sum_{j=1,\ldots,|\theta|} q_{ij}\theta_j$$

where $|\theta|$ is the number of DSs. The probability q_{ij} is zero if the j-th DS cannot produce the i-th SDS. Otherwise,

$$q_{ij} = \prod_{k \in H_i} p_k \prod_{\{k \in G_j | k \notin H_i\}} (1 - p_k)$$

where G_j is the set of detectors in the j-th DS, and H_i is the set of detectors in the i-th SDS. Combining the expected values into a single vector, the expected value becomes

$$E[Y] = A\theta$$

where the element in the i-th row and j-th column of A is

$$a_{ij} = wq_{ij}.$$

Let the data set Y serve as an approximation for $E[Y]$. Then the estimate $\hat{\theta}$ is the set of values that minimizes the difference between Y and $A\hat{\theta}$. To measure the difference use the function

$$f(\theta) = \sum_{i=1,\ldots,|Y|} \left| \frac{(A\theta)_i - y_i}{y_i + 0.01} \right|$$

where $|Y|$ is the number of SDSs. Then the estimate for $\hat{\theta}$ is

$$\arg\min f(\theta)$$

subject to $\theta \geq 0$. The objective function is minimized with sequential least squares programming with a non- negativity constraint.

ASSUMPTIONS AND SIMPLIFICATIONS

The method makes some assumptions and simplifications that deserve attention. These are addressed here.

The method assumes that each vehicle carries a maximum of one detectable Bluetooth-enabled device. In reality, some vehicles may carry two or more devices, which could result in multiple SDSs for the same vehicle trip. This assumption is easily relaxed by redefining w as the average number of detectable Bluetooth-enabled devices per vehicle (typically less than one), and treating each of the multiple SDSs as a separate trip. This redefinition has no effect on the solution method. However, for clarity the method will continue to assume a maximum of one device per vehicle.

The previous section assumed that grouping by vehicle ID would result in sequences that are each associated with one vehicle trip. In reality, if the data collection period is long enough, the data will have instances where the sequence for one vehicle includes detections from multiple trips. The method assumes that before continuing with the analysis these multi-trip sequences are split into separate sequences - one for each trip. This is usually easily done by looking for time gaps that are larger than some threshold between two adjacent detections.

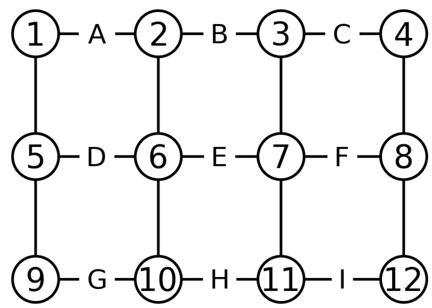

FIGURE 2 The test network.

The previous section also implied that a successful detection would result in a single data point. In reality there may be several duplicate detections all within a few seconds of each other. However, the method assumes that these duplicate detections would be combined into one representative detection before proceeding with the analysis.

The method also assumes that the data collection period is long enough relative to trip durations that the effect of trips that occur only partially during the data collection period is negligible.

The final assumption is that estimates for w and P are available. Estimating these values is a separate problem that is not addressed here.

PERFORMANCE

The performance of the method was tested by simulating Bluetooth data collection deployments using the network in Figure 2. This network has 21 nodes, labeled 1 through 12 and A through I. Every numbered node pair except for self-pairs are valid OD pairs, so that there are a total of 132 OD pairs. Every lettered node is detected by a Bluetooth detector. Every link in the network has a travel time (weight) of either one or two units, as indicated by the link lengths in the figure. Traffic only flows on least-time paths, and a single OD pair may have multiple least-time paths, making a total of 312 unique routes. These routes together produce 79 unique DSs.

Performance across different levels of traffic volume, detection probability, and penetration rate

The method was tested across different conditions. The tests used 12 different test cases. Each case had a different combination of penetration rates, detection probabilities, and traffic volume levels. Six cases used penetration rate of 0.05, and the other six cases used a penetration rate of 0.20; the two rates are labeled "low" and "high". In a similar manner, six of the cases used a set of nine low detection probabilities, and the remaining six cases used high detection probabilities. The low detection probabilities were chosen to be between .55 and .75; the high probabilities are equal to the low probabilities plus 0.20. The traffic volumes had three levels with four cases each: low, medium and high. The low volumes were chosen to be a set of 79 integers between 30 and 100. The medium and high volumes are equal to the low volumes multiplied by 10 and 100, respectively. The ranges for penetration rate, detection probabilities, and traffic volumes were chosen to cover a typical range of values.

Each of the 12 test cases was simulated 100 times. With each simulation, each trip was designated as detectable or non-detectable based on the results of a Bernoulli trial with probability of success equal to the penetration rate. For each detectable trip the SDS was generated by simulating a Bernoulli trial for each detector in the DS in turn, using the detection probabilities of the detectors. The SDS frequencies were then tabulated for each simulation. Frequencies for SDSs with length zero were dropped from the tabulation.

For each of the 79 DS flows from each of the 1200 simulations, the error was calculated by dividing the estimated value by the true value. A perfectly accurate estimation would have an error value of one. Errors less than one indicate that the estimation was too low; errors greater than one indicate that the estimation was too high.

The results of the simulations and estimations are shown in Figures 3 through 5. Each figure presents results from all 1200 simulations, but each disaggregates the results along a different dimension. The results are investigated by examining how each of the three dimensions (penetration rate, detection probability, and traffic volume) affects the performance of the method.

Of the three dimensions traffic volume seems to have the most dramatic effect. At low traffic volumes the method performs very poorly, but at high traffic volumes the model performs at least reasonably well. This result is not surprising. The proposed method depends on treating the dataset of SDS frequencies as an approximation of the expected values. At very low volumes, this approximation is not a very good one; at very high volumes, it often is. Compared to the effect of traffic volume, the effects of detection probability and penetration rate are small. However, the results behave as expected, with an increase probabilities and penetration rates causing better performance. Ignoring the low volume effect, the method performs reasonably well overall, with errors being centered on one and most errors falling between 0.5 and 1.5. Greater precision is limited mainly by the random nature of the underlying process, and the relatively small penetration rates of detectable vehicles.

Effect of Uncertainty on Performance

To test the effect of uncertainty on performance, a second set of estimates was made based on the same dataset from the 12 test cases. This second estimate introduced uncertainty by using estimates for the penetration rate and detection probabilities. For each simulation the estimates for penetration rate and detection probability were artificially generated by sampling from a uniform distribution constructed by setting the minimum and maximum equal to 0.9 and 1.1 times the true value, and then truncating to the range between 0.001 and 0.999 if necessary. This second estimate allows investigating the effect of uncertainty on the performance of the model.

The results are presented in Figure 6. As expected, the errors are closer to one when the penetration rate and probabilities are known than when they are estimated. However, the effect is small compared to the effect of traffic volume level.

Comparison to Naive Method

One of the research questions was whether missed detections can be ignored, or in other words whether the proposed method performs better than a naive method.

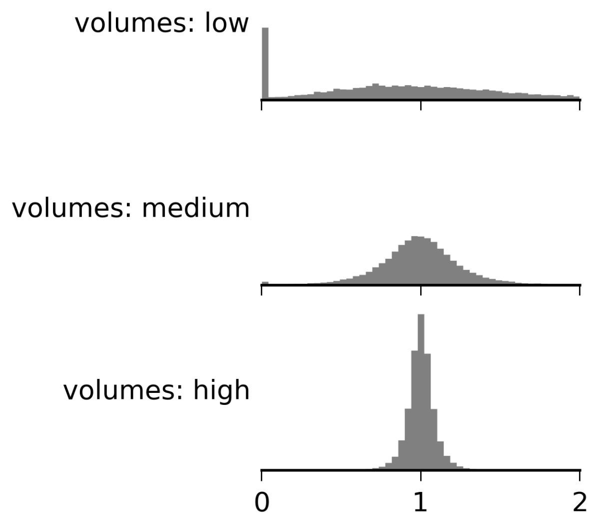

FIGURE 3 Performance results by volume level.

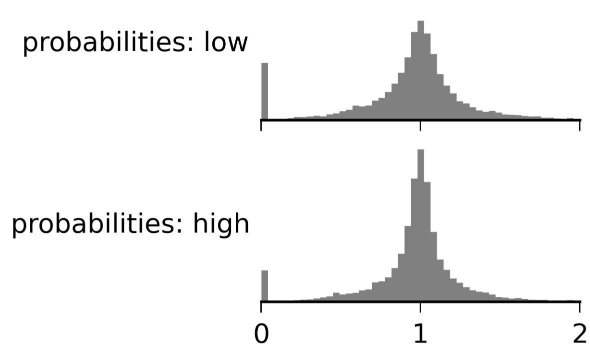

FIGURE 4 Performance results by detection probability level.

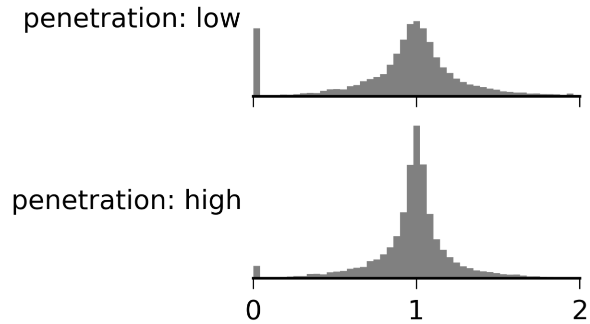

FIGURE 5 Performance results by penetration rate level.

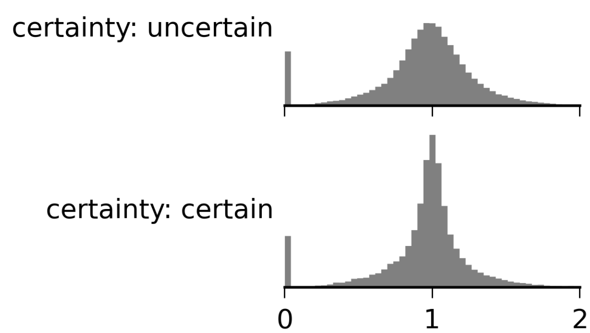

FIGURE 6 Performance results by certainty level.

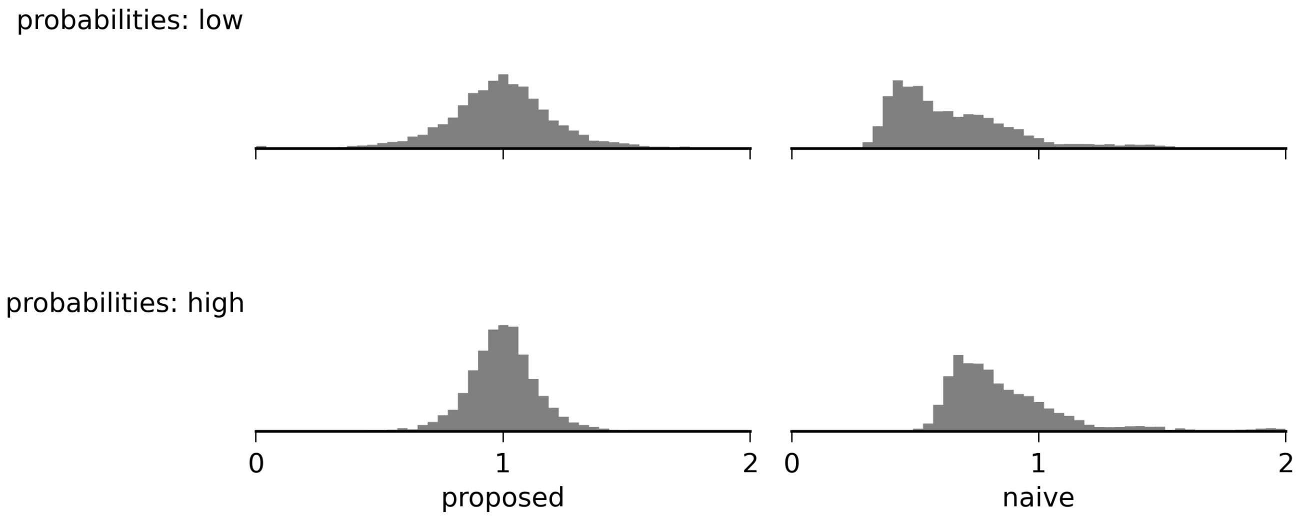

FIGURE 7 Performance comparison to naive method.

To answer the question, OD flows were estimated using the SDS datasets from the simulations and a naive method. This method aggregates the SDS counts according to first and last detector in each SDS, and then scales the counts by a single factor to represent the entire population of vehicles. Many methods for determining an appropriate factor can be invented. For the purposes of the research, the naive method was allowed to have the advantage of knowing the true total number of trips. Then the factor is the multiplier that will cause the estimated OD flows to have the same number of total trips. While this approach would not be possible in an actual application (since the true total number of trips is typically not known), it serves the research purposes here.

The estimates are compared to the estimates from the proposed method. Since the naive method is not capable of estimating DS flows, but only OD flows, the estimates from the proposed method are aggregated based on first and last detector in the DS, the comparison is made between OD flow estimates.

The comparisons are limited to the test cases with medium traffic volumes and a high penetration rate; and separate comparisons are made for cases with high and low probabilities. The results are presented in Figure 7. The estimates from the proposed method are generally centered on one while the naive estimates show bias, even though they have the advantage of knowing the true total number of trips, showing that missed detections cannot be ignored and the proposed method performs better than the naive method.

Example Application of Proposed Method

As a final test of the performance of the proposed method it is applied to estimate DS flows for an artificial deployment on the network in Figure 2, with DS flows chosen randomly from the uniform distribution between 300 and 10,000 and with a penetration rate of 0.10; and with detection probabilities set to equal the "low" detection probabilities plus 0.1. The deployment was simulated and the DS flows were estimated from the SDS count dataset using the proposed method. In addition, 95 percent bootstrap confidence intervals on the estimates were estimated by simulating the deployment 400 times while treating the estimate of DS flows as the true parameter value.

The resulting point estimates and intervals are presented in Figure 8. Some of the intervals are large - bigger than plus or minus roughly 50 percent of the estimated value - but most cover the true value. The estimation computations, including the point estimates and bootstrap simulations, took about five minutes on a desktop computer, demonstrating that the method is computationally feasible.

Summary of Performance Testing and Implications

The performance testing revealed several characteristics of the model that are important to consider when applying the method.

- First, the method is computationally feasible.
- Second, estimates using a naive method are biased and are worse than estimates from the proposed method. Thus an analyst should not be tempted to shortcut the estimation by ignoring missed detections.
- Third, the performance of the model improves with higher traffic volumes, so the analyst should try to increase the volumes by increasing the duration of the data collection period, or by increasing the number of repetitions of the data collection period. For example, if the period of interest is a three hour period in the afternoon on a weekday, data could be collected on each of 10 weekdays, rather than just one weekday. Then the DS flow estimates could be divided by 10 to estimate DS flows on one "average" weekday.
- Finally, most estimation errors are between 0.5 and 1.5. To the analyst this level of precision may be described as "planning" level. Thus the estimates from the proposed method should not be used for design or operational analysis, but may be used to aid analysis for high-level or long-term investment decisions.

CONCLUSION

In this paper a method for estimating OD flows and route choice using Bluetooth data was proposed. The performance of the model is limited by the random nature of missed detections and the relatively low penetration rate of detectable vehicles. However, the method performs better than a simpler naive method, and it estimates with planning-level precision or better. Thus the method is recommended for higher-level planning analyses.

One promising improvement to the method is to include information available from link counts. Practical applications have included link counts as marginal counts in a follow-on iterative proportional fitting process. However, integrating the link counts into a single unified model may offer significant improvements in performance.

Other potential improvements are related to the assumptions about the detection probability and penetration rate. The method assumed that each detector has a single probability of a successful detection, and that that probability is constant across all routes and vehicle trips. In reality,

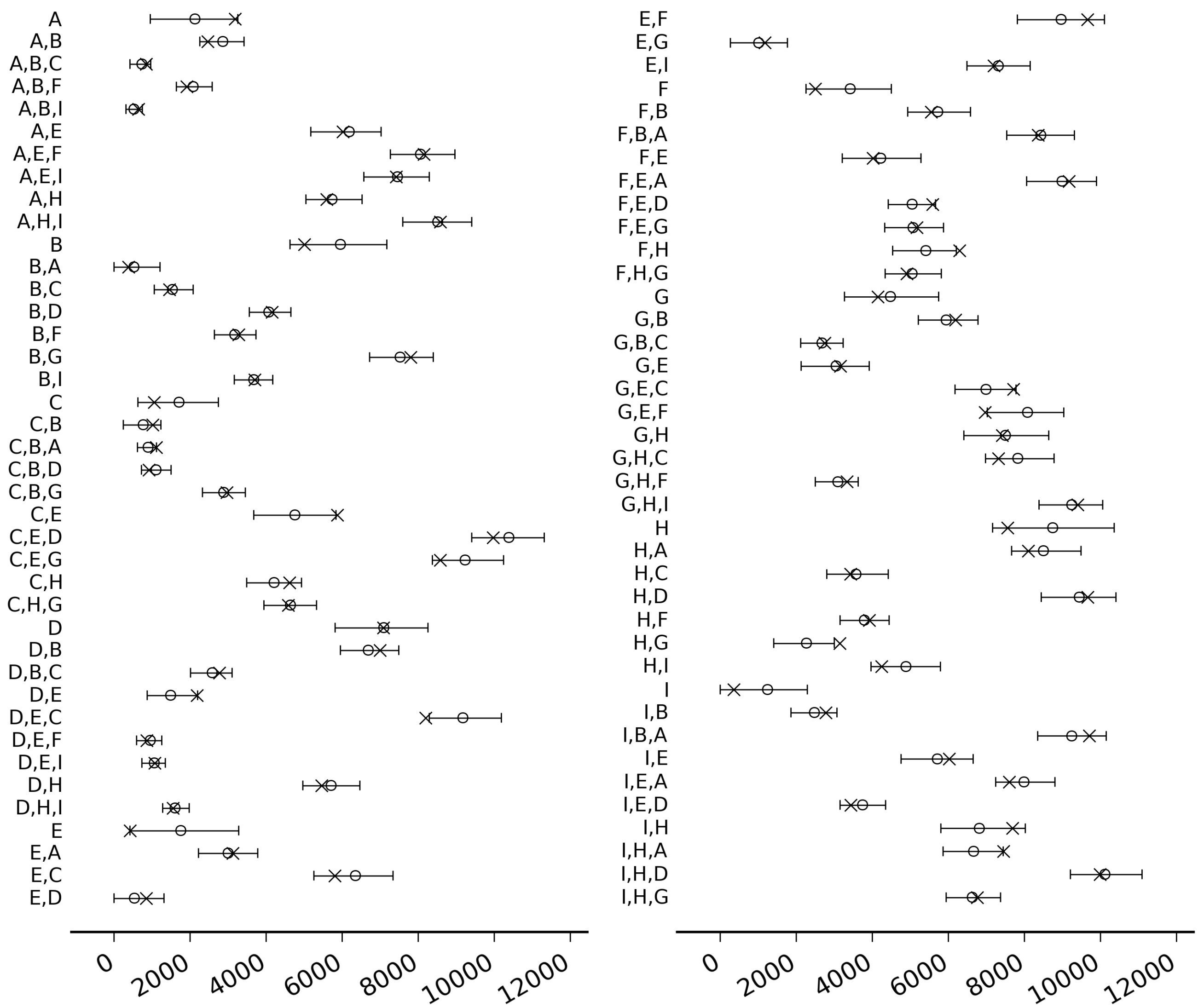

FIGURE 8 Point estimates (circles) and confidence intervals from an example application. The true DS flows are shown with "x".

probability of detection probably varies by type of vehicle, traffic volume, traffic speed, distance between the vehicle and the detector, and other factors. The variation can be limited (and the probability increased) through careful deployment of the detectors. However, in real deployments the assumption of constant probability may not hold. Also assumed is that the proportion of vehicles that are detectable is constant across all routes. This may not be totally true. For example, the commercial vehicle proportion may vary across routes, and this factor may affect the likelihood of a vehicle having a Bluetooth-enabled device. Future research may improve upon the proposed method by allowing non-constant detection probability and penetration rate.

ACKNOWLEDGEMENTS

The author would like to thank the Florida Department of Transportation and Florida's Turnpike Enterprise for their valuable support of this work.

REFERENCES

[1] Carpenter, C., M. Fowler, and T. J. Adler, Generating Route-Specific Origin-Destination Tables Using Bluetooth Technology. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2308, 2012, pp. 96–102.

[2] RSG, *I-75 & Palmetto Expressway Origin-Destination Study: Final Report*. RSG, White River Junction, VT, 2012.

[3] Brennan Jr, T. M., J. M. Ernst, C. M. Day, D. M. Bullock, J. V. Krogmeier, and M. Martchouk, Influence of vertical sensor placement on data collection efficiency from bluetooth MAC address collection devices. *Journal of Transportation Engineering*, Vol. 136, No. 12, 2010, pp. 1104–1109.

[4] Kwon, J. and P. Varaiya, Real-Time Estimation of Origin-Destination Matrices with Partial Trajectories from Electronic Toll Collection Tag Data. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 1923, 2005, pp. 119–126.

[5] Lees-Miller, J., E. Wilson, and S. Box, Hidden Markov Models for Vehicle Tracking with Bluetooth. In *TRB Annual Meeting Compendium of Papers, Transportation Research Board Annual Meeting*, 2013.

[6] Hainen, A. M., J. S. Wasson, S. M. L. Hubbard, S. M. Remias, G. D. Farnsworth, and D. M. Bullock, Estimating Route Choice and Travel Time Reliability with Field Observations of Bluetooth Probe Vehicles. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2256, 2011, pp. 43–50.

[7] Barceló, J., L. Montero, L. Marquès, and C. Carmona, Travel time forecasting and dynamic origin-destination estimation for freeways based on bluetooth traffic monitoring. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2175, 2010, pp. 19–27.