

# Origin-Destination Estimation using Sparsity

Q.Li, S. Chatterjee, G. Floetteroed, J. Gross

**Abstract**—For origin-destination (OD) estimation problem, we show promise of using sparse signal processing algorithms in this report. We evaluated performance on a simulated dataset for Stockholm city.

**Index Terms**—Sparse representations, mobility pattern, OD estimation.

## I. INTRODUCTION

The origin/destination (OD) matrix estimation problem is to estimate an OD matrix describing flows from all source nodes to all sink nodes in a network. The data available for this consists of noisy observations of arc flows on that network. Variations of several standard statistical methods have been tried out on the OD matrix estimation problem. Examples of existing methods are based on entropy maximization and information minimization [1], Bayesian estimation [2], generalized least squares [3], [4], [5], and maximum likelihood estimation [6]. A time dimension into the problem was first introduced in [7].

The OD estimation problem is, however, far from being solved. The key difficulty in the OD matrix estimation problem is its under-determinedness, which is usually constructed away by the inclusion of a “prior OD matrix” that is taken for instance from an outdated survey and included like a supplementary set of measurements in the estimation problem. The arbitrariness of this approach, however, is widely acknowledged.

*A key observation from a signal processing point of view is that the involved OD estimation problem is inherently sparse. This means OD estimation variables in a vector form contain a small number of high value elements; other way we can say most elements are either very small or zero.* A single individual faces literally a myriad of possible mobility plans (being essentially a sequence of route/mode/time-annotated trips that connect activity locations) to implement at any point in time [8]. However, only a single one is actually chosen. *On an aggregate level, the superposition of these individual OD mobility plans results in mobility patterns (which represent, at an aggregate scale, how individuals intend to travel over some time span up to an entire day) where sparsity is preserved.* Sparse signal processing [9], such as compressed sensing [10], [11] has recently received significant attention as key breakthroughs have been achieved. While existing sparse signal processing algorithms and theoretical results may not be directly applicable to an OD estimation problem in a setup of urban mobility sensing due to the enormous size and complex (non-standard) structure of the involved mathematical system, we provide an initial result in this technical report on a toy size model.

## II. SYSTEM SETUP FOR OD ESTIMATION

In OD matrix estimation problem, the system setup is

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w} \in \mathbb{R}^m, \quad (1)$$

where  $\mathbf{y}$  is a measurement vector representing arc flows,  $\mathbf{A}$  is the system matrix usually called assignment matrix,  $\mathbf{x} \in \mathbb{R}^n$  being the OD matrix (in vector form), and  $\mathbf{w}$  is an error (noise) term that represents unexplained deviations between model predictions and observations, including measurement errors. The observations  $\mathbf{y}$  comprise counting processes (i.e. number of people or vehicles during some time at some place), while the unknowns  $\mathbf{x}$  represent the occurrence of aggregate mobility patterns in the population. Mathematically speaking, we are interested in instantaneously estimation from  $\mathbf{y}$ , i.e. from a limited set of observations of individuals and vehicles, about the current mobility patterns  $\mathbf{x}$  and from this predicting the spatio-temporal evolution of the city-wide presence of individuals and vehicles. It is important to note that as we are only operating on aggregate counting data  $\mathbf{y}$ , the approach is privacy-preserving.

### A. Role of Sparsity

We already mentioned that  $\mathbf{x}$  is sparse. Typically the dimension of  $\mathbf{x}$  is also very large compared to the dimension of  $\mathbf{y}$  as the number of measurements of arcs are limited. In that case  $m \ll n$ , and the problem (1) is under-determined, having no conclusive (unique) mathematical solution. Recently the area of sparse signal processing, such as compressed sensing [10], [11] shown that it is possible to get good solution of the problem (1). The existing sparse signal processing approach will be tested in this report by recovering a synthetic OD matrix for the city of Stockholm from simulated link flows. The synthetic OD matrix is derived from an activity-based travel demand model, resulting in a realistic (sparsity) structure. We will use some existing computationally efficient algorithms from the area of sparse signal processing, such as subspace pursuit (SP) [12], orthogonal matching pursuit [13] for the experiments reported in the next section.

## III. RESEARCH METHODS

### A. Simulated Data for Stockholm

This study uses a system specification ( $\mathbf{A}$  matrix and  $\mathbf{w}$  noise) that are generated by a detailed simulation model system of the greater Stockholm region. The considered road network representation has approximately  $10^4$  links. The city is divided into approximately  $10^3$  zones. For each origin  $r$  and each destination  $s$ , the corresponding OD path is denoted by  $x_{rs}$ . A morning rush hour consisting of approximately  $10^5$  vehicle trips is simulated in the system, using an existing

origin/destination vector  $\mathbf{x} = [x_{rs}]$  and realistic assumptions about route choice and traffic flow propagation.

The concrete  $\mathbf{A}$  matrix used in this study is obtained by studying a snapshot of this simulation system at the peak of the simulated rush hour. The number  $n_l^{rs}$  of vehicles traveling from  $r$  to  $s$  and using link  $l$  is counted. The total flow on link  $l$ , denoted by  $y_l$ , then follows

$$y_l = \sum_{rs} n_l^{rs} \quad (2)$$

$$= \sum_{rs} \frac{n_l^{rs}}{x_{rs}} x_{rs}. \quad (3)$$

The system matrix is hence defined by

$$A = [a_l^{rs}] \quad (4)$$

$$a_l^{rs} = \frac{n_l^{rs}}{x_{rs}}. \quad (5)$$

This  $a_l^{rs}$  can be interpreted as the probability that a randomly selected vehicle traveling in OD relation  $rs$  can be observed on link  $l$ .

The idealized setting described so far does not include any error terms  $\mathbf{w} = [w_{rs}]$ , and their detailed characterization is indeed a delicate issue that depends on rather subtle assumptions made in the simulation model system. It is, however, generally true that all elements of  $\mathbf{y}$ ,  $\mathbf{A}$  and  $\mathbf{x}$  are non-negative. Further, the elements of  $\mathbf{y}$  have an upper bound that depends on the capacity of the respective link. These error distributions are difficult to characterize explicitly but simple to simulate: For a given  $\mathbf{x}$  and  $\mathbf{A}$  simulate some zero-mean  $\mathbf{w}$  and truncate the resulting  $\mathbf{Ax} + \mathbf{w}$  at zero and its upper bounds.

### B. Sparse Processing Algorithms

In this section, we briefly describe two sparse signal processing algorithms which are greedy in nature. Greedy algorithms are computationally simple and hence attractive for large systems. However, in signal processing field, it is typically used for a system with dense  $\mathbf{A}$  matrix. Here we will check their performance large and non-ideal system matrices.

First we summarize the main steps of the OMP in Algorithm 1 (see Algorithm 3 of [13]). Here  $K$  is the sparsity level of  $\mathbf{x}$ . Note that, in each iteration, the OMP performs a matched filter operation and an orthogonal projection. Using block-wise matrix inversion, the orthogonal projection operations can be performed recursively. Considering the worst case scenario that  $k = K$ , the necessary computation for each orthogonal projection is approximately  $7K^2 + 4KM$ . So, the total computation for each matched filter and each orthogonal projection is  $\mathcal{O}(MN + K^2 + KM)$ . Now, considering  $K$  iterations, the total complexity is  $\mathcal{O}(K(MN + K^2 + KM))$ .

Then we summarize the main steps of the SP algorithm in Algorithm 2 (see Algorithm 1 of [12]). The SP algorithm starts with an initial  $K$ -element support set  $\mathcal{I}_0$  and an initial residual  $\mathbf{r}_0 = \mathbf{y} - \mathbf{A}_{\mathcal{I}_0} \mathbf{A}_{\mathcal{I}_0}^\dagger \mathbf{y}$ . At the  $k$ 'th iteration stage, it forms the 'matched filter'  $\mathbf{A}^t \mathbf{r}_{k-1}$ , identifies the  $K$  highest amplitude coordinates, forms a dummy support set  $\mathcal{I}_{(u)} = \mathcal{I}_{k-1} \cup \mathcal{I}_{(p)}$ , refines out  $K$ -element support set  $\mathcal{I}_k$  from  $\mathcal{I}_{(u)}$ , solves a LS problem with the selected indices in  $\mathcal{I}_k$ , subtracts the LS fit and produces a new residual. Given the

---

#### Algorithm 1 : OMP for CS Recovery

---

Input:

1:  $\mathbf{A}$ ,  $\mathbf{y}$ ,  $K$ ;

Initialization:

1: Iteration counter  $k \leftarrow 0$ ;

2:  $\mathbf{r}_0 \leftarrow \mathbf{y}$ ,  $\mathcal{I}_0 \leftarrow \emptyset$ ;

Iterations:

1: **repeat**

2:  $k \leftarrow k + 1$ ;

3:  $i_k \leftarrow$  index of the highest amplitude of  $\mathbf{A}^t \mathbf{r}_{k-1}$ ;

4:  $\mathcal{I}_k \leftarrow \mathcal{I}_{k-1} \cup i_k$ ; (Note:  $|\mathcal{I}_k| = k$ )

5:  $\mathbf{r}_k \leftarrow \mathbf{y} - \mathbf{A}_{\mathcal{I}_k} \mathbf{A}_{\mathcal{I}_k}^\dagger \mathbf{y}$ ; (Orthogonal projection)

6: **until** ( $(\|\mathbf{r}_k\|_2 > \|\mathbf{r}_{k-1}\|_2)$  or  $(k > K)$ )

7:  $k \leftarrow k - 1$ ; (Previous iteration)

Output:

1:  $\hat{\mathbf{x}} \in \mathbb{R}^N$ , satisfying  $\hat{\mathbf{x}}_{\mathcal{I}_k} = \mathbf{A}_{\mathcal{I}_k}^\dagger \mathbf{y}$  and  $\hat{\mathbf{x}}_{\bar{\mathcal{I}}_k} = \mathbf{0}$ .

---



---

#### Algorithm 2 : SP for CS Recovery

---

Input:

1:  $\mathbf{A}$ ,  $\mathbf{y}$ ,  $K$ ;

Initialization:

1: Iteration counter  $k \leftarrow 0$ ;

2:  $\mathcal{I}_0 \leftarrow$  indices of the  $K$  highest amplitudes of  $\mathbf{A}^t \mathbf{y}$ ;

3:  $\mathbf{r}_0 \leftarrow \mathbf{y} - \mathbf{A}_{\mathcal{I}_0} \mathbf{A}_{\mathcal{I}_0}^\dagger \mathbf{y}$ ;

Iterations:

1: **repeat**

2:  $k \leftarrow k + 1$ ;

3:  $\mathcal{I}_{(p)} \leftarrow$  {indices of  $K$  highest amplitudes of  $\mathbf{A}^t \mathbf{r}_{k-1}$ };

4:  $\mathcal{I}_{(u)} \leftarrow \mathcal{I}_{k-1} \cup \mathcal{I}_{(p)}$ ; ( $K \leq |\mathcal{I}_{(u)}| \leq 2K$ )

5:  $\hat{\mathbf{x}}_{\mathcal{I}_{(u)}} \leftarrow \mathbf{A}_{\mathcal{I}_{(u)}}^\dagger \mathbf{y}$ ;  $\hat{\mathbf{x}}_{\bar{\mathcal{I}}_{(u)}} \leftarrow \mathbf{0}$ ; (Orthogonal projection)

6:  $\mathcal{I}_k \leftarrow$  {indices of the  $K$  highest amplitudes of  $\hat{\mathbf{x}}$ };

7:  $\mathbf{r}_k \leftarrow \mathbf{y} - \mathbf{A}_{\mathcal{I}_k} \mathbf{A}_{\mathcal{I}_k}^\dagger \mathbf{y}$ ; (Orthogonal projection)

8: **until** ( $\|\mathbf{r}_k\|_2 > \|\mathbf{r}_{k-1}\|_2$ )

9:  $k \leftarrow k - 1$ ; (Previous iteration)

Output:

1:  $\hat{\mathbf{x}} \in \mathbb{R}^N$ , satisfying  $\hat{\mathbf{x}}_{\mathcal{I}_k} = \mathbf{A}_{\mathcal{I}_k}^\dagger \mathbf{y}$  and  $\hat{\mathbf{x}}_{\bar{\mathcal{I}}_k} = \mathbf{0}$ .

---

sparsity level  $K$ , the algorithm estimates a support set of cardinality  $K$  in each iteration and runs until the residual norm minimization condition is violated. Note that, unlike in the case of serial atom selection based OMP algorithm, here the support set cardinality is not increased one-by-one through iterations. Rather, a  $K$ -element support set is refined through iterations by addition of potential new atoms and deletion of unnecessary atoms. An important point is to note how the  $K$ -element support set  $\mathcal{I}_k$  is chosen from the dummy support set  $\mathcal{I}_{(u)}$  through using the orthogonal projection that invokes LS solution. The dummy support set  $\mathcal{I}_{(u)}$  is formed through unionizing the previously estimated support set  $\mathcal{I}_{k-1}$  with the set of  $K$  new atoms' indices. Then the observation  $\mathbf{y}$  is orthogonally projected on the span of atoms that are indexed in  $\mathcal{I}_{(u)}$  followed by picking up  $K$  indices corresponding to the highest amplitude coefficients of the solution vector.

Normally, the SP algorithm converges less than  $K$  iterations. Assuming the worst case scenario that the SP algorithm runs at most  $K$  iterations, it performs  $K$  matched filtering and  $2K$  orthogonal projection operations. Note that, like OMP and OLS, the orthogonal projections can not be performed recursively. Hence, the total complexity for each iteration is  $\mathcal{O}(MN + K^2M)$ . Therefore, considering  $K$  iterations, the overall complexity of SP algorithm is  $\mathcal{O}(K(MN + K^2M))$ .

### C. Numerical Experiments

We used OMP and SP algorithms to evaluate the OD estimation performance. We first mention about the performance measure, called signal-to-reconstruction-noise ratio (SRNR), defined as

$$\text{SRNR} = \frac{\mathcal{E}\{\|\mathbf{x}\|_2^2\}}{\mathcal{E}\{\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2\}}$$

Here  $\hat{\mathbf{x}}$  is the estimated OD signal vector. The estimation is better when SRNR is higher.

Next we will discuss the experimental setups. The raw data could be divided into three parts: origins, destinations and the measurements, which is in the format as shown in Table I.

TABLE I  
RAW DATA

NUMBER	ORIGIN	DESTINATION	LINK
1	origin1	destination1	linka, linkb, ..., linkc
2	origin2	destination2	linkd, linke, ..., linkf
3	origin3	destination3	linkg, linkh, ..., linki
...	...	...	...
4227	origin4227	destination4227	linkj, linkk, ..., linkm

The measurements here correspond to the quantity of the vehicles that appear in some path and also appear in some link, which are identified with combination of numbers and letters. There are 974 different origins and 824 different destinations in the data. There is some overlap between the origins and destinations. Considering the overlaps, we found that we have 1158 different locations in total. There are 20201 different links. For this data, the dimensions of  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{A}$  are  $1340964 \times 1$ ,  $20201 \times 1$  and  $20201 \times 1340964$ , respectively.

We use fraction of sampling (or fraction of measurements) to show the estimation performance under different sampling levels. The fraction of sampling is defined as

$$\alpha = \frac{M'}{M},$$

where  $M' \leq M$ , and  $M = 20201$ . Note that  $\alpha \in (0, 1]$ . Further, Gaussian noise in different levels are added to the true measurement vector to evaluate the estimation performance in noisy environment (with the constraint of positiveness). The signal-to-measurement-noise-ratio (SMNR) is defined as

$$\text{SMNR} = \frac{\mathcal{E}\{\|\mathbf{x}\|_2^2\}}{\mathcal{E}\{\|\mathbf{w}\|_2^2\}}.$$

The OD estimation performance using OMP and SP algorithms are shown in Figure 1 and Figure 2. The x-axis shows the fraction of measurements. The y-axis shows the SRNR performance with noisy conditions with SMNR = 10, 20, 30

dB. We always used  $K$  as the half of  $M'$ . It is noted that performance deteriorates significantly with decrease in link measurements.

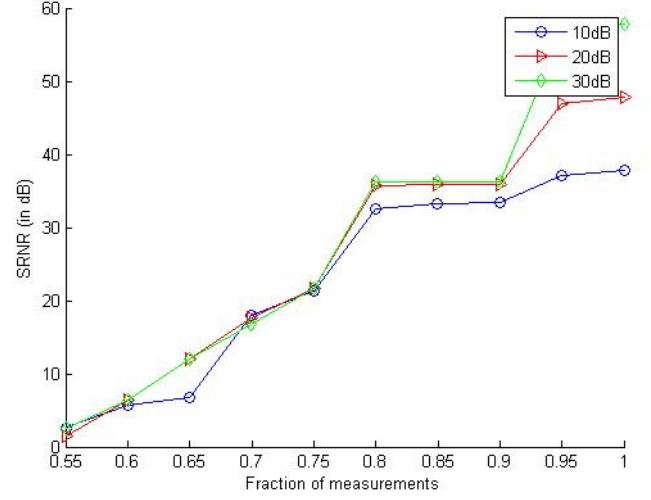


Fig. 1. OD estimation performance of OMP algorithm at varying SMNR.

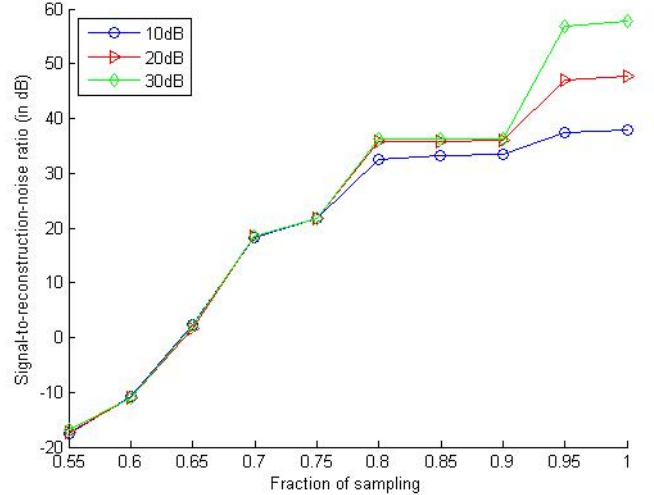


Fig. 2. OD estimation performance of SP algorithm at varying SMNR.

We observe that OMP and SP provide similar results. For execution of algorithms, the cpu running time of OMP is 10441 seconds, and the running time of SP is 477 seconds.

## IV. CONCLUSIONS

From our experimental study, we conclude that sparse processing has a high promise for OD estimation problem in a setup of urban mobility sensing. The future question is what happens for highly under-determined system where number of link measurements is very limited and the system size is very large in the sense that number of OD pairs is very large.

## REFERENCES

- [1] H. van Zuylen and L. G. Willumsen, "The most likely trip matrix estimated from traffic counts," *Transportation Research Part B*, vol. 14, no. 3, pp. 281–293, 1980.
- [2] M. Maher, "Inferences on trip matrices from observations on link volumes: a Bayesian statistical approach," *Transportation Research Part B*, vol. 17, no. 6, pp. 435–447, 1983.
- [3] M. Bell, "The estimation of origin-destination matrices by constrained generalised least squares," *Transportation Research Part B*, vol. 25, no. 1, pp. 13–22, 1991.
- [4] M. Bierlaire and P. Toint, "MEUSE: an origin-destination estimator that exploits structure," *Transportation Research Part B*, vol. 29, no. 1, pp. 47–60, 1995.
- [5] E. Cascetta, "Estimation of trip matrices from traffic counts and survey data: a generalised least squares estimator," *Transportation Research Part B*, vol. 18, no. 4/5, pp. 289–299, 1984.
- [6] H. Spiess, "A maximum likelihood model for estimating origin-destination models," *Transportation Research Part B*, vol. 21, no. 5, pp. 395–412, 1987.
- [7] E. Cascetta, D. Inaudi, and G. Marquis, "Dynamic estimators of origin-destination matrices using traffic counts," *Transportation Science*, vol. 27, no. 4, pp. 363–373, 1993.
- [8] J. Bowman and M. Ben-Akiva, "Activity based travel demand model systems," in *Equilibrium and advanced transportation modelling*, P. Marcotte and S. Nguyen, Eds. Kluwer, 1998, pp. 27–46.
- [9] M. Elad, *Sparse and redundant representations: From theory to applications in signal and image processing*. Springer, 2010.
- [10] D. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, april 2006.
- [11] E. Candes and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Proc. Magazine*, vol. 25, pp. 21–30, march 2008.
- [12] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *Information Theory, IEEE Transactions on*, vol. 55, no. 5, pp. 2230–2249, may 2009.
- [13] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *Information Theory, IEEE Transactions on*, vol. 53, no. 12, pp. 4655–4666, dec. 2007.