

PETRELS: SUBSPACE ESTIMATION AND TRACKING FROM PARTIAL OBSERVATIONS

Yuejie Chi¹, Yonina C. Eldar² and Robert Calderbank³

¹Dept. of Electrical Engineering, Princeton University, Princeton, NJ 08544, USA

²Dept. of Electrical Engineering, Technion - Israel Institute of Technology, Haifa 32000, Israel

³Dept. of Electrical Engineering, Duke University, Durham, NC 27708, USA

ABSTRACT

We consider the problem of reconstructing a data stream from a small subset of its entries, where the data stream is assumed to lie in a low-dimensional linear subspace, possibly corrupted by noise. It is also important to track the change of underlying subspace for many applications. This problem can be viewed as a sequential low-rank matrix completion problem in which the subspace is learned in an online fashion. The proposed algorithm, called Parallel Estimation and Tracking by REcursive Least Squares (PETRELS), identifies the underlying low-dimensional subspace via a recursive procedure for each row of the subspace matrix in parallel, and then reconstructs the missing entries via least-squares estimation if required. PETRELS outperforms previous approaches by discounting observations in order to capture long-term behavior of the data stream and be able to adapt to it. Numerical examples are provided for direction-of-arrival estimation and matrix completion, comparing PETRELS with state of the art batch algorithms.

Index Terms— subspace estimation and tracking, recursive least squares, matrix completion

1. INTRODUCTION

Many real world data can be viewed as an embedding of low-dimensional structure in a high-dimensional manifold. When the embedding is assumed linear, the underlying low-dimensional structure becomes a linear subspace. Subspace Identification and Tracking (SIT) plays an important role in various signal processing tasks such as online identification of network anomalies [1], moving target localization [2], beamforming [3], and denoising [4]. Conventional SIT algorithms collect full measurements of the data stream at each time, and subsequently update the subspace estimate by utilizing the track record of the stream history in different ways [5, 6]. However, in high-dimensional problems, it might be expensive and even impossible to collect data from all dimensions. For example in wireless sensor networks, collecting from all sensors continuously will quickly drain the battery power. Ideally we would prefer to only collect data from a fixed budget of sensors each time to increase the overall battery life, and still be able to identify the underlying structure.

Recent advances in Matrix Completion (MC) [7] theory, which enables reconstruction of a matrix from a few entries by assuming it is low-rank, have made it possible to infer data structure from incomplete observations. Identifying the underlying low-rank structure in MC is equivalent to subspace identification in a batch setting. When

the number of observed entries is slightly larger than the subspace dimension, it has been shown that with high probability, it is possible to test whether a highly incomplete vector of interest lies in a known subspace [8]. The GROUSE algorithm [9] was subsequently proposed for SIT from online partial observations using gradient descent rank-one updates of the estimated subspace on a Grassmannian manifold. However, due to the existence of “barriers” in the search path [10], GROUSE may be trapped at local minima. We demonstrate this behavior in the simulation section in the context of direction-of-arrival estimation.

In this paper we further study SIT given partial observations from a stream. In our proposed algorithm, called Parallel Estimation and Tracking by REcursive Least Squares (PETRELS), the underlying low-dimensional subspace is identified via a recursive procedure with discount factor for each row of the subspace matrix in parallel. The missing entries are then reconstructed via least-squares estimation if required. The discounting factor balances the algorithm’s ability to capture long term behavior and changes to that behavior to improve adaptivity. We also benefit from the fact that our optimization is not restricted to the Grassmannian which can be suboptimal and lead to issues of convergence to local minima. We provide numerical examples to measure the impact of the discount factor, and demonstrate the advantage of PETRELS over GROUSE for direction-of-arrival estimation. We also compare PETRELS with state of the art batch matrix completion algorithms.

The rest of the paper is organized as follows. Section 2 formulates the problem and Section 3 describes the algorithm in detail. Numerical results are provided in Section 4, after which we conclude in Section 5.

2. PROBLEM FORMULATION

At each time t , a vector $\mathbf{x}_t$ is generated as,

$$\mathbf{x}_t = \mathbf{U}_t \mathbf{a}_t + \mathbf{n}_t \in \mathbb{R}^M, \quad (1)$$

where the columns of $\mathbf{U}_t \in \mathbb{R}^{M \times r_t}$ span a low-dimensional subspace, the vector $\mathbf{a}_t \in \mathbb{R}^r$ specifying the linear combination of columns and $\mathbf{n}_t$ is additive noise. Assume only partial entries of the full vector $\mathbf{x}_t$ are observed, given by

$$\mathbf{y}_t = \mathbf{p}_t \odot \mathbf{x}_t = \mathbf{P}_t \mathbf{x}_t \in \mathbb{R}^M, \quad (2)$$

where $\mathbf{P}_t = \text{diag}[\mathbf{p}_t]$, $\mathbf{p}_t = [p_{1t}, p_{2t}, \dots, p_{Mt}]^T \in \{0, 1\}^M$, and $p_{mt} = 1$ if the m th entry is observed at time t . We are interested in identifying and tracking the changes in the subspace model, from streaming partial observations $(\mathbf{y}_t, \mathbf{P}_t)_{t=1}^\infty$.

To the end, we aim at minimizing the following *loss function* at each time n :

$$\mathbf{D}_n = \underset{\mathbf{D} \in \mathbb{R}^{M \times r}}{\text{argmin}} F_n(\mathbf{D}) = \underset{\mathbf{D} \in \mathbb{R}^{M \times r}}{\text{argmin}} \sum_{t=1}^n \lambda^{n-t} f_t(\mathbf{D}), \quad (3)$$

The work of Y. Chi and R. Calderbank was supported by ONR under Grant N00014-08-1-1110, by AFOSR under Grant FA 9550-09-1-0643, and by NSF under Grants NSF CCF -0915299 and NSF CCF-1017431. The work of Y. Eldar was supported in part by the Israel Science Foundation under Grant 170/10.

where the discount factor $0 \ll \lambda \leq 1$ discounts past observations, $\mathbf{D}$ is the estimated subspace of rank r , where r is assumed known and fixed throughout the algorithm (although it may not equal the true subspace dimension), and

$$f_t(\mathbf{D}) = \min_{\mathbf{a}} \|\mathbf{P}_t(\mathbf{x}_t - \mathbf{D}\mathbf{a})\|_2^2, \quad t = 1, \dots, n. \quad (4)$$

To motivate the loss function in (3) we note that if $\mathbf{U}_t = \mathbf{U}$ is not changing over time, then the RHS of (3) is minimized to zero when $\mathbf{D}_n$ spans the subspace defined by $\mathbf{U}$. If $\mathbf{U}_t$ is slowly changing, then λ is used to control the memory of the system and maintain tracking ability at time n . For example, by using $\lambda \rightarrow 1$ the algorithm gradually loses its ability to forget the past.

The GROUSE method can be viewed as optimizing (4) at each time n using stochastic gradient descent on the Grassmannian defined as $\{\mathbf{D} \in \mathbb{R}^{M \times r} : \mathbf{D}^T \mathbf{D} = \mathbf{I}_r\}$. The difference in the loss function (3) with respect to that of GROUSE is adding of the discount factor λ , and not restricting $\mathbf{D}$ to be unitary.

Fixing $\mathbf{D}$, (4) is minimized when

$$\mathbf{a}_t^*(\mathbf{D}) = (\mathbf{D}^T \mathbf{P}_t \mathbf{D})^\dagger \mathbf{D}^T \mathbf{P}_t \mathbf{x}_t.$$

Substituting $\mathbf{a}_t^*(\mathbf{D})$ in (4), we have

$$f_t(\mathbf{D}) = \mathbf{x}_t^T \left(\mathbf{P}_t - \mathbf{P}_t \mathbf{D} (\mathbf{D}^T \mathbf{P}_t \mathbf{D})^\dagger \mathbf{D}^T \mathbf{P}_t \right) \mathbf{x}_t. \quad (5)$$

Minimizing (5) over $\mathbf{D}$ is difficult. Instead, we propose PETRELS to approximately solve this optimization problem.

Before developing PETRELS we note that if there are further constraints on the coefficient $\mathbf{a}_t$, then a regularization term can be incorporated:

$$f_t(\mathbf{D}) = \min_{\mathbf{a} \in \mathbb{R}^r} \|\mathbf{P}_t(\mathbf{D}\mathbf{a} - \mathbf{x}_t)\|_2^2 + \beta_t \|\mathbf{a}\|_p,$$

where $p \geq 0$. For example, $p = 1$ enforces a sparse constraint on $\mathbf{a}_t$, and $p = 2$ enforces a norm constraint on $\mathbf{a}_t$. In this formulation the discount factor λ is fixed, and the influence of past estimates decreases geometrically; a more general online objective function can be given as

$$\mathbf{D}_n = \operatorname{argmin}_{\mathbf{D} \in \mathbb{R}^{M \times r}} F_n(\mathbf{D}) = \operatorname{argmin}_{\mathbf{D} \in \mathbb{R}^{M \times r}} \lambda_n F_{n-1}(\mathbf{D}) + f_n(\mathbf{D}).$$

However, for simplicity, we only consider equation (4) in this paper.

3. THE PETRELS ALGORITHM

3.1. Algorithm Details

The proposed PETRELS algorithm, as summarized by Algorithm ??, alternates between coefficient estimation and subspace update at each time n . In particular, the coefficient vector is estimated by solving

$$\mathbf{a}_n = \operatorname{argmin}_{\mathbf{a}} \|\mathbf{P}_n(\mathbf{x}_n - \mathbf{D}_{n-1}\mathbf{a})\|_2^2 \quad (6)$$

$$= (\mathbf{D}_{n-1}^T \mathbf{P}_n \mathbf{D}_{n-1})^\dagger \mathbf{D}_{n-1}^T \mathbf{P}_n \mathbf{x}_n, \quad (7)$$

where $\mathbf{D}_0$ is a random subspace initialization. The subspace $\mathbf{D}_n$ is then updated by minimizing

$$\mathbf{D}_n = \operatorname{argmin}_{\mathbf{D}} \sum_{t=1}^n \lambda^{n-t} \|\mathbf{P}_t(\mathbf{x}_t - \mathbf{D}\mathbf{a}_t)\|_2^2, \quad (8)$$

where $\mathbf{a}_t$, $t = 1, \dots, n$ are estimates from (6). The objective function in (8) can be decomposed for each row of $\mathbf{D}_n =$

$[\mathbf{d}_1^n, \mathbf{d}_2^n, \dots, \mathbf{d}_M^n]^T$ as

$$\mathbf{d}_m^n = \operatorname{argmin}_{\mathbf{d}_m} \sum_{t=1}^n \lambda^{n-t} p_{mt} (x_{mt} - \mathbf{a}_t^T \mathbf{d}_m)^2 \quad (9)$$

$$\begin{aligned} &= \left(\sum_{t=1}^n \lambda^{n-t} p_{mt} \mathbf{a}_t \mathbf{a}_t^T \right)^\dagger \left(\sum_{t=1}^n \lambda^{n-t} p_{mt} x_{mt} \mathbf{a}_t \right) \\ &= \mathbf{d}_m^{n-1} + p_{mn} (x_{mn} - \mathbf{a}_n^T \mathbf{d}_m^{n-1}) (\mathbf{R}_m^n)^\dagger \mathbf{a}_n, \end{aligned} \quad (10)$$

where $\mathbf{R}_m^n = \lambda \mathbf{R}_m^{n-1} + p_{mn} \mathbf{a}_n \mathbf{a}_n^T$. This results in a parallel procedure to update all rows of the subspace matrix $\mathbf{D}_n$. Finally, by the Recursive Least-Squares (RLS) updating formula, $(\mathbf{R}_m^n)^\dagger$ can be easily updated without matrix inversion using

$$\begin{aligned} (\mathbf{R}_m^n)^\dagger &= (\lambda \mathbf{R}_m^{n-1} + p_{mn} \mathbf{a}_n \mathbf{a}_n^T)^\dagger \\ &= \lambda^{-1} (\mathbf{R}_m^{n-1})^\dagger + p_{mn} \mathbf{G}_m^n; \end{aligned} \quad (11)$$

where $\mathbf{G}_m^n = \beta^{-1} \mathbf{v}_m^n (\mathbf{v}_m^n)^T$, with β_m^n and $\mathbf{v}_m^n$ given as

$$\beta_m^n = 1 + \lambda^{-1} \mathbf{a}_n^T (\mathbf{R}_m^{n-1})^\dagger \mathbf{a}_n, \quad (12)$$

$$\mathbf{v}_m^n = \lambda^{-1} (\mathbf{R}_m^{n-1})^\dagger \mathbf{a}_n. \quad (13)$$

To enable the RLS procedure, the matrix $(\mathbf{R}_m^0)^\dagger$ is initialized as a matrix with large entries on the diagonal, which we choose arbitrarily as the identity matrix $(\mathbf{R}_m^0)^\dagger = \delta \mathbf{I}_r$, $\delta > 0$ for all $m = 1, \dots, M$.

3.2. Convergence

In the full observation regime, PETRELS becomes equivalent to the well-known PAST [5] algorithm for subspace estimation, which is proved to converge to the global optima. However at this point the problem of deriving performance guarantees for PETRELS in the partial observation regime remains open.

Algorithm 1 PETRELS for Subspace Estimation

Input: a stream of vectors $\mathbf{y}_t$ and observed pattern $\mathbf{P}_t$.

Initialization: an $M \times r$ random matrix $\mathbf{D}_0$, and $(\mathbf{R}_m^0)^\dagger = \delta \mathbf{I}_r$, $\delta > 0$ for all $m = 1, \dots, M$.

```

1: for  $n = 1, 2, \dots$  do
2:    $\mathbf{a}_n = (\mathbf{D}_{n-1}^T \mathbf{P}_n \mathbf{D}_{n-1})^\dagger \mathbf{D}_{n-1}^T \mathbf{y}_n$ .
3:    $\mathbf{x}_n = \mathbf{D}_{n-1} \mathbf{a}_n$ .
4:   for  $m = 1, \dots, M$  do
5:      $\beta_m^n = 1 + \lambda^{-1} \mathbf{a}_n^T (\mathbf{R}_m^{n-1})^\dagger \mathbf{a}_n$ ,
6:      $\mathbf{v}_m^n = \lambda^{-1} (\mathbf{R}_m^{n-1})^\dagger \mathbf{a}_n$ ,
7:      $(\mathbf{R}_m^n)^\dagger = \lambda^{-1} (\mathbf{R}_m^{n-1})^\dagger + p_{mn} \beta_m^{-1} \mathbf{v}_m^n (\mathbf{v}_m^n)^T$ ,
8:      $\mathbf{d}_m^n = \mathbf{d}_m^{n-1} + p_{mn} (x_{mn} - \mathbf{a}_n^T \mathbf{d}_m^{n-1}) (\mathbf{R}_m^n)^\dagger \mathbf{a}_n$ .
9:   end for
10: end for
```

4. NUMERICAL RESULTS

We begin by examining the influence of the discount factor on the performance. Next we look at direction-of-arrival estimation and show that the proposed PETRELS algorithm demonstrates superior performance over GROUSE by identifying and tracking all the targets almost perfectly even in low SNR. Finally we compare our approach with matrix completion, and show that PETRELS is at least competitive with state of the art batch algorithms.

4.1. Choice of discounting factor

The choice of the discount factor λ plays an important role in how fast the algorithm converges. At each time t , a vector $\mathbf{x}_t$ of dimension $m = 500$ is generated as $\mathbf{x}_t = \mathbf{D}_{true}\mathbf{a}_t$, where $\mathbf{D}_{true}$ is an $(r = 10)$ -dimensional subspace generated with i.i.d. $\mathcal{N}(0, 1)$ entries, $\mathbf{a}_t$ is an $r \times 1$ vector with i.i.d. $\mathcal{N}(0, 1)$ entries. We assume that a fixed number of $K = 50$ entries in $\mathbf{x}_t$, a mere 10% percent of the full dimension, are revealed each time. This restriction is not necessary, but we make it here in order to guarantee a meaningful estimate of $\mathbf{a}_t$. Denoting the estimated subspace by $\hat{\mathbf{D}}$, we use the normalized subspace reconstruction error to examine the algorithm performance; this is calculated as $\|\mathcal{P}_{\hat{\mathbf{D}}_{\perp}}\mathbf{D}_{true}\|_F^2 / \|\mathbf{D}_{true}\|_F^2$, where $\mathcal{P}_{\hat{\mathbf{D}}_{\perp}}$ is the projector onto the orthogonal subspace $\hat{\mathbf{D}}_{\perp}$. We run the algorithm to time $n = 2000$ on the same data, and observe that the normalized subspace reconstruction error is minimized when λ is around 0.98; see Fig. 1. Hence, we keep $\lambda = 0.98$ hereafter.

Influences of other parameters including the initial estimated rank, number of measurements and noise level are also examined but not reported here due to limit of space.

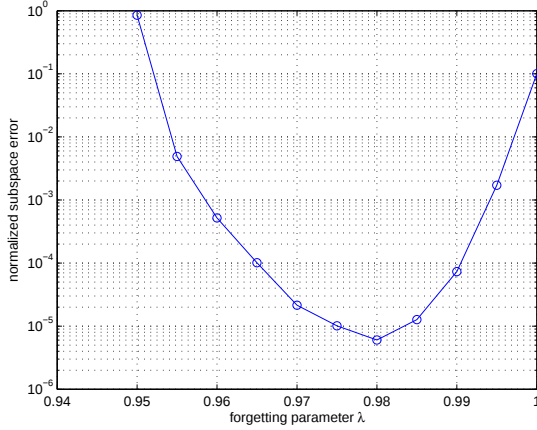


Fig. 1. The normalized subspace reconstruction error as a function of λ after running the algorithm until $n = 2000$ when 50 out of 500 entries of the signal are observed each time without noise.

4.2. Direction-Of-Arrival Analysis

We evaluate the resilience of PETRELS to direction-of-arrival analysis in array processing given GROUSE [9] as a baseline. Assume there are $n = 256$ sensors from a linear array, and the measurements from all sensors at time t are

$$\mathbf{x}_t = \mathbf{V}\Sigma\mathbf{a}_t + \mathbf{n}_t, \quad t = 1, 2, \dots \quad (14)$$

Here $\mathbf{V} \in \mathbb{C}^{n \times p}$ is a Vandermonde matrix defined by

$$\mathbf{V} = [\boldsymbol{\alpha}_1(\omega_1), \dots, \boldsymbol{\alpha}_p(\omega_p)], \quad (15)$$

where $\boldsymbol{\alpha}_i(\omega_i) = [1, e^{j2\pi\omega_i}, \dots, e^{j2\pi\omega_i(n-1)}]^T$, $0 \leq \omega_i < 1$, $\Sigma = \text{diag}\{\mathbf{d}\} = \text{diag}\{d_1, \dots, d_p\}$ is a diagonal matrix which characterizes the amplitudes of each mode, the coefficients $\mathbf{a}_t$ are generated with $\mathcal{N}(0, 1)$ entries, and the noise is generated with $\mathcal{N}(0, \epsilon^2)$ entries, where $\epsilon = 0.1$.

Each time we collect measurements from $K = 30$ random sensors. We are interested in identifying all $\{\omega_i\}_{i=1}^p$ and $\{d_i\}_{i=1}^p$. This

can be done by applying the well-known ESPRIT algorithm [11] to the estimated subspace $\hat{\mathbf{D}}$ of rank r , where r is specified a-priori corresponding to the number of modes to be estimated. Specifically, if $\mathbf{D}_1 = \hat{\mathbf{D}}(1 : n-1)$ and $\mathbf{D}_2 = \hat{\mathbf{D}}(2 : n)$ are the first and the last $n-1$ rows of $\hat{\mathbf{D}}$, then from the eigenvalues of the matrix $\mathbf{T} = \mathbf{D}_1^\dagger \mathbf{D}_2$, denoted by λ_i , $i = 1, \dots, r$, the set of $\{\omega_i\}_{i=1}^p$ can be recovered as

$$\omega_i = \frac{1}{2\pi} \arg \lambda_i, \quad i = 1, \dots, r. \quad (16)$$

The ESPRIT algorithm also plays a crucial role in recovery of the multipath delays from low-rate samples of the channel output from transmitting pulse streams with known shape [12].

Now consider 5 scatters at directions (frequencies) specified by $\omega = [0.1769, 0.1992, 0.2116, 0.6776, 0.7599]$, and amplitudes $d = [0.3, 0.8, 0.5, 1, 0.1]$. Note that there are three closely located modes and one weak mode, which makes the task challenging. We compare the performance of the proposed PETRELS algorithm and GROUSE. The rank specified in both algorithms is $r = 10$, which is the estimated number of modes; in our case it is twice the number of true modes. The estimated directions at each time for 10 modes are shown in Fig. 2. The color shown for each estimated mode points shows the amplitude corresponding to the color bar. The proposed PETRELS algorithm identifies and tracks all modes correctly, as shown by the 5 “lines” in Fig. 2 (a). It especially distinguishes the three closely-spaced modes perfectly, and the weak mode is identified later than the strong modes. The auxiliary modes are exhibited as “noise” in the scatter plot. With GROUSE the closely spaced nodes are erroneously estimated as one mode, the weak mode is missing, and spurious modes have been introduced.

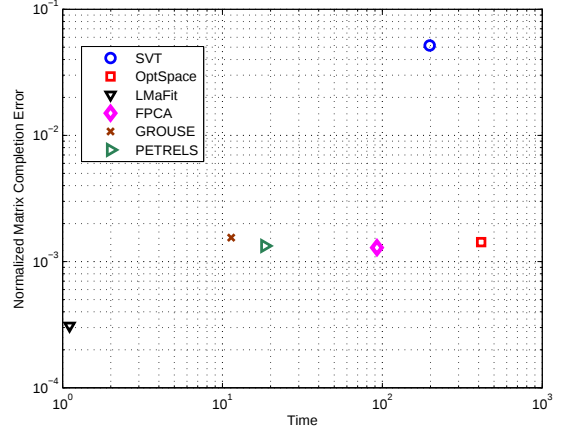


Fig. 3. Comparison with matrix completion batch algorithms shows that PETRELS is competitive in terms of computational time and accuracy.

4.3. Matrix Completion

Our algorithm can be viewed as an online version of MC, which has potential advantages for dealing with data size changes and small memory requirements. We compare performance of the proposed PETRELS algorithm for MC against batch algorithms LMaFit [13], FPCA [14], Singular Value Thresholding (SVT) [15], OptSpace [16] and the online algorithm GROUSE. The low-rank matrix is generated from a matrix factorization model with $\mathbf{X} = \mathbf{U}\mathbf{V}^T \in \mathbb{R}^{1000 \times 2000}$, where $\mathbf{U} \in \mathbb{R}^{1000 \times 10}$ and $\mathbf{V} \in \mathbb{R}^{2000 \times 10}$. The sampling rate is taken to be 0.05, so that

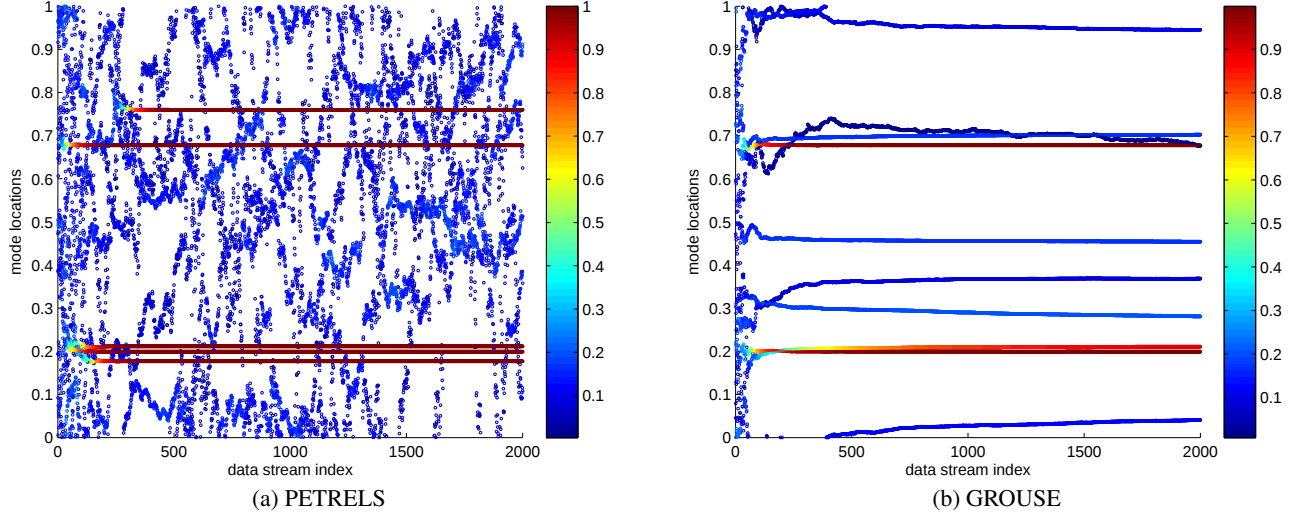


Fig. 2. Direction-Of-Arrival estimation using PETRELS and GROUSE algorithms: at each index, an estimation of 10 mode locations are made with color denoting mode amplitudes. All directions are identified and tracked successfully by PETRELS, but not by GROUSE.

only 5% of all entries are revealed. The running time is plotted against the normalized matrix reconstruction error, calculated as $\|\hat{\mathbf{X}} - \mathbf{X}\|_F / \|\mathbf{X}\|_F$; all entries in $\mathbf{U}$ and $\mathbf{V}$ are generated from the standard normal distribution $\mathcal{N}(0, 1)$. The proposed PETRELS algorithm matches the performance of batch algorithms, as shown in Fig. 3. Note that different algorithms have different input parameter requirements. For example, OptSpace needs to specify the tolerance to terminate the iterations, which directly decides the trade-off between accuracy and running time. Our simulation here only shows one particular realization and we simply conclude that PETRELS is competitive.

5. CONCLUSIONS

We considered the problem of reconstructing a data stream from a small subset of its entries, where the data stream is assumed to lie in a low-dimensional linear subspace, possibly corrupted by noise. This has significant implications for lessening the storage burden and reducing complexity, as well as tracking the changes for applications such as network monitoring and anomaly detection when the problem size is large. The PETRELS algorithm first identifies the underlying low-dimensional subspace via a discounted recursive procedure for each row of the subspace matrix in parallel, and then reconstructs the missing entries via least-squares estimation if required. The discount factor allows the algorithm capture long-term behavior as well as track the changes of the data stream. We demonstrate superior performance of PETRELS in direction-of-arrival estimation and showed that it is competitive with state of the art batch matrix completion algorithms.

6. REFERENCES

- [1] T. Ahmed, M. Coates, and A. Lakhina, "Multivariate online anomaly detection using kernel recursive least squares," *Proc. 26th IEEE International Conference on Computer Communications*, pp. 625–633, 2007.
- [2] S. Shahbazpanahi, S. Valaee, and M. H. Bastani, "Distributed source localization using esprit algorithm," *IEEE Transactions on Signal Processing*, vol. 49, no. 10, p. 21692178, 2001.
- [3] R. Kumaresan and D. Tufts, "Estimating the angles of arrival of multiple plane waves," *IEEE Transactions On Aerospace And Electronic Systems*, vol. AES-19, no. 1, pp. 134–139, 1983.
- [4] A. H. Sayed, *Fundamentals of Adaptive Filtering*. Wiley, NY, 2003.
- [5] B. Yang, "Projection approximation subspace tracking," *IEEE Transactions on Signal Processing*, vol. 43, no. 1, pp. 95–107, 1995.
- [6] K. Crammer, "Online tracking of linear subspaces," *In Proc. COLT 2006*, vol. 4005, pp. 438–452, 2006.
- [7] E. J. Candés and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Trans. Inform. Theory*, vol. 56, no. 5, pp. 2053–2080, 2009.
- [8] L. Balzano, B. Recht, and R. Nowak, "High-dimensional matched subspace detection when data are missing," in *Proc. ISIT*, June 2010.
- [9] L. Balzano, R. Nowak, and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," *Proc. Allerton 2010*, 2010.
- [10] W. Dai, O. Milenkovic, and E. Kerman, "Subspace Evolution and Transfer (SET) for Low-Rank Matrix Completion," *IEEE Trans. Signal Processing*, p. submitted, 2010.
- [11] R. Roy and T. Kailath, "ESPRIT—Estimation of signal parameters via rotational invariance techniques," *IEEE Trans. on Acoustics, Speech, Signal Processing*, vol. 37, no. 7, pp. 984–995, Jul. 1989.
- [12] K. Gedalyahu and Y. C. Eldar, "Time-delay estimation from low-rate samples: A union of subspaces approach," *IEEE Trans. on Signal Processing*, vol. 58, no. 6, pp. 3017–3031, 2010.
- [13] Z. Wen, W. Yin, and Y. Zhang, "Solving a low-rank factorization model for matrix completion by a non-linear successive over-relaxation algorithm," *Rice CAAM Tech Report TR10-07*, 2010.
- [14] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and Bregman iterative methods for matrix rank minimization," *Mathematical Programming*, vol. 1, no. 1, pp. 1–27, 2009.
- [15] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1–28, 2008.
- [16] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from noisy entries," *Preprint*, pp. 1–9, 2009. [Online]. Available: <http://arxiv.org/abs/0906.2027v1>