Federated Over-the-Air Subspace Learning and Tracking from Incomplete Data

Praneeth Narayanamurthy, Namrata Vaswani and Aditya Ramamoorthy
Iowa State University
{pkurpadn, namrata, adityar}@iastate.edu

Abstract

We consider a federated learning scenario where K peer nodes communicate with a master node via a wireless channel using the newly developed "over-the-air" superposition and broadcast paradigm. This means that (i) data transmitted from the nodes is directly summed at the master node using the superposition property of the wireless channel; and (ii) the master broadcasts this sum, or a processed version of it, to all the nodes. The implicit assumption here is that the aggregation to be performed at the master node is an additive operation. This new transmission mode is enabled by advances in wireless technology that allow for synchronous transmission by the K peer nodes. It is K times time- or bandwidth- efficient compared to the traditional digital transmission mode, but the tradeoff is that channel noise corrupts each iterate of the underlying ML algorithm being implemented. Additive noise in each algorithm iterate is a completely different type of perturbation than noise or outliers in the observed data. It introduces a novel set of challenges that have not been previously explored in the literature. In this work, we develop and analyze federated over-the-air solutions to two well-studied problems in unsupervised learning: (i) subspace learning and (ii) subspace tracking from incomplete data.

1 Introduction

Federated learning refers to a distributed learning scenario in which users/nodes keep their data private but only share intermediate locally computed iterates with the master node. The master, in turn, shares a global aggregate of these iterates with all the nodes at each iteration. There has been extensive recent work on solving ML problems in a federated setting [1–3] but all these assume a perfect channel between the peer nodes and the master. This is a valid assumption in the traditional "digital" transmission mode in which different peer nodes transmit in different time or frequency bands, and appropriate channel coding is done at lower network layers to enable error-free recovery with very high probability. Advances in wireless communication technology now allow for synchronous transmission by the various peer nodes[1] and thus enable an alternate computation/communication paradigm for learning algorithms in which the aggregation step at the master is a summation operation. In this alternate paradigm, the summation can be performed "over-the-air" using the superposition property of the wireless channel and the summed aggregate (or a processed version) can be broadcasted to all the nodes [4–6]. Assuming K peer nodes, this over-the-air addition is K-times more time- or bandwidth-efficient than the traditional mode. The

[1]Small amounts of asynchronism may occur and these can be handled using standard physical layer communication techniques (use piloting to estimate the amount of asynchronism, and repeat each symbol a few times to compensate for it).

tradeoff is that there is no error-correction redundancy, and hence, additive channel noise and channel fading effects corrupt the transmitted data. Fading can be estimated and compensated for using standard techniques (use pilots for estimation, use multiple receiver antennas and least squares for compensation) [7]. The main issue to be tackled therefore is the additive channel noise which now corrupts each algorithm iterate. This introduces a new, and very different, set of challenges in ML algorithm design and analysis compared to what has been explored in existing literature. The reason is that channel noise corrupts *each algorithm iterate and not the data.* In this work, we develop and analyze *federated over-the-air solutions* to two well-studied problems in unsupervised learning: (i) subspace learning, and (ii) subspace tracking from incomplete data. These problems have important applications in video analytics [8] , social network activity learning [9] and recommendation system design [10]. To the best of our knowledge, this work is the *first systematic attempt to investigate the effect of iteration noise on an ML algorithm.*

Problem setting. Assume that there are K distributed worker or peer nodes and one master node. Assume that node k observes the local data matrix $\boldsymbol{Y}_k \in \mathbb{R}^{n\times d_k}$, and let $\boldsymbol{Y} := [\boldsymbol{Y}_1, \boldsymbol{Y}_2, \cdots, \boldsymbol{Y}_K] \in \mathbb{R}^{n\times d}$ with $d = \sum_k d_k$ denote the complete data matrix. The goal of *fixed* subspace learning or PCA is to compute an r-dimensional subspace approximation in which the data matrix $\boldsymbol{Y}$ approximately lies. We use a *basis matrix* (tall matrix with orthonormal columns) $\boldsymbol{U}$ to denote this subspace.

A second, and more general problem we study is that of subspace learning and tracking from incomplete data. In this setup, at each time index $t = 1, \cdots, T_{\max}$, let α_k denote[2] the number of (possibly incomplete) n-dimensional data points at node node k, and let $\alpha := \sum_k \alpha_k$ denote the total amount of data at each time. Thus, $\boldsymbol{Y}_{k,(t)}$ is an $n \times \alpha_k$ matrix at node k and time t and let $\boldsymbol{Y}_{(t)} := [\boldsymbol{Y}_{1,(t)}, \cdots, \boldsymbol{Y}_{k,(t)}] \in \mathbb{R}^{n\times\alpha}$ denote the full data matrix at time t. Notice that after $T_{\max}$ time instants have elapsed, node k has now observed $d_k = T_{\max}\alpha_k$ vectors, similarly $d := \sum_k d_k$ is the total number of vectors observed across all nodes. We denote the "true" low-rank matrix at time t by $\boldsymbol{L}_{(t)}$. We would like to learn its column span at each time t, or every so often. Let $\boldsymbol{y}_i$ denote column i of the matrix $\boldsymbol{Y}_{k,(t)}$ (this is technically $\boldsymbol{y}_{i,k,(t)}$ but to keep the notation simple, we use $\boldsymbol{y}_i$ when the meaning is clear). We use $\mathcal{M}_i$ to denote the set of missing entries (whose values are set to 0), so that $(\mathcal{M}_i)^c$ (complement set of $\mathcal{M}_i$ w.r.t. $[n]$) is the set of observed entries. Thus, $\boldsymbol{y}_i$ satisfies

$$\boldsymbol{y}_i = \mathcal{P}_{\mathcal{M}_i^c}(\boldsymbol{\ell}_i) + \boldsymbol{v}_i, \quad i \in \mathcal{I}_{k,(t)}, \quad k \in [K] \tag{1}$$

where $\mathcal{P}$ is a binary mask, $\boldsymbol{\ell}_i = \boldsymbol{P}_{(i)}\boldsymbol{a}_i$ is one column of $\boldsymbol{L}_{(t)}$, with $\boldsymbol{P}_{(i)} \in \mathbb{R}^{n\times r}$, are the fixed or slowly changing subspace matrices, $\boldsymbol{a}_i \in \mathbb{R}^r$ are the subspace coefficient vectors, and $\boldsymbol{v}_i$ is the noise/modeling error. $\mathcal{I}_{k,(t)}$ denotes the set of vectors observed by k-th node at time t. Note that the subspace matrices are also technically $\boldsymbol{P}_{i,k,(t)}$ and thus, by slowly changing, we mean that $\boldsymbol{P}_{i,k,(t)}$'s vary slowly across time but not across the individual nodes. However, we denote them as $\boldsymbol{P}_{(i)}$ for simplicity. We formally define the "slowly changing model" in Sec. 3. The goal here is to detect the change, and track the subspaces $\boldsymbol{P}_{(i)}$ quickly and reliably.

In this work, we solve both the above problems in a federated over-the-air setup and the overall framework for an algorithm that respect these constraints is as follows

- At algorithm iteration l, the master node broadcasts the previous global estimate of the subspace, $\hat{\boldsymbol{Q}}_{l-1}$ (which is an estimate of the span of $\boldsymbol{U}$ or $\boldsymbol{P}_{(i)}$), to all the nodes.

[2]For notational simplicity, we assume that α_k's are constant over time. Our guarantees are not affected if they are functions of time as long as the total number of vectors, $\sum_k \alpha_k$ is lower-bounded as in Theorem 3.3.

- Each node uses this estimate and its local data matrix $\boldsymbol{Y}_k$ ($\boldsymbol{Y}_{k,(t)}$) to compute a new local estimate denoted $\tilde{\boldsymbol{U}}_{k,l}$.
- All the K nodes synchronously transmit $\tilde{\boldsymbol{U}}_{k,l}$ to the master node. The master node observes the sum of all the transmissions (over-the-air addition); but, the sum is corrupted by channel noise, $\boldsymbol{W}_l$. Thus, the master receives $\hat{\boldsymbol{U}}_l := \sum_{k=1}^{K} \tilde{\boldsymbol{U}}_{k,l} + \boldsymbol{W}_l$. We assume that each entry of the channel noise is i.i.d. Gaussian, zero-mean with variance σ_c^2.
- The master processes $\hat{\boldsymbol{U}}_l$ to get $\hat{\boldsymbol{Q}}_l$ and broadcasts it to all K worker nodes for next iteration.

Contributions. In the first part of this work, we study a power method (PM) based algorithm [11, 12], federated over-the-air PM (FedPM) for subspace learning. We show that, if the standard deviation of iteration channel noise is at most ϵ-times the r-th eigenvalue of $\boldsymbol{Y}\boldsymbol{Y}^T$, and if the ratio between the $(r+1)$-th and r-th eigenvalues is at most 0.99, then, with high probability, we can solve the problem to ϵ-accuracy, in at most $L = O(\log(1/\epsilon))$ iterations. We also consider two simple modifications of PM to improve noise robustness and convergence rate respectively. One special case of our result recovers the result of [13, 14] which studies the perturbed PM for a very different set of problems.

The second, and most important contribution of this work is a simple, fast, and provably correct solution for subspace tracking with missing entries in the data (ST-miss) that satisfies the federated over-the-air constraints (see Fig. 2). This work also improves upon all past works on centralized ST-miss solutions [15–18]. Unlike all these papers, we provide *provable* results for general time-varying subspaces - both the piecewise constant setting (Theorem 3.3) and the more general, subspace change at each time setting (Corollary 3.4). The overall algorithm idea is adapted from a recently studied work for centralized ST-miss [18]. In this framework, at each time (and iteration time) t, we locally solve a projected least squares (LS) problem for each individual data vector to estimate its missing entries, followed by federated computation of the top r singular vectors of the resultant global matrix. By carefully combining the result that we prove in the first part for FedPM with an extended version of (centralized) PCA in sparse data-dependent noise result [19], we prove the following guarantee. Fed-ST-Miss tracks time-varying subspace(s) to ϵ accuracy within $O(\log(1/\epsilon))$ time instants if (i) channel noise is small enough; (ii) the subspace "changes slowly enough" for at least $\Omega(\log(1/\epsilon))$ time instants, and the number of data points at time t, $\alpha = \Omega(r \log n)$; (iii) the number of missing entries in any row of this matrix is at most $O(1)$ times α, while the number in any column is $(1/r)$ times n; (iv) the subspaces satisfy the standard μ-incoherence assumption, and the subspace coefficients are i.i.d., zero mean, and bounded random vectors.

While there have been a few recent works on algorithms that exploit over-the-air aggregation [5,20] there are no recovery guarantees for these algorithms. Thus, to our best knowledge, this is the *first result for federated over-the-air learning for any ML problem.* The key technical challenge is to analyze the effect of the extra *channel noise* on *each algorithm iterate* and to ensure that the algorithm still converges to the correct solution.

Related Work. In terms of the problem, our work is closest to the recent array of papers [5,20] on developing stochastic gradient descent (SGD) based algorithms in the *federated over-the-air aggregation* setting. But these works focus on optimizing resource allocation to satisfy transmit power constraints and not on performance guarantees for the perturbed algorithm.

A related line of work is in developing *federated algorithms*, albeit not in the over-the-air aggregarion mode. Recent works such as [1, 21] attempt to empirically optimize the *communication efficiency* and show significant gains for a slew of learning tasks, Similarly, [22] studies the problem

Table 1: Comparing bounds on channel noise variance σ_c^2 and on number of iterations L. Let $\text{gap}_1 := \lambda_r - \lambda_{r+1}$, $\text{gap}_q := \lambda_r - \lambda_{q+1}$ for some $r \le q \le r'$. Also, we assume $\epsilon \le c/r$.

	Noisy Power Method [13, 14]	This Work
$\tau = 1$ $r' = r$	$\sigma_c = \mathcal{O}\left(\frac{\text{gap}_1 \epsilon}{\sqrt{n}}\right)$	$\sigma_c = \mathcal{O}\left(\frac{\lambda_r \epsilon}{\sqrt{n}}\right)$, $R < 0.99$
Random init	$L = \Omega\left(\frac{\lambda_r}{\text{gap}_q} \log\left(\frac{n}{\epsilon}\right)\right)$	$L = \Omega\left(\frac{1}{\log(1/R)} \log\left(\frac{n}{\epsilon}\right)\right)$
Good init ($\text{dist}_0 \le c_0$)	-	$L = \Omega\left(\frac{1}{\log(1/R)} \log\left(\frac{1}{\epsilon}\right)\right)$
$\tau = 1$ $r' > r$	$\sigma_c = \mathcal{O}\left(\frac{\text{gap}_q \epsilon}{\sqrt{n}}\right)$	–
$\tau > 1$ $r' = r$	–	$\sigma_c = \mathcal{O}\left(\frac{\lambda_r \epsilon}{R^\tau \tau}\right)$, $R < 0.99$, $\lambda_r > 1$
$\tau = \mathcal{O}(\log n)$ $r' = r$	–	$\sigma_c = \mathcal{O}\left(\frac{\lambda_r \epsilon n}{\log n}\right)$, $R < 0.99$, $\lambda_r > 1$

adaptive federated PCA algorithm, but there are no performance guarantees. Our algorithm and analysis is generalization of this setting since we also consider additive channel noise at each iteration. A second, tangentially, related line of work investigates *distributed algorithms* for PCA, ST-miss, and low-rank matrix completion (LRMC) but the goal here is to design provable, parallelizable algorithms in a decentralized setting. For example, there is a large amount of literature on distributed PCA algorithms as discussed in the review paper [10] and references therein; there is also some recent work [12, 23] that develop provable algorithms for distributed LRMC. However, both these lines of work are starkly different from our problem since in the above setting, communication only occurs *after the end of local computation.* In addition, there exist other heuristics for LRMC, robust ST and SGD based approaches (that possibly consider byzantine nodes) such as [24–27].

The algorithm that we analyze for the *fixed* subspace learning problem is similar to a meta algorithm studied in [13, 14] a very different context. We discuss this in detail in Sec. 2. The other set of related work on provable LRMC [8, 28, 29] and ST-miss [16–18, 30, 31] are discussed in Sec. 3.

2 Federated Over-the-Air Subspace Learning

The simplest algorithm for subspace learning is the power method (PM) [11]. The distributed PM is well known, but most previous works assume the noise-free setting, e.g., see the review in [10]. It proceeds as follows: at each iteration l, each node k computes $\tilde{\boldsymbol{U}}_{k,l} := \boldsymbol{Y}_k \boldsymbol{Y}_k^T \hat{\boldsymbol{Q}}_{l-1}$ and transmits it to the master which computes their sum followed by QR decomposition of the sum. But since we are assuming over-the-air summation, the sum itself is corrupted by channel noise. Thus, at every iteration l, instead of receiving $\sum_k \boldsymbol{Y}_k \boldsymbol{Y}_k^T \hat{\boldsymbol{Q}}_{l-1} = \boldsymbol{Y}\boldsymbol{Y}^T \hat{\boldsymbol{Q}}_{l-1}$, the master receives

$$\hat{\boldsymbol{U}}_l := \boldsymbol{Y}\boldsymbol{Y}^T \hat{\boldsymbol{Q}}_{l-1} + \boldsymbol{W}_l.$$

where $\boldsymbol{W}_l$ is the channel noise. The master computes a QR decomposition of $\hat{\boldsymbol{U}}_l$ either at every iteration l or after every τ iterations. The latter helps improve noise robustness. This is broadcast

Algorithm 1 FedPM: Wireless Federated Power Method

Input: $\boldsymbol{Y}$, rank r, L iterations,QR decomp. "frequency" τ, K worker nodes, $\boldsymbol{y}_i$ for each $i \in \mathcal{I}_k$

1: At master node, $\hat{\boldsymbol{U}}_0 \overset{i.i.d.}{\sim} \mathcal{N}(0, I)_{n\times r}$; $\hat{\boldsymbol{Q}}_0 \leftarrow \hat{\boldsymbol{U}}_0$, transmit to all K workers.
2: **for** $l = 1, \ldots, L\tau$ **do**
3: At k-th worker node, do $\tilde{\boldsymbol{U}}_{k,l} = \boldsymbol{Y}_k \boldsymbol{Y}_k^T \hat{\boldsymbol{Q}}_{l-1}$
4: All k nodes transmit $\tilde{\boldsymbol{U}}_{k,l}$ synchronously to the master.
5: Master receives $\hat{\boldsymbol{U}}_\tau := \sum_k \tilde{\boldsymbol{U}}_{k,l} + \boldsymbol{W}_{k,l}$, with $\sum_k \boldsymbol{W}_{k,l} = \boldsymbol{W}_l$.
6: $\hat{\boldsymbol{Q}}_l \leftarrow \hat{\boldsymbol{Q}}_{l-1}$
7: **if** $(l \mod \tau) = 0$ **then** $\hat{\boldsymbol{Q}}_l \boldsymbol{R}_l \overset{QR}{\leftarrow} \boldsymbol{U}_l$ **end if**
8: Master broadcasts $\hat{\boldsymbol{Q}}_l$ to all nodes
9: **end for**
10: All k nodes compute $\boldsymbol{Y}_k \boldsymbol{Y}_k^T \hat{\boldsymbol{Q}}_L$, transmit synchronously to master node
11: Master receives $\boldsymbol{B} = \sum_k \boldsymbol{Y}_k \boldsymbol{Y}_k^T \hat{\boldsymbol{Q}}_L + \boldsymbol{W}_L$, computes the top eigenvalue, $\hat{\lambda}_1 = \lambda_{\max}(\hat{\boldsymbol{Q}}_L^T \boldsymbol{B})$.

Output: $\hat{\boldsymbol{Q}}_L$, $\hat{\lambda}_1$.

back to all the user nodes for use in the next iteration. We summarize the complete approach in Algorithm 1. Notice that it can either use random or a "good" initialization. The latter is easy to get in the tracking setting (see Sec. 3) and helps speed up algorithm convergence significantly.

Subspace Recovery Guarantee. We use the sine of the maximum principal angle as the metric to quantify the distance between subspaces. For two subspaces that correspond to the spans of $n \times r$ basis matrices, $\boldsymbol{U}_1$, $\boldsymbol{U}_2$, it is computed as $\text{dist}(\boldsymbol{U}_1, \boldsymbol{U}_2) = \|(\boldsymbol{I} - \boldsymbol{U}_1 {\boldsymbol{U}_1}^T)\boldsymbol{U}_2\|$. Here and below $\|.\|$ denotes the induced 2-norm of a matrix. We use dist_l to denote $\text{dist}(\hat{\boldsymbol{Q}}_l, \boldsymbol{U})$. We reuse the letter C to denote different numerical constants in each use.

Let λ_i denote the i-th eigenvalue of $\boldsymbol{Y}\boldsymbol{Y}^T$. Also, define the following quantities: the ratio of $(r+1)$-th to r-th eigenvalue, $R := \lambda_{r+1}/\lambda_r$, the noise to signal ratio, $\text{NSR} := \sigma_c/\lambda_r$, and $\tilde{R} := \max(R, 1/\lambda_r)$. Thus we have the following main result:

Theorem 2.1. *Consider Algorithm 1 with initial subspace estimation error* dist_0.

1. *Let* $\tau = 1$. *Assume that* $R < 0.99$. *If, at each iteration, the channel noise* $\boldsymbol{W}_l$ *satisfies* $\text{NSR} < c \min\left(\frac{\epsilon}{\sqrt{n}}, 0.2\sqrt{\frac{1-\text{dist}_{l-1}^2}{r}}\right)$ *then, after* $L = \Omega\left(\frac{1}{\log(1/R)}\left(\log\frac{1}{\epsilon} + \log\frac{1}{\sqrt{1-\text{dist}_0^2}}\right)\right)$ *iterations, with probability at least* $1 - L\exp(-cr)$, $\text{dist}(\boldsymbol{U}, \hat{\boldsymbol{Q}}_L) \le \epsilon$.
2. *Let* $\tau > 1$. *If* $\lambda_r > 1$, *and if* $\text{NSR} < c \min\left(\frac{\epsilon}{\sqrt{n}} \cdot \frac{1}{\sqrt{\tau}R^{\tau-1}}, 0.2\sqrt{\frac{\lambda_r^2-1}{\lambda_r^2}} \cdot \sqrt{\frac{1-\text{dist}_{(l-1)\tau}^2}{r}}\right)$, *then the above conclusion holds.*
3. *If* $\hat{\boldsymbol{U}}_0 \overset{i.i.d}{\sim} \mathcal{N}(0, \boldsymbol{I})_{n\times r}$, *then* $\text{dist}_0 = \mathcal{O}(\sqrt{1 - 1/\gamma nr})$ *with probability* $1 - 1/\gamma$.

To understand the above theorem, first consider $\tau = 1$. In this case, we require $\text{NSR}\sqrt{n} < \epsilon$ to achieve ϵ-accurate recovery of the subspace. In this setting, with a random initialization, our result essentially recovers the main result of [13, 14]. But we can choose to pick $\tau > 1$. To understand its advantage, suppose that $\lambda_r > 1.5$ (this is easy to satisfy by assuming that all the data transmitted is scaled by a large enough factor). Then, clearly, $\lambda_r^2/(\lambda_r^2 - 1) < 3$ and so the first term in the upper

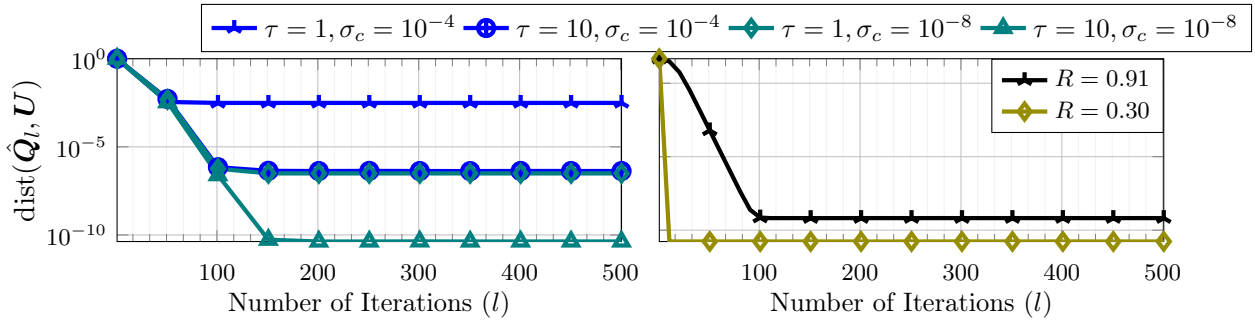

Figure 1: Numerical verification of Theorem 2.1: **Left:** increasing τ increases robustness to noise; **Right:** Increasing the "gap" helps achieve faster convergence *and* lower recovery error.

bound of NSR dominates. Thus, as τ is increased, we only require $\mathrm{NSR}\sqrt{n} \cdot \sqrt{\tau} R^{\tau-1} \leq \epsilon$ which is a significantly weaker requirement. Thus, a larger τ means we can allow the noise variance to be larger. However, we cannot pick τ too large because it will lead to numerical problems (bit overflow problems) and may also result in violation of the transmit power constraint. As an example, if we set $\tau = C \log n$, for a constant C that is large enough (depends on $\tilde{R}$), then the we only require $(\mathrm{NSR}\sqrt{n}/\log n) \leq \epsilon$ which provides a $\log n$ factor of noise robustness. Observe that the number of iterations needed, L, depends on the initialization. If $\mathrm{dist}_0 < c_0$ with c_0 being a constant, then we only need $L = \Omega\left(\frac{1}{\log(1/R)} \log(1/\epsilon)\right)$ iterations (which we leverage in the ST-miss result). Finally, if we use random initialization we need $L = \Omega\left(\frac{1}{\log(1/R)} \log(nr/\epsilon)\right)$, i.e., $O(\log nr)$ more iterations. We validate these experimentally in Fig. 1. We provide the result for first eigenvalue estimate (Line 11, Algorithm 1) as Corollary A.6 in the Appendix.

Discussion. We note that a similar algorithm dubbed "Noisy Power Method" [13, 14] has been studied in a different context. The authors analyze perturbed PM for two reasons: (i) a solution for streaming PCA can be understood as perturbed versions of noisy PM; (ii) for solving the private PCA problem, carefully designed random noise is added at *each iteration* in order to preserve privacy. In (i), no statistical model can be assumed on the noise and one needs worst-case bounds, while in (ii), the algorithm deliberately simulates and adds *just enough noise* so that an attacker cannot distinguish two data points from one another. Our problem setting is easier than (i) but harder than (ii) because we cannot design the noise statistics ourselves. Since [14] improves upon the result of [13], we compare with its result in Table 1. Both these papers attempt to learn an $r' \geq r$ dimensional subspace in order to improve the noise robustness of PM. Observe that when $\tau = 1$ and $r' = r$, we essentially recover the results of [14] up to constant factors. When $r' > r$, our result does not apply. But when considering approximate low-rank matrices so that $\mathrm{gap}_1 \approx \mathrm{gap}_q \approx \lambda_r$, our result is still comparable in this case. If $\tau > 1$, we require a weaker bound on channel noise than what [13, 14] need. We validate this through numerical experiments in Sec. 4 (Fig. 1). Finally, when a good initialization available, the number of iterations required reduces by a constant times $\log n$, which we leverage in the proof of the subspace tracking problem.

3 Federated Over-the-Air ST-miss

As explained in the provable subspace tracking literature [32], we need to assume a piecewise constant subspace change model, wherein, the $n \times r$-dimensional subspace is fixed for a few data points. This is required since if the subspace changes for each data point, then we have nr unknowns, but only n equations, and this is general cannot be solved. A necessary requirement is that the subspace is

fixed for at least r points, but we will see in the Theorem 3.3 that $\Omega(r \log n \log(1/\epsilon))$ vectors suffice to obtain an ϵ-accurate recovery. We use t_j's to denote the subspace change times and thus, for all $t \in [t_j, t_{j+1})$, the i-th column of the true data matrix $\boldsymbol{L}_{(t)}$ can be written as

$$\boldsymbol{\ell}_i = \boldsymbol{P}_{i,k,(t)} \boldsymbol{a}_i = \boldsymbol{P}_j \boldsymbol{a}_i$$

In practice, however, typically the subspaces change by a little at each time and do not follow the above assumption. As we explain later this can be modeled as piecewise constant subspace change plus modeling error $\boldsymbol{v}$ (Corollary 3.4. This explains why ST-miss algorithms work for video data.

In the rest of the section we will propose and analyze an algorithm to accurately estimate the $\boldsymbol{P}_j$'s in the federated, over-the-air setting. Below we explain the main idea of the algorithm, and we provide the complete algorithm as Algorithm 2 in the Appendix.

Algorithm Idea. We use the overall algorithmic idea of the approach from [18] (solves LRMC and subspace tracking with missing entries) since it can be easily modified to develop a federated over-the-air subspace tracking algorithm. At each time t, it consists of (i) a projected least squares (LS) step applied locally to each individual data vector, $\boldsymbol{y}_i$, to estimate the missing entries, and hence, and get an estimate of $\boldsymbol{\ell}_i$ denoted $\hat{\boldsymbol{\ell}}_i$, followed by (ii) a subspace update step which toggles between the "subspace update" mode, to obtain refined estimates of the subspace, and the "change detect" mode, (which the algorithm enters *after* the current subspace has been estimated to ϵ-accuracy) to provably detect subspace changes.

Projected LS: The projected LS problem is a column-wise operation that is solved locally for each $\boldsymbol{y}_i$. There is a slight difference between $t = 1$ and the rest. For $t > 1$ and $t \in [t_j, t_{j+1})$, it proceeds as follows. Let $\hat{\boldsymbol{P}}_{j,t-1}$ denote the $t-1$-th estimate (this is a basis matrix) and let $\boldsymbol{\Psi} = \boldsymbol{I} - \hat{\boldsymbol{P}}_{j,t-1} \hat{\boldsymbol{P}}_{j,t-1}^T$ denote a matrix to project orthogonal to it. The following gives "an estimate" of the missing entries:

$$\hat{\boldsymbol{\ell}}_i = \boldsymbol{y}_i - \boldsymbol{I}_{\mathcal{M}_i} \boldsymbol{\Psi}_{\mathcal{M}_i}^{\dagger} (\boldsymbol{\Psi} \boldsymbol{y}_i) \tag{2}$$

The above uses the fact that $\boldsymbol{y}_i$ can be written as $\boldsymbol{y}_i = -\boldsymbol{I}_{\mathcal{M}_i} (\boldsymbol{I}_{\mathcal{M}_i}^T \boldsymbol{\ell}_i) + \boldsymbol{\ell}_i + \boldsymbol{v}_i$ and that $\boldsymbol{\ell}_i = \boldsymbol{P}_j \boldsymbol{a}_i$. Projecting $\boldsymbol{y}_i$ orthogonal to $\hat{\boldsymbol{P}}_{j,t-1}$ helps mostly nullify $\boldsymbol{\ell}_i$ but gives projected measurements of the vector of missing entries, $(\boldsymbol{I}_{\mathcal{M}_i}^T \boldsymbol{\ell}_i)$. These are then recovered via LS while treating $\boldsymbol{\Psi}\boldsymbol{\ell}_i + \boldsymbol{v}_i$ as the "noise" seen by the LS step. Thus, estimate $\hat{\boldsymbol{\ell}}_i$ satisfies

$$\hat{\boldsymbol{\ell}}_i = \boldsymbol{\ell}_i + \boldsymbol{e}_i, \text{ where } \boldsymbol{e}_i := -\boldsymbol{I}_{\mathcal{M}_i} \left(\boldsymbol{\Psi}_{\mathcal{M}_i}\right)^{\dagger} \boldsymbol{\Psi} \boldsymbol{\ell}_i \tag{3}$$

The above is done for each data vector $\hat{\boldsymbol{\ell}}_i$, $i \in \mathcal{I}_k(t)$ at each node k. After this step, we have the estimates' sub-matrix $\hat{\boldsymbol{L}}_{k,(t)}$ at node k. At $t = 1$, one starts with a zero initialization of the subspace and thus the projected LS step does not do anything.

Subspace Update: This computes the top r singular vectors of the matrix formed by the entire batch of $\hat{\boldsymbol{\ell}}_i$'s at all the nodes at the current time instant using FedPM (Algorithm 1) to estimate the $\hat{\boldsymbol{P}}_{j,t}$ using the $n \times \alpha$ matrix $\hat{\boldsymbol{L}}_{(t)} := [\hat{\boldsymbol{L}}_{1,(t)}, \hat{\boldsymbol{L}}_{2,(t)}, \hat{\boldsymbol{L}}_{K,(t)}]$. Observe that the error $\boldsymbol{e}_i$ is sparse with support $\mathcal{M}_i$ and it depends linearly on the true data $\boldsymbol{\ell}_i$. The problem of recovering the subspace of $\boldsymbol{\ell}_i$ from this type of data is one of PCA in sparse data-dependent noise. This centralized version of this problem has been studied in recent work [19]. We use (an improved version of) this result to argue that (i) $\hat{\boldsymbol{L}}_{(t)} \hat{\boldsymbol{L}}_{(t)}{}^T$ has good eigengap. Thus Theorem 2.1 implies that, assuming small enough channel noise, FedPM returns the "correct" estimate of the span of the top r singular vectors of $\hat{\boldsymbol{L}}_{(t)}$. Moreover, we also show that the estimated span is a better approximation of the span of columns of

$\boldsymbol{P}_j$ than the previous one. This proof requires a careful application of the max-miss-frac-row bound and the fact that the error $\boldsymbol{e}_i$ is sparse with changing support. Again, at $t = 1$, we compute the top r singular vectors of $\hat{\boldsymbol{L}}_{(0)} = \boldsymbol{Y}_{(0)}$ using FedPM. A combination of the two results described above is used to show that this step returns a good enough estimate of $\boldsymbol{P}_1$, i.e., that $\text{dist}(\hat{\boldsymbol{P}}_{1,1}, \boldsymbol{P}_1) \leq 0.1$. One then uses this estimate to solve projected LS at $t = 2$ to fill missing entries followed by a second subspace estimation step to get a better estimate of the subspace. We then argue that each new subspace estimate is better than the previous one because the errors $\boldsymbol{e}_i$ in the estimates $\hat{\boldsymbol{\ell}}_i$ are smaller at t than at $t-1$ and thus at $t = T = C\log(1/\epsilon)$, we get an ϵ-accurate subspace estimate.

Change Detect: Assume that the j-th subspace, $\boldsymbol{P}_j$ has been estimated to ϵ-accuracy, i.e. we have completed T subspace update steps. The key idea for detecting change is to consider the matrix $\boldsymbol{B} := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{j,T}\hat{\boldsymbol{P}}_{j,T}^T)\hat{\boldsymbol{L}}_{(t)}$. If the subspace has not changed, this matrix will be nearly zero, and "large" otherwise. We explain this idea in detail in Appendix A. Thus, a simple way to detect change is to compute any of the first r singular values of $\boldsymbol{B}$ and check if it is above a threshold or not. This can be implemented by broadcasting $\hat{\boldsymbol{P}}_{j,T}$ to all the nodes, which then project their local $\hat{\boldsymbol{L}}_{k,(t)}$ matrices orthogonal to it and then implementing FedPM with $r = 1$ to compute the top eigen-vector and value of $\boldsymbol{B}\boldsymbol{B}^T$ and check if this is above a carefully chosen threshold (see Thm. 3.3).

Assumptions needed for identifiability. It is well known from the LRMC literature [8,28,29] that we need to assume incoherence (w.r.t. the standard basis) of the left and right singular vectors of the matrix. In this vein, we assume incoherence of the subspace basis matrices, $\boldsymbol{P}_j$, i.e., assume that for all j, for some constant μ, the following holds $\max_i \|\boldsymbol{P}_j^{(i)}\|_2^2 \leq \mu r/n$ where $\boldsymbol{P}_j^{(i)}$ denotes the i-th row of $\boldsymbol{P}_j$. Since we study the subspace tracking problem, we use the following statistical model on the subspace coefficients in lieu of right μ-incoherence.

Definition 3.1 (Statistical Right μ-Incoherence)**.** *Assume that all the $\boldsymbol{a}_i$'s are zero mean; mutually independent; have identical diagonal covariance matrix $\boldsymbol{\Lambda}$, i.e., that $\mathbb{E}[\boldsymbol{a}_i\boldsymbol{a}_i^T] = \boldsymbol{\Lambda}$ with $\boldsymbol{\Lambda}$ diagonal; and are bounded such that $\max_i \|\boldsymbol{a}_i\|^2 \leq \mu r \lambda_{\max}(\boldsymbol{\Lambda})$.*

Moreover, if a few complete rows (columns) of the entries are missing, it is impossible to recover the underlying matrix. This can be avoided by either assuming bounds on the number of missing entries in any row and in any column, or by assuming that each entry is observed uniformly at random with probability ρ independent of all others. While most work assumes the Bernoulli model, in this work we assume the former which is a much weaker requirement. We need the following definition.

Definition 3.2 (Missing Entry Fractions)**.** *Consider the $n \times \alpha$ observed matrix $\boldsymbol{Y}_{(t)} := [\boldsymbol{Y}_{1,(t)}, \ldots, \boldsymbol{Y}_{K,(t)}]$ at time t across all the K nodes. We use max-miss-frac-col (max-miss-frac-row) to denote the maximum of the fraction of missing entries in any column (row) of this matrix.*

Before stating the main result, we need to define a few quantities. Recall that $\boldsymbol{\Lambda} := \mathbb{E}[\boldsymbol{a}_i\boldsymbol{a}_i{}^T]$. Let $\lambda^+ := \lambda_{\max}(\boldsymbol{\Lambda})$, $\lambda^- := \lambda_{\min}(\boldsymbol{\Lambda})$, Also assume for simplicity in stating the results that the condition number of the covariance matrix of the data, $f := \lambda^+/\lambda^-$ is a numerical constant.

Theorem 3.3. *Consider Algorithm 2. Assume $\boldsymbol{v}_{i,(t)}$ are i.i.d. zero-mean, bounded r.v's; independent of $\boldsymbol{L}_{(t)}$. Let $\lambda_v^+ := \|\mathbb{E}[\boldsymbol{v}_{i,(t)}\boldsymbol{v}_{i,(t)}^T]\|$ and $\max_{i,t} \|\boldsymbol{v}_{i,(t)}\|^2 \leq Cr\lambda_v^+$ and that $\text{dist}(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j) \geq c\sqrt{\lambda_v^+/\lambda^-}$. Pick an ϵ that satisfies $c\sqrt{\lambda_v^+/\lambda^-} \leq \epsilon \leq 0.2$. Set $T := C\log(1/\epsilon)$, $L = C\log(n/\epsilon_t)$ with $\epsilon_t := \max(\epsilon, 0.01 \cdot (0.3)^{t-1})$ and the detection threshold, $\omega_{evals} = 2\epsilon^2\lambda^+$. Assume that the following hold: $\alpha \in \Omega(r \log n)$*

1. **Incoherence:** *$\boldsymbol{P}_j$'s satisfy μ-incoherence, and $\boldsymbol{a}_i$'s satisfy statistical right μ-incoherence;*

2. **Missing Entries:** *max-miss-frac-col $\in O(1/\mu r)$, max-miss-frac-row $\in O(1)$;*

3. **Channel Noise:** *the channel noise seen by each FedPM iteration is mutually independent at all times, isotropic, and zero mean Gaussian with variance $\sigma_c^2 \leq \epsilon_t \lambda^- / \sqrt{n}$.*

4. **Piecewise constant subspace:** *the subspace is constant for at least $T_{cons} = \Omega(\log(1/\epsilon)$ time instants, i.e., $t_{j+1} - t_j > T_{cons}$ for all j;*

then, with probability at least $1 - 10dn^{-10} - c\gamma$,

$$\operatorname{dist}(\hat{\boldsymbol{P}}_{j,t}, \boldsymbol{P}_j) \leq \begin{cases} \max(0.01 \cdot (0.3)^{t-1}, \varepsilon) & \text{if } t < T \\ \epsilon & \text{if } t = T. \end{cases}$$

Additionally, $\|\hat{\boldsymbol{\ell}}_{i,(t)} - \boldsymbol{\ell}_{i,(t)}\| \leq 1.2 \cdot \operatorname{dist}(\hat{\boldsymbol{P}}_{j,t}, \boldsymbol{P}_j) \|\boldsymbol{\ell}_{i,(t)}\|$ for all i and t, and the j-th subspace change is detected within at most 1 time instant, i.e., $t_j \leq \hat{t}_j \leq t_j + 1$.

Time complexity at node k: $\mathcal{O}(n\alpha_k r \log n \log(1/\epsilon))$; total time complexity: $\mathcal{O}(ndr \log n \log(1/\epsilon))$.

Finally, consider the setting when subspace changes a "little" at each time, but has significant changes at times $t = t_j$. We can interpret this as a piecewise constant plus some small "noise". Concretely, for $t \in [t_j, t_{j+1})$, let $\tilde{\boldsymbol{\ell}}_i = \boldsymbol{P}_{i,k,(t)} \tilde{\boldsymbol{a}}_i$. Assume that $\tilde{\boldsymbol{a}}_i$'s are zero-mean, i.i.d, and bounded with diagonal covariance $\tilde{\boldsymbol{\Lambda}}$. Let $\tilde{\lambda}^+$ and $\tilde{f}$ denote is max. eigenvalue and condition number respectively. Define $\boldsymbol{P}_j$ as the top-r left singular vectors of $\tilde{\boldsymbol{L}}_{(t)} = [\tilde{\boldsymbol{L}}_{1,(t)}, \cdots, \tilde{\boldsymbol{L}}_{K,(t)}]$; let $\boldsymbol{a}_i := \boldsymbol{P}_j{}' \tilde{\boldsymbol{\ell}}_i$, $\boldsymbol{\ell}_i := \boldsymbol{P}_j \boldsymbol{a}_i$ and $\boldsymbol{v}_i := \tilde{\boldsymbol{\ell}}_i - \boldsymbol{\ell}_i = (\boldsymbol{I} - \boldsymbol{P}_j \boldsymbol{P}_j^T) \tilde{\boldsymbol{\ell}}_i := \boldsymbol{P}_{j,\perp} \tilde{\boldsymbol{\ell}}_i$.

Corollary 3.4 (Subspace change at each time)**.** *Under the conditions of Theorem 3.3, with the above subspace change model, as long as for all $t \in [t_j, t_{j+1})$, $\operatorname{dist}(\boldsymbol{P}_j, \boldsymbol{P}_{i,k,(t)}) \leq 0.1\epsilon^2 / \tilde{f}^2$, all the conclusions of the above theorem hold with $\boldsymbol{P}_j$, $\boldsymbol{\ell}_i$ and $\boldsymbol{v}_i$ as defined above.*

Discussion. As there are no other guarantees for our setting, we instead provide a brief comparison with centralized ST-miss and LRMC work. A few algorithms for ST-miss include [16, 17, 30] but these do not come with complete guarantees or cannot provably detect subspace changes. A key advantage of our approach is that we are able to detect subspace changes in near real-time. In applications such as dynamic social network connectivity pattern detection, this is the most important information needed. In comparison to the only provable result of ST-miss, [18], our algorithm is online (and not mini-batch), and it respects the federated over-the-air constraints. This requires novel changes to the algorithm design; in particular for the change detection step. Moreover, our guarantee uses a significantly weaker version of statistical right incoherence than [18], which assumes $\max_i \|(\boldsymbol{a}_i)\|_\infty^2 \leq \mu \lambda_{\max}(\boldsymbol{\Lambda})$. We also provide the *first provable* result for setting subspace changing at each time. We show competitive experimental comparison (Fig. 3, 2). In comparison with LRMC, (i) our result does not require any probabilistic model on the set of observed entries, however the disadvantage is it needs many more observed entries in the initial few of time instants than LRMC methods. The probabilistic model is often an impractical requirement in many applications such as recommendation system design; (ii) Speed-wise, our algorithm (upto constants) is equal to that of computing a rank-r vanilla SVD on the data. Thus, it is slower than the fastest non-convex LRMC approach [33] ($O(nr^3 \log^2 n \log^2(1/\epsilon))$), but much faster than the convex approaches ($O(nd^2/\epsilon)$).

Figure 2: Visual performance in background recovery. The result of Fed-STmiss is comparable to that of NORST, but outperforms PETRELS (has noticeable specularities) and GROUSE (outputs static background). Time taken (in milliseconds) per frame is displayed below the algorithm label.

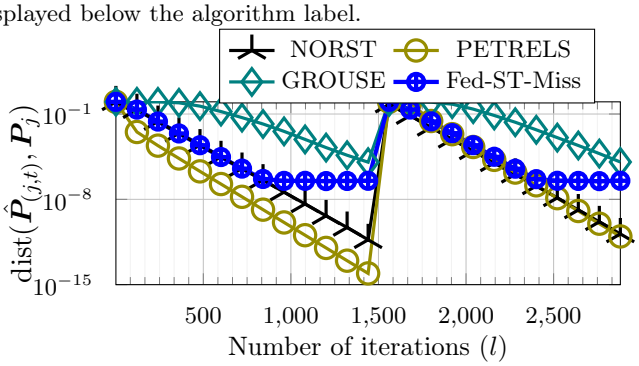

Figure 3: Comparison of ST-Miss Algorithms. Fed-ST-Miss is comparable to the state-of-the-art centralized ST-Miss methods.

4 Numerical Experiments

Experiments are performed on a Desktop Computer with Intel® Xeon 8-core CPU with 32GB RAM and the results are averaged over 50 independent trials. The codes are provided at `https://github.com/praneethmurthy/distributed-pca`.

Federated Power Method. Consider FedPM. We first generate $\boldsymbol{X} = \boldsymbol{U}\Lambda\boldsymbol{V}^T + \boldsymbol{U}_\perp\Lambda_\perp\boldsymbol{V}_\perp^T$ with $\boldsymbol{U}^* = [\boldsymbol{U}, \boldsymbol{U}_\perp]$, $\boldsymbol{V}^* = [\boldsymbol{V}, \boldsymbol{V}_\perp]$ being orthonormal matrices of appropriate dimensions. We then set $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{X}^T$ and the goal is to estimate the span of the $n \times r$ dimensional matrix, $\boldsymbol{U}$. We choose $n = 1000$ and $r = 30$. We consider two settings where $\Lambda = 1.1\boldsymbol{I}$, $\Lambda_\perp = \boldsymbol{I}$ so that $R = 0.91$; and $\Lambda = 3.3\boldsymbol{I}$, $\Lambda_\perp = \boldsymbol{I}$ so that $R = 0.33$. At each iteration we generate channel noise as i.i.d. $\mathcal{N}(0, \sigma_c^2)$. We verify the claims of Theorem 2.1 and (i) show that choosing a larger value of τ considerably increases robustness to noise. We set $R = 0.91$, and consider $\tau = 1, 10$ and $\sigma_c = 10^{-4}, 10^{-4}$. See from Fig. 1(a) that increasing τ has a similar effect as that of reducing σ_c (the $\tau = 10, \sigma_c = 10^{-8}$ plot overlaps with $\tau = 1, \sigma_c = 10^{-8}$); and (ii) in Fig. 1(b) we show that choosing a smaller value of R speeds up convergence, and also increases noise robustness. Here we use $\sigma_c = 10^{-8}$ and consider two eigengaps, $R = \{0.91, 0.30\}$.

Federated ST-Miss.

Next we illustrate the performance of Algorithm 2. We generate the data as done in most subspace

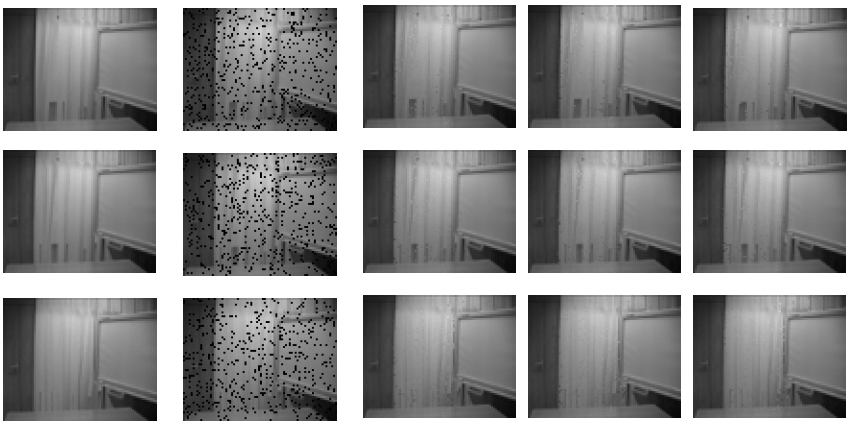

Figure 4: Understanding the effect of channel noise for Background Subtraction.

tracking literature. We set $\boldsymbol{\ell}_i = \boldsymbol{P}_j \boldsymbol{a}_i$ with *one subspace change* at $t = t_1 = 1500$. We generate $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ by orthonormalizing a standard Gaussian matrix of size $n \times r$ with $n = 1000$ and $r = 30$. The entries of $\boldsymbol{a}_i$ are chosen i.i.d. from a uniform distribution, $\mathcal{U}[-1, 1]$. Thus, all assumptions of Theorem 3.3 as satisfied. We do not generate modeling noise in this experiment ($\boldsymbol{v}_i = 0$). For the implementation of Algorithm 2, we additionally simulate channel noise, $\boldsymbol{W}_l \sim \mathcal{N}(0, \sigma_c^2)$ with $\sigma_c = 10^{-6}$. We compare the proposed method with 3 state-of-the-art (centralized) methods for ST-Miss: NORST [18], PETRELS [15], and GROUSE [16].

We implemented Algorithm 2 with $\alpha = Cf^2 r \log n = 60$, $\omega_{evals} = 2\epsilon^2\lambda^+ = 7 \times 10^{-4}$, $T = 25$. Notice that our algorithm converges to the noise level (channel noise) whereas NORST and PETRELS are able to track the subspace to approximately 10^{-12}. GROUSE has a slower convergence (since this is a first order method) and thus it also tracks to only 10^{-6}. As can be seen from Fig. 3, all algorithms are able to satisfactorily track the underlying subspace while PETRELS has the best performance. Despite the addition of channel noise, our method is comparable to GROUSE.

Background Recovery. We also tested Algorithm 2 on several datasets for the background recovery and considered two models for missing data. We show the results for the moving object model [32] in Fig. 2. For Fed-STmiss we added i.i.d. Gaussian channel noise with $\sigma_c = 10^{-6}$. We implemented Fed-STmiss with $\alpha = 60$, $T = 3$, $L = 500$, $L_{\text{det}} = 10$. For all algorithms we set $r = 30$. Notice that Fed-STmiss is able to visually match the performance obtained by NORST and is significantly better than the output produced by PETRELS and GROUSE. For NORST, PETRELS and GROUSE, we use the default parameter settings. For PETRELS, we use `max_cycles`= 10 since with the default setting of `max_cycles`= 1, the algorithm always failed. We illustrate the effect of increasing channel noise in Fig. 4. Notice that since the image data ranges from $0 - 255$, even with iteration noise chosen as $\mathcal{N}(0, 1)$, our method is able to satisfactorily recover the background.

We test the proposed method on the following data sets:

`Meeting Room (MR) dataset`: The meeting room sequence is set of 1964 images of resolution 64×80. The first 1755 frames consists of outlier-free data. Henceforth, we consider only the first 1755 frames since none can deal with sparse outliers. We show the results at $t = 110, 200, 500$.

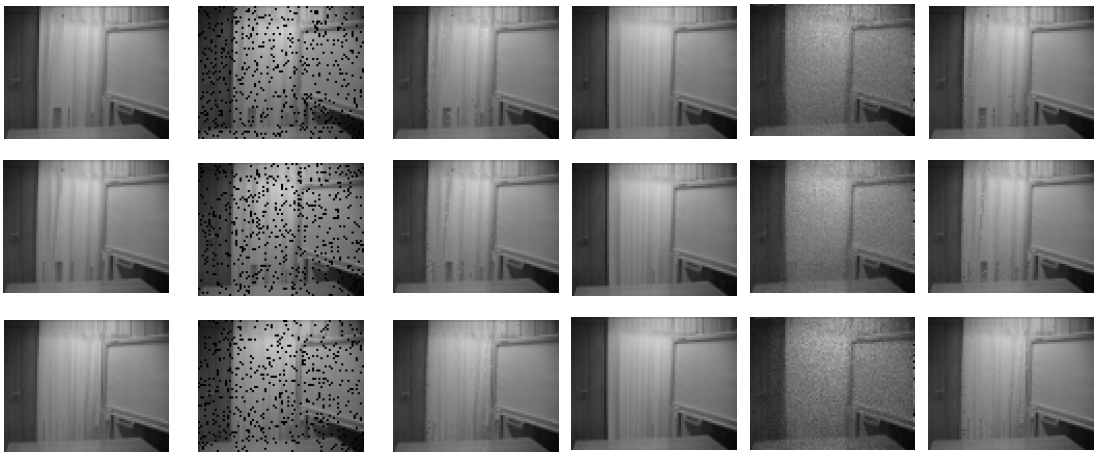

Figure 5: Comparison of visual performance in Foreground Background separation.

Figure 6: Comparison of visual performance for Background separation in Lobby dataset.

`Lobby (LB) dataset`: This dataset contains 1555 images of resolution 128×160. The first 341 frames are outlier free which we use for all algorithms (since none can deal with sparse outliers). We show the results at $t = 110, 200, 300$. As can be seen from Fig. 6, this is an easy dataset and all algorithms work well.

`Switch Light (SL) dataset`: This dataset contains 2100 images of resolution 120×160. The first 770 frames are outlier free. This is a challenging sequence because there are drastic changes in the subspace. As can be seen from Fig. 7, the output of PETRELS contains some artifacts, and GROUSE outputs a static background (notice the computer monitor) but the proposed method is comparable to NORST which has *no channel noise*.

Figure 7: Comparison of visual performance for Background separation in Switch Light dataset.

References

[1] Jakub Konecny, H Brendan McMahan, Daniel Ramage, and Peter Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.

[2] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.

[3] Shiqiang Wang, Tiffany Tuor, Theodoros Salonidis, Kin K Leung, Christian Makaya, Ting He, and Kevin Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, 2019.

[4] Mohammad Mohammadi Amiri and Deniz Gündüz, "Federated learning over wireless fading channels," *arXiv preprint arXiv:1907.09769*, 2019.

[5] Mohammad Mohammadi Amiri and Deniz Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," in *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2019, pp. 1432–1436.

[6] Kai Yang, Tao Jiang, Yuanming Shi, and Zhi Ding, "Federated learning via over-the-air computation," *IEEE Transactions on Wireless Communications*, 2020.

[7] David Tse and Pramod Viswanath, *Fundamentals of wireless communication*, Cambridge university press, 2005.

[8] E. J. Candes and B. Recht, "Exact matrix completion via convex optimization," *Found. of Comput. Math*, , no. 9, pp. 717–772, 2008.

[9] Ali Zare, Alp Ozdemir, Mark A Iwen, and Selin Aviyente, "Extension of pca to higher order data structures: An introduction to tensors, tensor decompositions, and tensor pca," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1341–1358, 2018.

[10] Sissi Xiaoxiao Wu, Hoi-To Wai, Lin Li, and Anna Scaglione, "A review of distributed algorithms for principal component analysis," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1321–1340, 2018.

[11] Gene H Golub and Charles F Van Loan, "Matrix computations," *The Johns Hopkins University Press, Baltimore, USA*, 1989.

[12] Yingyu Liang, Maria-Florina F Balcan, Vandana Kanchanapally, and David Woodruff, "Improved distributed principal component analysis," in *NIPS*, 2014, pp. 3113–3121.

[13] Moritz Hardt and Eric Price, "The noisy power method: A meta algorithm with applications," in *Advances in Neural Information Processing Systems*, 2014, pp. 2861–2869.

[14] Maria-Florina Balcan, Simon Shaolei Du, Yining Wang, and Adams Wei Yu, "An improved gap-dependency analysis of the noisy power method," in *Conference on Learning Theory*, 2016, pp. 284–309.

[15] Y. Chi, Y. C. Eldar, and R. Calderbank, "Petrels: Parallel subspace estimation and tracking by recursive least squares from partial observations," *IEEE Transactions on Signal Processing*, December 2013.

[16] D. Zhang and L. Balzano, "Global convergence of a grassmannian gradient descent algorithm for subspace estimation," in *AISTATS*, 2016.

[17] A. Gonen, D. Rosenbaum, Y. C. Eldar, and S. Shalev-Shwartz, "Subspace learning with partial information," *Journal of Machine Learning Research*, vol. 17, no. 52, pp. 1–21, 2016.

[18] P. Narayanamurthy, V. Daneshpajooh, and N. Vaswani, "Provable subspace tracking from missing data and matrix completion," *IEEE Transactions on Signal Processing*, pp. 4245–4260, 2019.

[19] N. Vaswani and P. Narayanamurthy, "Pca in sparse data-dependent noise," in *ISIT*, 2018, pp. 641–645.

[20] D. Gunduz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, "Machine learning in the air," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2184–2199, 2019.

[21] Jakub Konecny, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[22] Andreas Grammenos, Rodrigo Mendoza-Smith, Cecilia Mascolo, and Jon Crowcroft, "Federated pca with adaptive rank estimation," *arXiv preprint arXiv:1907.08059*, 2019.

[23] Lester Mackey, Ameet Talwalkar, and Michael I Jordan, "Distributed matrix completion and robust factorization," *The Journal of Machine Learning Research*, vol. 16, no. 1, pp. 913–960, 2015.

[24] X. He, Q. Ling, and T. Chen, "Byzantine-robust stochastic gradient descent for distributed low-rank matrix completion," in *2019 IEEE Data Science Workshop (DSW)*, 2019, pp. 322–326.

[25] Christina Teflioudi, Faraz Makari, and Rainer Gemulla, "Distributed matrix completion," in *2012 ieee 12th international conference on data mining*. IEEE, 2012, pp. 655–664.

[26] Dan Alistarh, Zeyuan Allen-Zhu, and Jerry Li, "Byzantine stochastic gradient descent," in *Advances in Neural Information Processing Systems*, 2018, pp. 4613–4623.

[27] Cong Xie, Sanmi Koyejo, and Indranil Gupta, "Fall of empires: Breaking byzantine-tolerant sgd by inner product manipulation," *arXiv preprint arXiv:1903.03936*, 2019.

[28] B. Recht, "A simpler approach to matrix completion," *Journal of Machine Learning Research*, vol. 12, no. Dec, pp. 3413–3430, 2011.

[29] P. Netrapalli, P. Jain, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *STOC*, 2013.

[30] Chuang Wang, Yonina C Eldar, and Yue M Lu, "Subspace estimation from incomplete observations: A high-dimensional analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1240–1252, 2018.

[31] Laura Balzano, Yuejie Chi, and Yue M Lu, "Streaming pca and subspace tracking: The missing data case," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1293–1310, 2018.

[32] P. Narayanamurthy and N. Vaswani, "Nearly optimal robust subspace tracking," in *International Conference on Machine Learning*, 2018, pp. 3701–3709.

[33] Y. Cherapanamjeri, K. Gupta, and P. Jain, "Nearly-optimal robust matrix completion," *ICML*, 2016.

[34] C. Davis and W. M. Kahan, "The rotation of eigenvectors by a perturbation. iii," *SIAM J. Numer. Anal.*, vol. 7, pp. 1–46, Mar. 1970.

[35] J. A. Tropp, "User-friendly tail bounds for sums of random matrices," *Found. Comput. Math.*, vol. 12, no. 4, 2012.

[36] Roger A Horn and Charles R Johnson, *Matrix analysis*, Cambridge university press, 2012.

[37] Roman Vershynin, *High-dimensional probability: An introduction with applications in data science*, vol. 47, Cambridge university press, 2018.

[38] Mark Rudelson and Roman Vershynin, "Smallest singular value of a random rectangular matrix," *Communications on Pure and Applied Mathematics*, vol. 62, no. 12, pp. 1707–1739, 2009.

Supplementary Material

In Appendix A, we provide the the complete algorithm for Federated Over The Air Dynamic Subspace Learning (Algorithm 2), the key idea of change detection, and the proofs for Theorem A.1 (Static Subspace, noise-free ST-miss), Corollary A.2 (Static subspace, noisy ST-miss) which essentially provides the proof of the main result, Theorem 3.3. Finally, we provide the proof of the setting where the subspace is allowed to change a little at each time (Corollary 3.4).

In Appendix B we provide the proof for the convergence analysis of FedPM (Algorithm 1), i.e., we prove Theorem 2.1. In Appendix C, we state and prove a result to analyze the problem of PCA in Sparse and Data-Dependent Noise (PCA-SDDN), which is a critical tool in the convergence analysis of the Federated STMiss problem.

A Federated Over-the-Air Subspace Tracking with Missing Entries

For simplicity in proof consider the setting of a static subspace, with missing entries. Furthermore, first assume that there is no modeling error.

Theorem A.1 (Federated Subspace Tracking: fixed subspace and no modeling error)**.** *Consider Algorithm 2 with a fixed subspace $\boldsymbol{P}$. Pick a final desired error level $\epsilon > 0$. Set $T := C\log(1/\epsilon)$, set $L = C\log(n/\epsilon_t)$ where $\epsilon_t := \max(\epsilon, 0.01\cdot(0.3)^{t-1})$ and $\tau = 1$ in the FedPM algorithm. Assume that the following hold: $\alpha \in \Omega(r\log n)$*

1. **Incoherence:** *$\boldsymbol{P}$ satisfies μ-incoherence, (3) with μ constant, and the $\boldsymbol{a}_i$'s satisfy statistical μ-right incoherence (Definition 3.1);*
2. **Missing Entries:** *max-miss-frac-col $\in O(1/\mu r)$, max-miss-frac-row $\in O(1)$;*
3. **Channel Noise:** *the channel noise seen by each FedPM iteration is mutually independent at all times, isotropic, and zero mean Gaussian with variance $\sigma_c^2 \le \epsilon_t\lambda^-/\sqrt{n}$.*

then, with probability at least $1 - 10dn^{-10} - c\gamma$,

$$\mathrm{dist}(\hat{\boldsymbol{P}}_t, \boldsymbol{P}) \le \begin{cases} \max(0.01\cdot(0.3)^{t-1}, \varepsilon) & \text{if } t < T \\ \epsilon & \text{if } t = T. \end{cases}$$

Also, $\|\hat{\boldsymbol{\ell}}_{i,(t)} - \boldsymbol{\ell}_{i,(t)}\| \le 1.2\cdot\mathrm{dist}(\hat{\boldsymbol{P}}_t, \boldsymbol{P})\|\boldsymbol{\ell}_{i,(t)}\|$ for all i and t (these are only recovered locally at each node).

Time complexity at node k: $\mathcal{O}(n\alpha_k r\log n\log(1/\epsilon))$.

Next we have the following result for non-zero modeling error.

Corollary A.2 (nonzero modeling error)**.** *Assume that the modeling error, $\boldsymbol{v}_{i,(t)}$, is bounded, i.i.d., is independent of the true low rank matrix, and is zero mean. Let $\lambda_v^+ := \|\mathbb{E}[\boldsymbol{v}_{i,(t)}\boldsymbol{v}_{i,(t)}^T]\|$. If $\max_{i,t}\|\boldsymbol{v}_{i,(t)}\|^2 \le Cr\lambda_v^+$ for a numerical constant C, and $\sqrt{\lambda_v^+/\lambda^-} \le 0.2$, then all assumptions of Theorem A.1 hold with ε replaced by $c\sqrt{\lambda_v^+/\lambda^-}$.*

Here we provide the proof of the above resuls. In Appendix A.2, we explain the subspace change detection idea in detail and explain why it works, give the stepwise algorithm, and then prove the key new lemma needed for detecting subspace change.

To keep notation simple, we will use $\boldsymbol{y}_i$ to denote $\boldsymbol{y}_{i,(t)}$ (since the dependence on t is implicit).

A.1 Proof of Theorem A.1 and Corollary A.2

Throughout this section, we denote the FedPM algorithm output by $\tilde{\boldsymbol{P}}_{(t)}$. Recall from Theorem A.1 (and Corollary A.2) that $\epsilon_t = \max(0.01(0.3)^t, \epsilon_v)$ where $\epsilon_v = c\sqrt{\lambda_v^+/\lambda^-}$ and we assume that $\epsilon_v \leq 0.2$ and thus, at all times t, it follows that $\epsilon_t \leq 0.2$. Additionally, in Theorem A.1 we stated that max-miss-frac-row $= \mathcal{O}(1)$ and max-miss-frac-col $= \mathcal{O}(1/\mu r)$ to keep the statement simple but in the supplement, we will use max-miss-frac-row $\leq (0.01/f)^2$ and max-miss-frac-col $\leq 0.1/\mu r$. Again, $f = \lambda^+/\lambda^-$ is the condition number and we treated f, μ, the incoherence parameter as constants. There are the following two parts in the proof:

1. First, we need to show that $\tilde{\boldsymbol{P}}_{(t)}$ is *close* to $\hat{\boldsymbol{P}}_{(t)}$ where $\hat{\boldsymbol{P}}_{(t)}$, by definition is the top r left singular vectors of $\hat{\boldsymbol{L}}_{(t)}$. In particular, in the t-th subpsace update step, we show that $\text{dist}(\tilde{\boldsymbol{P}}_{(t)}, \hat{\boldsymbol{P}}_{(t)}) \leq \epsilon_t/2$.
2. Next, we use the above result, and a result for Principal Components Analysis in Sparse, Data-Dependent Noise (PCA SDDN) to show that $\text{dist}(\tilde{\boldsymbol{P}}_{(t)}, \boldsymbol{P}) \leq \text{dist}(\tilde{\boldsymbol{P}}_{(t)}, \hat{\boldsymbol{P}}_{(t)}) + \text{dist}(\hat{\boldsymbol{P}}_{(t)}, \boldsymbol{P}) \leq \epsilon_t$.

Key Results Needed. The above two steps rely on the following key results.

The lemma below is a restatement of Theorem 2.1 with $\tau = 1$, and using random initialization (Item 3 of Theorem 2.1).

Lemma A.3 (FedPM with $\tau = 1$ and random initialization)**.** *Consider Algorithm 1 with $\tau = 1$ and with initial subspace estimate, $\hat{\boldsymbol{U}}_0 \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$. If, at each iteration, the channel noise $\boldsymbol{W}_l \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_c^2)$ with $\sigma_c < \epsilon_{\text{PM}} \lambda_r(\boldsymbol{Y}\boldsymbol{Y}^T)/(5\sqrt{n})$ and if $R = \lambda_{r+1}(\boldsymbol{Y}\boldsymbol{Y}^T)/\lambda_r(\boldsymbol{Y}\boldsymbol{Y}^T) < 0.99$, then, after*

$$L = \Omega\left(\frac{1}{\log(1/R)}\left(\log\frac{nr}{\epsilon_{\text{PM}}}\right)\right)$$

iterations, with probability at least $1 - L\exp(-cr) - (c\gamma)$, $\text{dist}(\boldsymbol{U}, \hat{\boldsymbol{Q}}_L) \leq \epsilon_{\text{PM}}$.

The following result is used to analyze the PCA-SDDN problem in a centralized setting. It is a significant generalization of the result proved in [19] where this problem was first studied: the result below holds under a weaker statistical right incoherence assumption than what was needed in [19]. We only require a bound on $\|\boldsymbol{a}_i\|$ and not on each entry of it. The proof given in Appendix C uses the Davis-Kahan $\sin\theta$ theorem [34] to bound the subspace distance between the column spans of $\hat{\boldsymbol{P}}$ and of $\boldsymbol{P}$, followed by using the Matrix Bernstein inequality [35] to obtain high probability bounds on each of the terms in the Davis-Kahan bound.

Lemma A.4 (PCA-SDDN)**.** *For $i = 1, \cdots, \alpha$, suppose that $\boldsymbol{y}_i = \boldsymbol{\ell}_i + \boldsymbol{e}_i + \boldsymbol{v}_i$ with $\boldsymbol{e}_i = \boldsymbol{I}_{\mathcal{M}_i}\boldsymbol{M}_{2,i}\boldsymbol{\ell}_i$ being sparse, data-dependent noise with support $\mathcal{M}_i$; $\boldsymbol{\ell}_i = \boldsymbol{P}\boldsymbol{a}_i$ where $\boldsymbol{P}$ is a $n \times r$ basis matrix which satisfies μ left incoherence and and $\boldsymbol{a}_i$'s satisfy the μ-statistical right-incoherence assumption given in Definition 3.1; and $\boldsymbol{v}_i$ is small bounded noise with $\lambda_v^+ := \|\mathbb{E}[\boldsymbol{v}_i\boldsymbol{v}_i^T]\|$ be the noise power and let $\max_i \|\boldsymbol{v}_i\|^2 \leq Cr\lambda_v^+$.*

Let $\hat{\boldsymbol{P}}$ be the matrix of top r eigenvectors of $\frac{1}{\alpha}\sum_i \boldsymbol{y}_i\boldsymbol{y}_i^T$. Assume that $\max_i \|\boldsymbol{M}_{2,i}\boldsymbol{P}\| \leq q$ for a $q \leq 3$ and that the fraction of non-zeros in any row of the matrix $[\boldsymbol{e}_1, \cdots, \boldsymbol{e}_\alpha]$ is bounded by b. Pick an $\epsilon_{\text{SE}} > 0$. If

$$6\sqrt{b}qf + \lambda_v^+/\lambda^- < 0.4\epsilon_{\text{SE}}, \tag{4}$$

and if $\alpha \geq \alpha^*$ *where*

$$\alpha^* := C \max\left(\frac{q^2 f^2}{\epsilon_{\text{SE}}^2} r \log n, \frac{\frac{\lambda_v^+}{\lambda^-} f}{\epsilon_{\text{SE}}^2} r \log n\right), \tag{5}$$

then, w.p. at least $1 - 10n^{-10}$, $\text{dist}(\hat{\boldsymbol{P}}, \boldsymbol{P}) \leq \epsilon_{\text{SE}}$.

Furthermore, as long as $\alpha \geq \alpha^*$, *we have that with probability at least* $1 - 10n^{-10}$,

$$\begin{aligned}
\|\text{perturb}\| &:= \left\| \frac{1}{\alpha} \sum_i (\boldsymbol{\ell}_i \boldsymbol{e}_i^T + \boldsymbol{e}_i \boldsymbol{\ell}_i^T + \boldsymbol{e}_i \boldsymbol{e}_i^T + \boldsymbol{v}_i \boldsymbol{v}_i^T + \boldsymbol{\ell}_i \boldsymbol{v}_i^T + \boldsymbol{v}_i \boldsymbol{\ell}_i^T + \boldsymbol{v}_i \boldsymbol{e}_i^T + \boldsymbol{e}_i \boldsymbol{v}_i^T) \right\| \\
&\leq \left\| \frac{1}{\alpha} \sum_i \boldsymbol{e}_i \boldsymbol{e}_i^T \right\| + 2 \left\| \frac{1}{\alpha} \sum_i \boldsymbol{\ell}_i \boldsymbol{e}_i^T \right\| + 2 \left\| \frac{1}{\alpha} \sum_i \boldsymbol{\ell}_i \boldsymbol{v}_i^T \right\| + 2 \left\| \frac{1}{\alpha} \sum_i \boldsymbol{v}_i \boldsymbol{e}_i^T \right\| + \left\| \frac{1}{\alpha} \sum_i \boldsymbol{v}_i \boldsymbol{v}_i^T \right\| \\
&\leq \left(6.6\sqrt{b} q f + 5.5 \frac{\lambda_v^+}{\lambda^-} \right) \lambda^-
\end{aligned}$$

and

$$\lambda_r \left(\frac{1}{\alpha} \sum_i \boldsymbol{\ell}_i \boldsymbol{\ell}_i^T \right) \geq 0.99 \lambda^-.$$

Proof. The proof follows from Theorem C.1 by setting $\boldsymbol{M}_{1,i} = \boldsymbol{I}_{\mathcal{M}_i}$. Thus, $b = \|\frac{1}{\alpha} \sum_i \boldsymbol{I}_{\mathcal{M}_i} \boldsymbol{I}_{\mathcal{M}_i}^T\|$ is equal to the maximum fraction of missing entries in any row of $[\boldsymbol{e}_1, \cdots, \boldsymbol{e}_\alpha]$. ⊠

We first use Lemma A.3 with $\boldsymbol{Y} = \frac{1}{\sqrt{\alpha}} \hat{\boldsymbol{L}}_{(t)}$ to bound $\text{dist}(\tilde{\boldsymbol{P}}_{(t)}, \hat{\boldsymbol{P}}_{(t)})$. To apply the result, we need to first lower bound its r-th eigenvalue and upper bound its $r+1$-th eigenvalue. Recall that $\hat{\boldsymbol{\ell}}_i = \boldsymbol{\ell}_i + \boldsymbol{e}_i + \tilde{\boldsymbol{v}}_i$ with

$$\boldsymbol{e}_i = \boldsymbol{I}_{\mathcal{M}_i} \left(\boldsymbol{\Psi}_{\mathcal{M}_i}^T \boldsymbol{\Psi}_{\mathcal{M}_i} \right)^{-1} \boldsymbol{I}_{\mathcal{M}_i}^T \boldsymbol{\Psi} \boldsymbol{\ell}_i, \tag{6}$$

as shown in (3) (noise-free case) and $\boldsymbol{\Psi} = \boldsymbol{I} - \hat{\boldsymbol{P}}_{(t-1)} \hat{\boldsymbol{P}}_{(t-1)}^T$. Additionally,

$$\tilde{\boldsymbol{v}}_i = \boldsymbol{I}_{\mathcal{M}_i} \left(\boldsymbol{\Psi}_{\mathcal{M}_i}^T \boldsymbol{\Psi}_{\mathcal{M}_i} \right)^{-1} \boldsymbol{I}_{\mathcal{M}_i}^T \boldsymbol{\Psi} \boldsymbol{v}_i$$

We will use the following simple facts in various places in our proof.

Fact A.5. *Let* $\boldsymbol{P}$, $\hat{\boldsymbol{P}}$ *be two basis matrices of dimension* $n \times r$. *Let* $\boldsymbol{\Psi} = \boldsymbol{I} - \hat{\boldsymbol{P}} \hat{\boldsymbol{P}}^T$. *Then, for any set* $\mathcal{M}$

1. $$\|\boldsymbol{I}_{\mathcal{M}}^T \hat{\boldsymbol{P}}\| \leq \|\boldsymbol{I}_{\mathcal{M}}^T (\boldsymbol{I} - \boldsymbol{P}\boldsymbol{P}^T) \hat{\boldsymbol{P}}\| + \|\boldsymbol{I}_{\mathcal{M}}^T \boldsymbol{P}\boldsymbol{P}^T \hat{\boldsymbol{P}}\| \leq \text{dist}(\hat{\boldsymbol{P}}, \boldsymbol{P}) + \|\boldsymbol{I}_{\mathcal{M}}^T \boldsymbol{P}\|$$

2. $$\begin{aligned}
\|(\boldsymbol{\Psi}_{\mathcal{M}}^T \boldsymbol{\Psi}_{\mathcal{M}})^{-1}\| &= \|(\boldsymbol{I}_{\mathcal{M}}^T (\boldsymbol{I} - \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}^T) \boldsymbol{I}_{\mathcal{M}})^{-1}\| = \|(\boldsymbol{I} - \boldsymbol{I}_{\mathcal{M}}^T \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}^T \boldsymbol{I}_{\mathcal{M}})^{-1}\| \\
&= \frac{1}{\lambda_{\min}(\boldsymbol{I} - \boldsymbol{I}_{\mathcal{M}}^T \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}^T \boldsymbol{I}_{\mathcal{M}})} = \frac{1}{1 - \lambda_{\max}(\boldsymbol{I}_{\mathcal{M}}^T \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}^T \boldsymbol{I}_{\mathcal{M}})} \\
&= \frac{1}{1 - \|\boldsymbol{I}_{\mathcal{M}}^T \hat{\boldsymbol{P}}\|^2}
\end{aligned}$$

3. At all times t, since we assumeed that $\epsilon_t \le 0.2$, and using μ-incoherence and the bound on max-miss-frac-col, we have that for $\boldsymbol{\Psi} = \boldsymbol{I} - \hat{\boldsymbol{P}}_{(t-1)}\hat{\boldsymbol{P}}_{(t-1)}^T$,

$$\begin{aligned}\|(\boldsymbol{\Psi}_{\mathcal{M}_i}^T\boldsymbol{\Psi}_{\mathcal{M}_i})^{-1}\| &\le \frac{1}{1-\|\boldsymbol{I}_{\mathcal{M}_i}^T\hat{\boldsymbol{P}}_{(t-1)}\|^2} \le \frac{1}{1-2\|\boldsymbol{I}_{\mathcal{M}_i}^T\boldsymbol{P}\|^2 - 2\text{dist}^2(\hat{\boldsymbol{P}}_{(t-1)},\boldsymbol{P})} \\ &\le \frac{1}{1-2\cdot(0.1)^2-2\cdot(0.2)^2} \le 1.2\end{aligned}$$

Thus, using Fact A.5, notice that for all i, $\|\tilde{\boldsymbol{v}}_i\| \le \|\left(\boldsymbol{\Psi}_{\mathcal{M}_i}^T\boldsymbol{\Psi}_{\mathcal{M}_i}\right)^{-1}\|\|\boldsymbol{\Psi}\|\|\boldsymbol{v}_i\| \le 1.2\|\boldsymbol{v}_i\|$. Similarly, $\|\mathbb{E}[\tilde{\boldsymbol{v}}_i\tilde{\boldsymbol{v}}_i^T]\| \le 1.44\|\mathbb{E}[\boldsymbol{v}_i\boldsymbol{v}_i^T]\| \le 1.44\lambda_v^+$.

We now bound the eigen-ratio for the matrix $\hat{\boldsymbol{L}}_{(t)}\hat{\boldsymbol{L}}_{(t)}^T/\alpha$ using Weyl's inequality, and Lemma A.4. In the notation of Lemma A.4, $\boldsymbol{y}_i \equiv \hat{\boldsymbol{\ell}}_i$, $\boldsymbol{e}_i \equiv \boldsymbol{e}_i$, $\boldsymbol{v}_i = \tilde{\boldsymbol{v}}_i$, $\mathcal{M}_i \equiv \mathcal{M}_i$ (recall that this is the index set of missing entries), $\boldsymbol{\ell}_i \equiv \boldsymbol{\ell}_i$, $\hat{\boldsymbol{P}} = \hat{\boldsymbol{P}}_{(t)}$, $\boldsymbol{P} = \boldsymbol{P}$, and $\boldsymbol{M}_{2,i} = -\left(\boldsymbol{\Psi}_{\mathcal{M}_i}^T\boldsymbol{\Psi}_{\mathcal{M}_i}\right)^{-1}\boldsymbol{\Psi}_{\mathcal{M}_i}^T$ with $\boldsymbol{\Psi} = \boldsymbol{I}$ for $t = 1$ and $\boldsymbol{\Psi} = \boldsymbol{I} - \hat{\boldsymbol{P}}_{t-1}\hat{\boldsymbol{P}}_{t-1}^T$ for $t > 1$. Thus $b = $ max-miss-frac-row $\le (0.01/f)^2$ and q is an upper bound on $\|\boldsymbol{M}_{2,i}\boldsymbol{P}\|$. For $t = 1$, using the μ-incoherence assumption and the bound on max-miss-frac-col, we get $\|\boldsymbol{M}_{2,i}\boldsymbol{P}\| = \|\boldsymbol{I}_{\mathcal{M}_i}^T\boldsymbol{P}\| \le |\mathcal{M}_i|\max_j\|\boldsymbol{I}_j^T\boldsymbol{P}\| \le$ max-miss-frac-col $\cdot\, \mu r/n \le 0.1 = q$. The approach for obtaining q for $t > 1$ is slightly different. Since $\boldsymbol{\Psi} = \boldsymbol{I} - \hat{\boldsymbol{P}}_{(t-1)}\hat{\boldsymbol{P}}_{(t-1)}^T$ we have that $\|\boldsymbol{\Psi}\boldsymbol{P}\| = \text{dist}(\hat{\boldsymbol{P}}_{(t-1)},\boldsymbol{P}) \le \epsilon_{t-1}$. Thus, using Fact A.5, we get that $\|\boldsymbol{M}_{2,i}\boldsymbol{P}\| \le 1.2\epsilon_{t-1} = q$. Thus,

$$\begin{aligned}\lambda_r\left(\frac{1}{\alpha}\hat{\boldsymbol{L}}_{(t)}\hat{\boldsymbol{L}}_{(t)}^T\right) &\ge \lambda_r\left(\frac{1}{\alpha}\boldsymbol{L}_{(t)}\boldsymbol{L}_{(t)}^T\right) + \lambda_{\min}(\text{perturb}) \ge \lambda_r\left(\frac{1}{\alpha}\boldsymbol{L}_{(t)}\boldsymbol{L}_{(t)}^T\right) - \|\text{perturb}\| \\ &\ge 0.99 - 6.6\sqrt{b}qf - 6\frac{\lambda_v^+}{\lambda^-} \ge (0.99 - 0.7\max(0.1,\epsilon_{t-1}))\lambda^-\end{aligned}$$

Similarly,

$$\lambda_{r+1}\left(\frac{1}{\alpha}\hat{\boldsymbol{L}}_{(t)}\hat{\boldsymbol{L}}_{(t)}\right) \le \lambda_{r+1}\left(\frac{1}{\alpha}\boldsymbol{L}_{(t)}\boldsymbol{L}_{(t)}^T\right) + \|\text{perturb}\| = \|\text{perturb}\| \le 0.6\max(0.1,\epsilon_{t-1})\lambda^-$$

Thus, $R = \lambda_{r+1}/\lambda_r \le 1/20$ for all $t \ge 1$ and this ratio becomes smaller since λ_{r+1} decreases with each subspace update step. Additionally, since the channel noise is bounded as assumed in Theorem A.2, Lemma A.3 can be applied with $\epsilon_{\text{PM}} = \epsilon_t/2$.

Finally, notice that in the first subspace update step, we need $L = (C/\log 20)\cdot\log(nr)$ iterations to obtain $\epsilon_{\text{PM}} = 0.1$ accuracy. This is because, we are randomly initializing FedPM, we incur the $\log(nr)$ factor. In the subsequent subspace update steps, we initialize FedPM with the estimate from the previous subspace update, $\tilde{\boldsymbol{P}}_{(t-1)}$, and since $\text{dist}(\tilde{\boldsymbol{P}}_{(t-1)},\boldsymbol{P}) \le \epsilon_{t-1}$, and we only need to ensure that $\text{dist}(\tilde{P}_{(t)},\hat{\boldsymbol{P}}_{(t)}) \le \epsilon_t/2 = (0.3/2)\epsilon_{t-1}$, the number of iterations required is a constant as described by Lemma A.3. More precisely, we need to perform just $L = (C/\log 20)\cdot(\log C_2)$ iterations.

We now prove the second part, i.e., we show that $\text{dist}(\hat{\boldsymbol{P}}_{(t)},\boldsymbol{P}) \le \epsilon_t/2$. This uses Lemma A.4 and the following simple facts.

In the application of Lemma A.4, we will analyze each interval separately. Consider the first α frames, $\hat{\boldsymbol{P}}_{(t-1)} = \boldsymbol{0}$ (zero initialization) and so, during this time, $\boldsymbol{\Psi} = \boldsymbol{I}$. Now we apply Lemma A.4 to the $\hat{\boldsymbol{\ell}}_i$'s. Recall that $\hat{\boldsymbol{\ell}}_i = \boldsymbol{\ell}_i + \boldsymbol{e}_i + \tilde{\boldsymbol{v}}_i$ with $\boldsymbol{e}_i$ satisfying (3) and, and it is thus, sparse, and data-dependent. In addition, $\tilde{\boldsymbol{v}}_i$ satisfies the conditions under the assumptions of Theorem A.2. In the notation of Lemma A.4, $\boldsymbol{y}_i \equiv \hat{\boldsymbol{\ell}}_i$, $\boldsymbol{e}_i \equiv \boldsymbol{e}_i$, $\boldsymbol{v}_i = \tilde{\boldsymbol{v}}_i$, $\mathcal{M}_i \equiv \mathcal{M}_i$ (recall that this is the

index set of missing entries), $\boldsymbol{\ell}_i \equiv \boldsymbol{\ell}_i$, $\hat{\boldsymbol{P}} = \hat{\boldsymbol{P}}_{(1)}$, $\boldsymbol{P} = \boldsymbol{P}$, and $\boldsymbol{M}_{2,i} = -\left(\boldsymbol{\Psi}_{\mathcal{M}_i}^T \boldsymbol{\Psi}_{\mathcal{M}_i}\right)^{-1} \boldsymbol{\Psi}_{\mathcal{M}_i}^T$ with $\boldsymbol{\Psi} = \boldsymbol{I}$. Thus, using the μ-incoherence assumption and the bound on max-miss-frac-col, we get $\|\boldsymbol{M}_{2,i}\boldsymbol{P}\| = \|\boldsymbol{I}_{\mathcal{M}_i}^T \boldsymbol{P}\| \leq |\mathcal{M}_i| \max_j \|\boldsymbol{I}_j^T \boldsymbol{P}\| \leq \text{max-miss-frac-col} \cdot \mu r/n \leq 0.1 = q_0 \equiv q$. Notice that $b \equiv \text{max-miss-frac-row} \leq 0.001/f^2$, and the assumption on $\boldsymbol{v}_i$ ensures that we can apply Lemma A.4 with $\varepsilon_{\text{SE}} = 0.5 \max(q/4, \epsilon_v)$. Under the conditions of Theorem A.1, $\alpha = Cf^2 r \log n$ satisfies $\alpha \geq \alpha^*$ since the assumption on $\boldsymbol{v}_i$'s ensures that the two terms in the α^* expression are equal upto numerical constants. Furthermore, because max-miss-frac-row $= 0.001/f^2$, and the assumption on $\boldsymbol{v}_i$ ensures that (4) holds[3]. Thus, we conclude that $\text{dist}(\hat{\boldsymbol{P}}_{(1)}, \boldsymbol{P}) \leq \varepsilon_{\text{SE}} = 0.5 \max(\epsilon_v, q/4) := q_1 := \epsilon_1/2$ whp.

In the subsequent subspace update steps, we use almost the same approach as done in the first α frames, $t = 1$. The difference is in how we bound $\|\boldsymbol{I}_{\mathcal{M}_i}^T \boldsymbol{\Psi}\boldsymbol{P}\|$. Recall that t-th subspace update step, $\boldsymbol{\Psi} = \boldsymbol{I} - \hat{\boldsymbol{P}}_{(t-1)}\hat{\boldsymbol{P}}_{(t-1)}^T$. We know that $\|\boldsymbol{\Psi}\boldsymbol{P}\| = \text{dist}(\hat{\boldsymbol{P}}_{(t-1)}, \boldsymbol{P}) \leq \epsilon_{t-1}$ now. Thus, $\|\boldsymbol{M}_{2,i}\boldsymbol{P}\| \leq 1.2\epsilon_{t-1} = q_t \equiv q$. Now we can apply Lemma A.4 with $\varepsilon_{\text{SE}} = 0.5\epsilon_t = 0.5 \max(\epsilon_v, 1.2q/4) = 0.5 \max(\epsilon_v, 0.3q_{t-1})$.

Note: Observe that Lemma A.4 requires independence of $\boldsymbol{a}_{i,(t)}$'s, and the set of missing entries, $\mathcal{M}_{i,(t)}$'s. We have assumed $\boldsymbol{a}_{i,(t)}$'s are independent over i and over t. Notice $\hat{\boldsymbol{P}}_{(t-1)}$ is computed using $\boldsymbol{a}_{i,(t-1)}$'s and older data. Thus, $\hat{\boldsymbol{P}}_{(t-1)}$ is independent of $\boldsymbol{a}_{i,(t)}$'s. At iteration t, we apply Lemma A.4 by conditioning on $\hat{\boldsymbol{P}}_{(t-1)}$, and thus all the matrices being summed are mutually independent conditioned on $\hat{\boldsymbol{P}}_{(t-1)}$.

A.2 Subspace change detection

Main idea of change detection and why it works. We summarize the complete algorithm with the change detection step in Algorithm 2. We perform a projected LS step to interpolate the missing entries, and the subspace update step toggles between the "update phase", and the "detect" phase. Initially it starts in the update phase. After T iterations of update, with T set proportional to $\log(1/\epsilon)$, w.h.p., the first subspace has been accurately estimated. At this point the algorithm enters the "detect" phase. It remains in detect phase until a change is detected after which it enters the update phase again.

The main idea for detecting change is the following. Consider the j-th change and let $\boldsymbol{B} := \boldsymbol{\Psi}\hat{\boldsymbol{L}}_{(t)}$ where $\boldsymbol{\Psi} := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{j-1}\hat{\boldsymbol{P}}_{j-1}^T)$ where $\hat{\boldsymbol{P}}_{j-1} = \hat{\boldsymbol{P}}_{j-1,(T)}$ is the final estimate of the previous subspace. Very briefly, if the subspace has not changed, this matrix will be nearly zero while if it has it will not be. Thus, we can detect change by checking if the top eigenvalue of $\boldsymbol{B}\boldsymbol{B}^T$ is above a threshold. More precisely, it is possible to show that, if the subspace has changed, then $\lambda_{\max}(\boldsymbol{B}\boldsymbol{B}^T) \geq c\text{dist}^2(\hat{\boldsymbol{P}}_{j-1}, \boldsymbol{P}_j)\lambda^-$ w.h.p. where as if there is no change, then $\lambda_{\max}(\boldsymbol{B}\boldsymbol{B}^T) \leq 2\epsilon^2\lambda^-$. Thus by setting the threshold to anywhere between these bounds, one can guarantee correct detection and no false alarms whp.

We now explain how to accurately approximate $\lambda_{\max}(\boldsymbol{B}\boldsymbol{B}^T)$ in a federated fashion. This can be done as follows.

- The master node broadcasts $\hat{\boldsymbol{P}}_{j-1}$ (final estimate of previous subspace) to all the nodes. Each node then computes $\boldsymbol{B}_k := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{j-1}\hat{\boldsymbol{P}}_{j-1}^T)\hat{\boldsymbol{L}}_{k,(t)}$
- The nodes and master then implement FedPM to compute top r eigenvectors of $\boldsymbol{B}\boldsymbol{B}^T = \sum_k \boldsymbol{B}_k\boldsymbol{B}_k^T$. Denote the final output of this algorithm at the L-th iteration as $\hat{\boldsymbol{Q}}_L$.

[3]We point out to the reader that in the l.h.s. of (4), we have $C\epsilon_v^2$ and not $C\epsilon_v$, and thus assuming that $\epsilon_v < 0.2$ is not problematic.

Algorithm 2 FedSTMiss: Federated Over-the-Air Subspace Tracking with Missing Entries.

Input: $\boldsymbol{Y}$, $\mathcal{M}$,

1: Parameters: $T = C\log(1/\epsilon)$, phase = update, $L = C\log(nr)$, $L_{\text{det}} = C\log(nr)$
2: $\tilde{t} = 1$, $j = 1$
3: $\hat{\boldsymbol{P}}_{(0)} \leftarrow \boldsymbol{0}_{n\times r}$,
4: **for all** $t > 0$ **do**
5: at each worker node k, for each $i \in \mathcal{I}_k(t)$ do
6: $\boldsymbol{\Psi} \leftarrow \boldsymbol{I} - \hat{\boldsymbol{P}}_{(t-1)}\hat{\boldsymbol{P}}_{(t-1)}^T$
7: $\tilde{\boldsymbol{y}}_{i,(t)} \leftarrow \boldsymbol{\Psi}\boldsymbol{y}_{i,(t)}$;
8: $\hat{\boldsymbol{\ell}}_{i,(t)} \leftarrow \boldsymbol{y}_{i,(t)} - \mathcal{I}_{\mathcal{M}_{i,(t)}}(\boldsymbol{\Psi}_{\mathcal{M}_i,(t)})^{\dagger}\tilde{\boldsymbol{y}}_{i,(t)}$.
9: **if** phase = update **then**
10: $\hat{\boldsymbol{P}}_{(j,t)} \leftarrow$ `FedPM(`$\hat{\boldsymbol{L}}_{(t)}$`, r, L,` $\hat{\boldsymbol{P}}_{(j,\tilde{t}-1)}$`)`
11: **if** $\tilde{t} = T$ **then**
12: $\hat{\boldsymbol{P}}_j = \hat{\boldsymbol{P}}_{j,T}$, phase = detect
13: **end if**
14: **if** phase = detect **then**
15: $\hat{U}_{\text{det}}$, $\hat{\lambda}_{\text{det}} \leftarrow$ `FedPM(`$\boldsymbol{\Psi}\hat{\boldsymbol{L}}_{(t)}$`, r,` L_{det}`)` {(projected) FedPM}
16: $\tilde{t} = \tilde{t} + 1$
17: **if** $\hat{\lambda}_{\text{det}} \geq \omega_{evals}$ **then**
18: $j = j + 1$, phase = update, $\tilde{t} = 1$
19: **end if**
20: **end if**
21: **end if**
22: **end for**

Output: $\hat{\boldsymbol{P}}$

- In the final iteration, we also have the nodes output $\boldsymbol{B}\boldsymbol{B}^T\hat{\boldsymbol{Q}}_L = \sum_k \boldsymbol{B}_k\boldsymbol{B}_k^T\hat{\boldsymbol{Q}}_L$.
- The master then uses this and computes $\hat{\boldsymbol{Q}}_L^T\boldsymbol{B}\boldsymbol{B}^T\hat{\boldsymbol{Q}}_L$ and computes its top eigenvalue.

It can be shown that $\lambda_{\max}(\hat{\boldsymbol{Q}}_L^T\boldsymbol{B}\boldsymbol{B}^T\hat{\boldsymbol{Q}}_L)$ lies between $0.9\lambda_{\max}(\boldsymbol{B}\boldsymbol{B}^T)$ and $\lambda_{\max}(\boldsymbol{B}\boldsymbol{B}^T)$ w.h.p. and this what allows use to use this as a surrogate for $\lambda_{\max}(\boldsymbol{B}\boldsymbol{B}^T)$. This follows from the result given below.

Corollary A.6 (Eigenvalue convergence)**.** *Consider Lines 10,11 of Algorithm 1 (with $\tau = 1$). Assume that $R < 0.99$ and pick $L = \Omega\left(\frac{1}{\log(1/R)} \cdot \log(n/\epsilon)\right)$. Under the assumptions of Theorem 2.1, with probability at least $1 - L\exp(-cr)$,*

$$\lambda_1(1 - 4\epsilon^2) - \lambda_{r+1}\epsilon^2 - \lambda_r\epsilon \leq \lambda_{\max}(\hat{\boldsymbol{\Lambda}}) \leq (1 + \epsilon)\lambda_1$$

where λ_i is the i-th largest eigenvalue of $\boldsymbol{A}$. Finally, even if $R = 1$, the upper bound still holds.

Observe that the lower bound in Corollary A.6 is positive because it can be further lower bounded by $\lambda_r(1 - 4\epsilon^2) - \lambda_{r+1}\epsilon^2$ and it is assumed that $\lambda_{r+1}/\lambda_r < 0.99$.

Finally, notice that the above approach to approximate the first (top) eigenvalue of $\boldsymbol{B}\boldsymbol{B}^T$ via FedPM does not require any assumptions on gap between its first and second eigenvalues. Just assuming gap between r-th and $(r + 1)$-th eigenvalues is enough.

A.3 Proof that subspace change detection works

We quantify the above intuition in the following lemma. Again, for simplicity, consider that $\boldsymbol{v}_{i,(t)} = 0$.

Lemma A.7 (Subspace Change Detection)**.** *Consider α data vectors in the j-th subspace so that $\boldsymbol{\ell}_i := \boldsymbol{P}_j \boldsymbol{a}_i$. For this proof, let $L = L_{\text{det}} = C \log nr$ and let $\hat{\boldsymbol{Q}}_L$ denote the output of (projected) FedPM – line 13 of Algorithm 2. Recall from the algorithm that the detection threshold $\omega_{evals} = 2\varepsilon^2\lambda^+$. Then, under the assumptions of Theorem 3.3, the following holds.*

1. *If $\boldsymbol{\Psi} := \boldsymbol{I} - \hat{\boldsymbol{P}}_{j-1}\hat{\boldsymbol{P}}_{j-1}^T$ and $\text{dist}(\hat{\boldsymbol{P}}_{j-1}, \boldsymbol{P}_{j-1}) \le \epsilon$, with probability at least $0.99 - 10n^{-10}$,*

$$\lambda_{\max}\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right) \ge \lambda_r\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right) \ge 0.28\lambda^- \text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j)$$
$$\lambda_{\max}\left(\hat{\boldsymbol{Q}}_L^T\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right)\hat{\boldsymbol{Q}}_L\right) \ge 0.9\lambda_{\max}\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right) \ge 0.2\lambda^- \text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j) > \omega_{evals}$$

2. *If $\boldsymbol{\Psi} := \boldsymbol{I} - \hat{\boldsymbol{P}}_j\hat{\boldsymbol{P}}_j^T$ and $\text{dist}(\hat{\boldsymbol{P}}_j, \boldsymbol{P}_j) \le \epsilon$, with probability at least $0.99 - 10n^{-10}$,*

$$\lambda_r\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right) \le \lambda_{\max}\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right) \le 1.37\epsilon^2\lambda^+$$
$$\lambda_{\max}\left(\hat{\boldsymbol{Q}}_L^T\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right)\hat{\boldsymbol{Q}}_L\right) \le 1.1\lambda_{\max}\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right) \le 1.5\varepsilon^2\lambda^+ < \omega_{evals}$$

Before we give the proof, we should mention that the second line of each item of the above lemma (the bounds on $\lambda_{\max}\left(\hat{\boldsymbol{Q}}_L^T\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right)\hat{\boldsymbol{Q}}_L\right)$) follows from the from the first line by using Corollary A.6.

Proof of Lemma A.7. Consider the proof of item 1. Observe that

$$\begin{aligned}
\lambda_r(\boldsymbol{B}\boldsymbol{B}^T) = \lambda_r\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\hat{\boldsymbol{\ell}}_i\hat{\boldsymbol{\ell}}_i^T\boldsymbol{\Psi}\right) &= \lambda_r\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}(\boldsymbol{P}_j\boldsymbol{a}_i\boldsymbol{a}_i^T\boldsymbol{P}_j^T + \boldsymbol{\ell}_i\boldsymbol{e}_i^T + \boldsymbol{e}_i\boldsymbol{\ell}_i^T + \boldsymbol{e}_i\boldsymbol{e}_i^T)\boldsymbol{\Psi}\right) \\
&\ge \lambda_r\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\boldsymbol{P}_j\boldsymbol{a}_i\boldsymbol{a}_i^T\boldsymbol{P}_j^T\boldsymbol{\Psi}\right) + \lambda_{\min}\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}(\boldsymbol{\ell}_i\boldsymbol{e}_i^T + \boldsymbol{e}_i\boldsymbol{\ell}_i^T + \boldsymbol{e}_i\boldsymbol{e}_i^T)\boldsymbol{\Psi}\right) \\
&\ge \lambda_r\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\boldsymbol{P}_j\boldsymbol{a}_i\boldsymbol{a}_i^T\boldsymbol{P}_j^T\boldsymbol{\Psi}\right) - \left\|\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\boldsymbol{e}_i\boldsymbol{e}_i^T\boldsymbol{\Psi}\right\| - 2\left\|\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\boldsymbol{e}_i\boldsymbol{\ell}_i^T\boldsymbol{\Psi}\right\| \\
&\ge \lambda_r\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}\boldsymbol{P}_j\boldsymbol{a}_i\boldsymbol{a}_i^T\boldsymbol{P}_j^T\boldsymbol{\Psi}\right) - 5.4\sqrt{b_0}\lambda^+(\varepsilon^2 + \text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j))
\end{aligned}$$

where the last line follows from Lemma C.2 with $q \equiv \text{dist}(\hat{\boldsymbol{P}}_{j-1}, \boldsymbol{P}_j) \le \text{dist}(\hat{\boldsymbol{P}}_{j-1}, \boldsymbol{P}_{j-1}) + \text{dist}(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j) \le \varepsilon + \text{dist}(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j)$ and $b_0 \equiv$ max-miss-frac-row $\le 0.001/f^2$. Next consider the first term. We define

$\boldsymbol{\Psi P}_j = \boldsymbol{E}_j \boldsymbol{R}_j$ as the reduced QR decomposition. Then,

$$\lambda_r\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi P}_j \boldsymbol{a}_i \boldsymbol{a}_i^T \boldsymbol{P}_j^T \boldsymbol{\Psi}\right) = \lambda_r\left(\boldsymbol{E}_j \boldsymbol{R}_j \left(\frac{1}{\alpha}\sum_i \boldsymbol{a}_i \boldsymbol{a}_i^T\right) \boldsymbol{R}_j^T \boldsymbol{E}_j^T\right)$$
$$= \lambda_{\min}\left(\boldsymbol{R}_j \left(\frac{1}{\alpha}\sum_i \boldsymbol{a}_i \boldsymbol{a}_i^T\right) \boldsymbol{R}_j^T\right)$$

additionally, from Lemma C.2, we know that with high probability $\lambda_{\min}(\frac{1}{\alpha}\sum_i \boldsymbol{a}_i \boldsymbol{a}_i^T) \geq \lambda^- - \epsilon$ and thus, $\frac{1}{\alpha}\sum_i \boldsymbol{a}_i \boldsymbol{a}_i^T - (\lambda^- - \epsilon)\boldsymbol{I} \succeq \boldsymbol{0}$, which gives that

$$0 \leq \lambda_{\min}\left(\boldsymbol{R}_j \left(\frac{1}{\alpha}\sum_i \boldsymbol{a}_i \boldsymbol{a}_i^T - (\lambda^- - \epsilon)\boldsymbol{I}\right) \boldsymbol{R}_j^T\right) \leq \lambda_{\min}\left(\boldsymbol{R}_j \left(\frac{1}{\alpha}\sum_i \boldsymbol{a}_i \boldsymbol{a}_i^T\right) \boldsymbol{R}_j^T\right) - (\lambda^- - \epsilon)\lambda_{\max}(\boldsymbol{R}_j \boldsymbol{R}_j^T)$$

where the last term in the r.h.s. follows from Weyl's inequality. Additionally, since $\sigma_i(\boldsymbol{R}_j) = \sigma_i(\boldsymbol{\Psi P}_j)$ we have

$$\lambda_{\max}(\boldsymbol{R}_j \boldsymbol{R}_j^T) = \lambda_{\max}(\boldsymbol{P}_j^T(\boldsymbol{I} - \boldsymbol{P}_{j-1}\boldsymbol{P}_{j-1}^T + \boldsymbol{P}_{j-1}\boldsymbol{P}_{j-1}^T - \hat{\boldsymbol{P}}_{j-1}\hat{\boldsymbol{P}}_{j-1}^T)\boldsymbol{P}_j) \geq \text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j) - 2\varepsilon$$

Simplifying the above and under the assumptions of Theorem 3.3 with high probability,

$$\lambda_r(\boldsymbol{B}\boldsymbol{B}^T) \geq 0.28\lambda^- \text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j)$$

Similarly,

$$\lambda_{r+1}(\boldsymbol{B}\boldsymbol{B}^T) = \lambda_{r+1}\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}(\boldsymbol{P}_j \boldsymbol{a}_i \boldsymbol{a}_i^T \boldsymbol{P}_j^T + \boldsymbol{\ell}_i \boldsymbol{e}_i^T + \boldsymbol{e}_i \boldsymbol{\ell}_i^T + \boldsymbol{e}_i \boldsymbol{e}_i^T)\boldsymbol{\Psi}\right)$$
$$\leq \lambda_{r+1}\left(\frac{1}{\alpha}\sum_i \boldsymbol{\Psi P}_j \boldsymbol{a}_i \boldsymbol{a}_i^T \boldsymbol{P}_j^T \boldsymbol{\Psi}\right) + \lambda_{\max}\left(\frac{1}{\alpha}\sum_i (\boldsymbol{\ell}_i \boldsymbol{e}_i^T + \boldsymbol{e}_i \boldsymbol{\ell}_i^T + \boldsymbol{e}_i \boldsymbol{e}_i^T)\boldsymbol{\Psi}\right)$$
$$\leq \left\|\frac{1}{\alpha}\sum_i \boldsymbol{\Psi}(\boldsymbol{\ell}_i \boldsymbol{e}_i^T + \boldsymbol{e}_i \boldsymbol{\ell}_i^T + \boldsymbol{e}_i \boldsymbol{e}_i^T)\boldsymbol{\Psi}\right\|$$
$$\leq 2.7\sqrt{b_0}\lambda^+(\varepsilon^2 + \text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j)^2 \leq 0.1\text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j)\lambda^-$$

Thus, under the assumptions of Theorem 3.3, $\lambda_{r+1}(\boldsymbol{B}\boldsymbol{B}^T)/\lambda_r(\boldsymbol{B}\boldsymbol{B}^T) \leq 0.5$ which ensures convergence of FedPM. To be precise, we can use Lemma A.3 to conclude that $\hat{\boldsymbol{Q}}_L$ is within ϵ-accuracy of the top r left singular vectors of $\boldsymbol{B}$. Next, we use Corollary A.6 to lower bound the largest eigenvalue of $\hat{\boldsymbol{\Lambda}} = \hat{\boldsymbol{Q}}_L^T \boldsymbol{B}\boldsymbol{B}^T \hat{\boldsymbol{Q}}_L + \hat{\boldsymbol{Q}}_L^T \boldsymbol{W}_L$.

In the case that the subspace has changed, we showed above that $\lambda_{\max}(\boldsymbol{B}\boldsymbol{B}^T) \geq \lambda_r(\boldsymbol{B}\boldsymbol{B}^T) \geq 0.28\lambda^- \text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j)$ and $\lambda_{r+1}(\boldsymbol{B}\boldsymbol{B}^T) \leq 0.1\lambda^- \text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j)$ hence picking $\epsilon = 0.01$, with high probability,

$$\lambda_{\max}(\hat{\boldsymbol{Q}}_L^T \boldsymbol{B}\boldsymbol{B}^T \hat{\boldsymbol{Q}}_L + \hat{\boldsymbol{Q}}_L^T \boldsymbol{W}_L) \geq 0.2\lambda^- \text{dist}^2(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j)$$

Finally, when the subspace has not changed, all eigenvalues of the matrix, $\boldsymbol{B}\boldsymbol{B}^T$ are of the order of $\varepsilon^2\lambda^+$ (the proof is same as [32] and thus we do not repeat this here) and now using the result of Eigenvalue Convergence,

$$\lambda_{\max}(\hat{\boldsymbol{Q}}_L^T \boldsymbol{B}\boldsymbol{B}^T \hat{\boldsymbol{Q}}_L + \hat{\boldsymbol{Q}}_L^T \boldsymbol{W}_L) \leq \lambda_{\max}(\boldsymbol{B}\boldsymbol{B}^T) + 1.5\epsilon\lambda_r(\boldsymbol{B}\boldsymbol{B}^T) \leq 1.5\lambda^+\varepsilon^2$$

$\boxtimes$

A.4 Proof of Theorem 3.3 (time-varying subspaces)

The only difference in the proof of Theorem 3.3 with the proof of Theorem A.2 is the subspace change detection step. We showed in Lemma A.7 that the (projected) FedPM algorithm is provably capable of detecting subspace changes. In fact, the subspace change is detected within 1 time periods[4]. The idea for this is as follows. Suppose the subspace changed from $\boldsymbol{P}_{j-1}$ to $\boldsymbol{P}_j$ at time t_j. Then, all the data vectors at time $t_j + 1$ are now generated from the subspace $\boldsymbol{P}_j$, but we have a good estimate for the previous subspace which satisfies $\text{dist}(\hat{\boldsymbol{P}}_{j-1,(T)}, \boldsymbol{P}_{j-1}) \leq \varepsilon$, and thus, as explained in Lemma A.7, the matrix, $\sum_k \boldsymbol{B}_k = \sum_k (\boldsymbol{I} - \hat{\boldsymbol{P}}_{j-1,(T)} \hat{\boldsymbol{P}}_{j-1,(T)}^T) \boldsymbol{L}_k$ will have all top r singular values $\Omega(\sqrt{\alpha\lambda^-}\text{dist}(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j))$ and thus the detection steps provably works. In case the subspace has not changed, all the singular values of the matrix are $\mathcal{O}(\sqrt{\alpha\lambda^+}\varepsilon)$. Choosing the threshold, ω_{evals} carefully as specified in Algorithm 2 ensures that there are no false subspace change detections.

Finally, after a subspace change is detected, Algorithm 2 returns to the *update* phase. In the first time instant ($t = 1$) of the j-th subspace interval we start with a different initialization compared to the static case and thus we need to show that the $\boldsymbol{e}_i$'s follow all the required conditions. Since we start with $\hat{\boldsymbol{P}}_{(j-1,T)}$ and since $\text{dist}(\hat{\boldsymbol{P}}_{j-1,(T)}, \boldsymbol{P}_j) \leq \text{dist}(\hat{\boldsymbol{P}}_{j-1,(T)} \boldsymbol{P}_{j-1}) + \text{dist}(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j) \leq \varepsilon + \text{dist}(\boldsymbol{P}_{j-1}, \boldsymbol{P}_j)$. Thus, again, the conditions of Lemma A.4 (condition on $\|\boldsymbol{M}_{2,i}\boldsymbol{P}_j\| = \|(\boldsymbol{\Psi}_{\mathcal{M}_i}^T \boldsymbol{\Psi}_{\mathcal{M}_i})^{-1}\| \text{dist}(\hat{\boldsymbol{P}}_{j-1,(T)}, P) < 3$) is satisfied. Everything else: conditions on $\mathcal{M}_i$, the channel noise $\boldsymbol{W}_l$, the modeling error $\boldsymbol{v}_i$ is exactly the same.

A.5 Proof of Corollary 3.4

The proof follows from using the same idea as Theorem 3.3. Recall that in general, $\mathbb{E}[\boldsymbol{\ell}_t \boldsymbol{v}'] \neq 0$ (this is different from the main result). By Cauchy-Schwarz, we can bound it as $\|\mathbb{E}[\boldsymbol{\ell}_t \boldsymbol{v}']\| \leq \sqrt{\lambda^+ \lambda_v^+}$. Thus, to analyze this case, we need to modify Lemma A.4 for PCA-SDDN as follows: we now need $6\sqrt{b}qf + \frac{\lambda_v^+}{\lambda^-} + \sqrt{\frac{\lambda_v^+}{\lambda^-} f} < 0.4\epsilon_{\text{dist}}$. There is no change to the required lower bound on α. Thus the only change needed to Theorem 3.3 is that we now need $\lambda_v^+/\lambda^- \leq 0.1\epsilon^2/f$. From our definition of $\boldsymbol{v}$, $\lambda_v^+ \leq \text{dist}(\boldsymbol{P}_j, \boldsymbol{P}_{i,k,(t)})^2 \tilde{\lambda}^+$. Using $\lambda^+ \leq \tilde{\lambda}^+$, $\tilde{\lambda}^- < \lambda^-$, a simpler sufficient condition is $\text{dist}(\boldsymbol{P}_j, \boldsymbol{P}_{i,k,(t)})^2 \leq 0.1\epsilon^2/\tilde{f}^2$.

B Convergence Analysis for FedPM

First we define two auxiliary quantities

$$\Gamma_{num}^2(\tau) := \frac{1 + \lambda_{r+1}^2 + \lambda_{r+1}^4 + \ldots \lambda_{r+1}^{2\tau-2}}{\lambda_r^{2\tau-2}}, \quad \Gamma_{denom}^2(\tau) := \frac{1 + \lambda_r^2 + \lambda_r^4 + \cdots + \lambda_r^{2\tau-2}}{\lambda_r^{2\tau-2}}$$

Intuitively, $\Gamma_{num}(\tau)$ captures the effect of the ratio of the "effective channel noise orthogonal to the signal space", and signal energy, while $\Gamma_{denom}(\tau)$ captures the "effective channel noise along the signal space" and the signal energy. The following lemma bounds the reduction in error from iteration $(l-1)\tau$ to $l\tau$.

[4]This is different from the result of existing provable literature which can deal with time-varying subspaces, such as [32] which required two time instants

Lemma B.1 (Descent Lemma)**.** *Consider Algorithm 1. Assume that $R < 0.99$. With probability at least $1 - \exp(-cr)$, the following holds:*

$$\text{dist}_{l\tau} \leq \frac{R^\tau \ \text{dist}_{(l-1)\tau} + \sqrt{n} \ NSR \ \Gamma_{num}(\tau)}{0.9\sqrt{1 - \text{dist}^2_{(l-1)\tau}} - \sqrt{r} \ NSR \ \Gamma_{denom}(\tau)}$$

By recursively applying the above lemma at each iteration, we have the following. It assumes that the initial subspace estimate has error $\text{dist}_0 := \text{dist}(\hat{\boldsymbol{U}}_0, \boldsymbol{U})$. The proof is provided in Appendix B.

B.1 Proof of Lemma B.1 and Theorem 2.1

Proof of Lemma B.1. Consider the setting where we normalize our subpsace estimates every t_0 iterations. Essentially we start with a basis matrix estimate at t_0, and then analyze the subspace error after t iterations, i.e., $\tau = t - t_0$ un-normalized iterations. The subspace estimate can be written as

$$\begin{aligned}
\hat{\boldsymbol{U}}_{t_0+1} &= \boldsymbol{A} Q_{t_0} + \boldsymbol{W}_{t_0+1} \\
\hat{\boldsymbol{U}}_{t_0+2} &= \boldsymbol{A}\hat{\boldsymbol{U}}_{t_0+1} + \boldsymbol{W}_{t_0+2} = \boldsymbol{A}^2 Q_{t_0} + \boldsymbol{A}\boldsymbol{W}_{t_0+1} + \boldsymbol{W}_{t_0+2} \\
&\vdots \\
\hat{\boldsymbol{U}}_{t_0+\tau} &= \hat{\boldsymbol{U}}_t = \boldsymbol{A}^\tau Q_{t_0} + \sum_{i=1}^{\tau} A^{\tau-i}\boldsymbol{W}_{t_0+i}
\end{aligned}$$

which gives

$$\begin{aligned}
\hat{\boldsymbol{U}}_t &= \boldsymbol{A}^\tau \hat{\boldsymbol{U}}_{t_0} R_{t_0}^{-1} + \sum_{i=1}^{\tau} \boldsymbol{A}^{\tau-i}\boldsymbol{W}_{t_0+i} \\
&= \boldsymbol{A}^\tau(\boldsymbol{U}\boldsymbol{U}^T\hat{\boldsymbol{U}}_{t_0} + \boldsymbol{U}_\perp\boldsymbol{U}_\perp^T\hat{\boldsymbol{U}}_{t_0})R_{t_0}^{-1} + \sum_{i=1}^{\tau} \boldsymbol{A}^{\tau-i}(\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{W}_{t_0+i} + \boldsymbol{U}_\perp\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i}) \\
&= \boldsymbol{U}\Lambda^\tau(\boldsymbol{U}^T\hat{\boldsymbol{U}}_{t_0})R_{t_0}^{-1} + \boldsymbol{U}_\perp\Lambda_\perp^\tau(\boldsymbol{U}_\perp^T\hat{\boldsymbol{U}}_{t_0})R_{t_0}^{-1} + \sum_{i=1}^{\tau}\left[\boldsymbol{U}\Lambda^{\tau-i}(\boldsymbol{U}^T\boldsymbol{W}_{t_0+i}) + \boldsymbol{U}_\perp\Lambda_\perp^{\tau-i}(\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i})\right]
\end{aligned}$$

and thus, $\text{dist}(\boldsymbol{U}, \hat{\boldsymbol{U}}_t) = \|\boldsymbol{U}_\perp^T\hat{\boldsymbol{U}}_t R_t^{-1}\|$ simplifies to

$$\begin{aligned}
\text{dist}(\boldsymbol{U}, \hat{\boldsymbol{U}}_t) &= \left\| \left[\Lambda_\perp^\tau(\boldsymbol{U}_\perp^T\hat{\boldsymbol{U}}_{t_0})R_{t_0}^{-1} + \sum_{i=1}^{\tau}\Lambda_\perp^{\tau-i}(\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i})\right] R_t^{-1} \right\| \\
&\leq \left(\|\Lambda_\perp^\tau\|\|\boldsymbol{U}_\perp^T\hat{\boldsymbol{U}}_{t_0}R_{t_0}^{-1}\| + \left\|\sum_{i=1}^{\tau}\Lambda_\perp^{\tau-i}(\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i})\right\| \right) \|R_t^{-1}\| \\
&= \left(\|\Lambda_\perp^\tau\|\text{dist}(\boldsymbol{U}, \hat{\boldsymbol{U}}_{t_0}) + \left\|\sum_{i=1}^{\tau}\Lambda_\perp^{\tau-i}(\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i})\right\| \right) \|R_t^{-1}\| \\
&\leq \frac{\|\Lambda_\perp^\tau\|\text{dist}(\boldsymbol{U}, \hat{\boldsymbol{U}}_{t_0}) + \left\|\sum_{i=1}^{\tau}\Lambda_\perp^{\tau-i}(\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i})\right\|}{\sigma_r(R_t)}
\end{aligned}$$

We also have that

$$\begin{aligned}\sigma_r^2(R_t) = \sigma_r^2(\hat{\boldsymbol{U}}_t) &= \lambda_{\min}((\boldsymbol{U}\boldsymbol{U}^T\hat{\boldsymbol{U}}_t + \boldsymbol{U}_\perp\boldsymbol{U}_\perp^T\hat{\boldsymbol{U}}_t)^T(\boldsymbol{U}\boldsymbol{U}^T\hat{\boldsymbol{U}}_t + \boldsymbol{U}_\perp\boldsymbol{U}_\perp^T\hat{\boldsymbol{U}}_t)) \\ &\geq \lambda_{\min}(\hat{\boldsymbol{U}}_t^T\boldsymbol{U}\boldsymbol{U}^T\hat{\boldsymbol{U}}_t) = \sigma_r^2(\boldsymbol{U}^T\hat{\boldsymbol{U}}_t) \\ \implies \sigma_r(\boldsymbol{U}^T\hat{\boldsymbol{U}}_t) &= \sigma_r\left(\Lambda^\tau\left(\boldsymbol{U}^TQ_{t_0} + \sum_{i=1}^{\tau}\Lambda^{-i}\boldsymbol{U}^T\boldsymbol{W}_{t_0+i}\right)\right) \\ &\geq \lambda_r^\tau\left[\sigma_r(\boldsymbol{U}^TQ_{t_0}) - \left\|\sum_{i=1}^{\tau}\Lambda^{-i}\boldsymbol{U}^T\boldsymbol{W}_{t_0+i}\right\|\right]\end{aligned}$$

We define $\text{dist}(\boldsymbol{U}, \hat{\boldsymbol{U}}_{t_0}) = \text{dist}(\boldsymbol{U}, Q_{t_0}) = \text{dist}_{t_0}$ and $R = \lambda_{r+1}/\lambda_r$, $\nu = \max(1, \lambda_{r+1})/\lambda_r$ and thus we have

$$\begin{aligned}\text{dist}(\boldsymbol{U}, \hat{\boldsymbol{U}}_t) &\leq \frac{\|\Lambda_\perp^\tau\|\text{dist}(\boldsymbol{U}, \hat{\boldsymbol{U}}_{t_0}) + \left\|\sum_{i=1}^{\tau}\Lambda_\perp^{\tau-i}(\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i})\right\|}{\lambda_r^\tau\left[\sqrt{1-\text{dist}^2(\boldsymbol{U}, \hat{\boldsymbol{U}}_{t_0})} - \|\sum_{i=1}^{\tau}\Lambda^{-i}\boldsymbol{U}^T\boldsymbol{W}_{t_0+i}\|\right]} \\ &\leq \frac{R^\tau\text{dist}_{t_0} + \lambda_r^{-\tau}\|\sum_{i=1}^{\tau}\Lambda_\perp^{\tau-i}\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i}\|}{\sqrt{1-\text{dist}_{t_0}^2} - \|\sum_{i=1}^{\tau}\Lambda^{-i}\boldsymbol{U}^T\boldsymbol{W}_{t_0+i}\|}\end{aligned}$$

notice that the entries of $\boldsymbol{U}^T\boldsymbol{W}_{t_0+i}$ and $\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i}$ are i.i.d. Gaussian r.v's with variance σ_c^2. Next we define the matrix $M = \sum_{i=1}^{\tau}\Lambda_\perp^{\tau-i}(\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i})$ and we apply Theorem B.2 to M. We can apply this theorem because we know that each entry of M is a weighted sum of τ indepdendent Gaussian r.v.'s. In other words

$$\begin{aligned}M_{jk} &= \sum_{i=1}^{\tau}(\lambda_\perp)_j^{\tau-i}(\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i})_{jk} \\ \implies M_{jk} \sim \mathcal{N}\left(0, \sigma_c^2\sum_{i=1}^{\tau}(\lambda_\perp)_j^{2(\tau-i)}\right) &\implies \max_{jk}\|(M)_{jk}\|_{\psi_2} = \sigma_c\sqrt{\sum_{i=1}^{\tau}\lambda_{r+1}^{2(\tau-i)}}\end{aligned}$$

Recall that there is a factor of $\lambda_r^{-\tau}$ multiplying M so effectively, the sub-Gaussian norm is $K = \lambda_r^{-\tau}\sigma_c\sqrt{\sum_{i=1}^{\tau}\lambda_{r+1}^{2(\tau-i)}} = \text{NSR}\cdot\Gamma_{num}(\tau)$. Now, using Theorem B.2, we get that with probability at least $1 - e^{-\epsilon^2}$

$$\|\sum_{i=1}^{\tau}\Lambda_\perp^{\tau-i}\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i}\| \leq C\text{NSR}\cdot\Gamma_{num}(\tau)\cdot(\sqrt{n-r} + \sqrt{r} + \epsilon)$$

and now picking $\epsilon = 0.01\sqrt{n}$ followed by simple algebra yields

$$\Pr\left(\|\sum_{i=1}^{\tau}\Lambda_\perp^{\tau-i}\boldsymbol{U}_\perp^T\boldsymbol{W}_{t_0+i}\| \leq \sqrt{n}\text{NSR}\cdot\Gamma_{num}(\tau)\right) \geq 1 - \exp(-cn)$$

Next consider the denominator term. Again, we notice that the matrix $M = \sum_{i=1}^{\tau} \Lambda^{-i} \boldsymbol{U}^T \boldsymbol{W}_{t_0+i}$ has entries that are gaussian r.v.'s and are independent. Moreover, the sub Gaussian norm bound is

$$M_{jk} = \sum_{i=1}^{\tau} \lambda_j^{-i} (\boldsymbol{U}^T \boldsymbol{W}_{t_0+i})_{jk}$$

$$\implies M_{jk} \sim \mathcal{N}\left(0, \sigma_c^2 \sum_{i=1}^{\tau} \lambda_j^{-2i}\right) \implies \max_{jk} \|(M)_{jk}\|_{\psi_2} = \sigma_c \sqrt{\sum_{i=1}^{\tau} \lambda_r^{-2i}} := \mathrm{NSR} \cdot \Gamma_{denom}(\tau)$$

Now we apply Theorem B.2 to get that with probability $1 - \exp(-\epsilon^2)$

$$\left\| \sum_{i=1}^{\tau} \Lambda^{-i} \boldsymbol{U}^T \boldsymbol{W}_{t_0+i} \right\| \leq \mathrm{NSR} \cdot \Gamma_{denom}(\tau) \cdot (2\sqrt{r} + \epsilon)$$

picking $\epsilon = 0.01\sqrt{r}$ yields that

$$\Pr\left(\left\| \sum_{i=1}^{\tau} \Lambda^{-i} \boldsymbol{U}^T \boldsymbol{W}_{t_0+i} \right\| \leq \sqrt{r} \mathrm{NSR} \cdot \Gamma_{denom}(\tau) \right) \geq 1 - \exp(-cr)$$

This completes the proof of Lemma B.1. ⊠

Proof of Theorem 2.1. The idea for proving Theorem 2.1 is a straightforward extension from Lemma B.1. Consider $\tau = 1$, and assume that the initial subspace estimtate, $\hat{\boldsymbol{U}}_0$ satisfies $\mathrm{dist}(\hat{\boldsymbol{U}}_0, \boldsymbol{U}) = \mathrm{dist}_0 < 1$ we know that with probability $1 - \exp(-cr) - \exp(-cn)$,

$$\begin{aligned} \mathrm{dist}(\hat{\boldsymbol{U}}_\tau, \boldsymbol{U}) &\leq \frac{R^\tau \mathrm{dist}_0 + \sqrt{n}\mathrm{NSR}\Gamma_{num}(\tau)}{0.9\sqrt{1 - \mathrm{dist}_0^2} - \sqrt{r}\mathrm{NSR}\Gamma_{denom}(\tau)} \\ &= \frac{R\mathrm{dist}_0 + \sqrt{n}\mathrm{NSR}}{0.9\sqrt{1 - \mathrm{dist}_0^2} - \sqrt{r}\mathrm{NSR}} \end{aligned}$$

thus, as long as $\mathrm{NSR} \leq 0.2\sqrt{\frac{1-\mathrm{dist}_0^2}{r}}$ the denominator is positive. Next, to achieve an ϵ-accurate estimate, we note that the second term in the numerator is the larger term (since $R < 1$ and this goes to 0 with every iteration) and thus as long as $\mathrm{NSR} \leq \frac{\epsilon}{\sqrt{n}}$ we can ensure that the numerator is small enough. Combining the two bounds, followed by a union bound over L iterations gives the final conclusion.

Finally, consider the case of $\tau > 1$ and the l-th iteration. Assume that $\lambda_r > 1$. This is used to simplify the $\Gamma_{denom}(\tau)$ expression as follows: $\Gamma_{denom}^2(\tau) = (1 + \lambda_r^2 + \cdots + \lambda_r^{2\tau-2})/\lambda_r^{2\tau-2} = \sum_{i=0}^{\tau-1} 1/\lambda_r^{2i} \leq \sum_{i=0}^{\infty} 1/\lambda_r^{2i} = \frac{\lambda_r^2}{\lambda_r^2 - 1}$. Using the same reasoning as in the $\tau = 1$ case, as long as

$$\mathrm{NSR} \leq 0.2\sqrt{\frac{\lambda_r^2 - 1}{\lambda_r^2}} \cdot \sqrt{\frac{1 - \mathrm{dist}_{(l-1)\tau}^2}{r}}$$

the denominator is positive. We also have that $\Gamma_{num}^2(\tau) = \sum_{i=1}^{\tau} \lambda_{r+1}^{2(\tau-i)}/\lambda_r^{2\tau} \leq \tau R^{2\tau-2}$. Thus, as long as $\mathrm{NSR} \leq \frac{\epsilon}{\sqrt{n}} \cdot \frac{1}{\sqrt{\tau} R^{\tau-1}}$ the first term of the numerator is small enough and this gives us the final result. ⊠

B.2 Eigenvalue Convergence

Proof of Corollary A.6. We now wish to compute the error bounds of in convergence of eigenvalues. To this end, at the end of L iterations, we compute $\hat{\Lambda} = \hat{\boldsymbol{Q}}_L^T \boldsymbol{A} \hat{\boldsymbol{Q}}_L + \hat{\boldsymbol{Q}}_L^T \boldsymbol{W}_L$. The intuition is that if the eigenvectors are estimated well, then this matrix will be approximately diagonal (off diagonal entries $\approx \epsilon$), and the diagonal entries will be close to the true eigenvalues. Furthermore, in the application of this result for the Subspace Change detection problem, we will only consider the largest eigenvalue of $\hat{\boldsymbol{\Lambda}}$ and thus we have

$$\begin{aligned}\lambda_{\max}(\hat{\boldsymbol{\Lambda}}) &= \lambda_{\max}(\hat{\boldsymbol{Q}}_L^T \boldsymbol{A} \hat{\boldsymbol{Q}}_L + \hat{\boldsymbol{Q}}_L^T \boldsymbol{W}_L) = \lambda_{\max}(\Lambda + (\hat{\boldsymbol{Q}}_L^T \boldsymbol{A} \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}) + \hat{\boldsymbol{Q}}_L^T \boldsymbol{W}_L) \\ &\geq \lambda_{\max}(\boldsymbol{\Lambda}) - \|\hat{\boldsymbol{Q}}_L^T \boldsymbol{A} \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}\| - \|\hat{\boldsymbol{Q}}_L^T \boldsymbol{W}_L\| \geq \lambda_1 - \|\hat{\boldsymbol{Q}}_L^T \boldsymbol{A} \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}\| - \|\boldsymbol{W}_L\|\end{aligned}$$

The second term can be upper bounded as follows

$$\begin{aligned}\|\hat{\boldsymbol{Q}}_L^T \boldsymbol{A} \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}\| &= \|(\hat{\boldsymbol{Q}}_L^T \boldsymbol{U} \boldsymbol{\Lambda} \boldsymbol{U}^T \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}) + \hat{\boldsymbol{Q}}_L^T \boldsymbol{U}_\perp \boldsymbol{\Lambda}_\perp \boldsymbol{U}_\perp^T \hat{\boldsymbol{Q}}_L\| \\ &\leq \|\hat{\boldsymbol{Q}}_L^T \boldsymbol{U} \boldsymbol{\Lambda} \boldsymbol{U}^T \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}\| + \|\hat{\boldsymbol{Q}}_L^T \boldsymbol{U}_\perp \boldsymbol{\Lambda}_\perp \boldsymbol{U}_\perp^T \hat{\boldsymbol{Q}}_L\| \\ &\leq \|\hat{\boldsymbol{Q}}_L^T \boldsymbol{U} \boldsymbol{\Lambda} \boldsymbol{U}^T \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}\| + \|\boldsymbol{\Lambda}_\perp\| \|\boldsymbol{U}_\perp^T \hat{\boldsymbol{Q}}_L\|^2 \\ &= \|\hat{\boldsymbol{Q}}_L^T \boldsymbol{U} \boldsymbol{\Lambda} \boldsymbol{U}^T \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}\| + \|\boldsymbol{\Lambda}_\perp\| \|\boldsymbol{U}_\perp \boldsymbol{U}_\perp^T \hat{\boldsymbol{Q}}_L\|^2 \\ &\leq \|\hat{\boldsymbol{Q}}_L^T \boldsymbol{U} \boldsymbol{\Lambda} \boldsymbol{U}^T \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}\| + \lambda_{r+1} \mathrm{dist}^2(\hat{\boldsymbol{Q}}_L, \boldsymbol{U})\end{aligned}$$

The first term above can be bounded as

$$\begin{aligned}\|\hat{\boldsymbol{Q}}_L^T \boldsymbol{U} \boldsymbol{\Lambda} \boldsymbol{U}^T \hat{\boldsymbol{Q}}_L - \boldsymbol{\Lambda}\| &= \|(\boldsymbol{I} - \boldsymbol{I} + \hat{\boldsymbol{Q}}_L^T \boldsymbol{U}) \boldsymbol{\Lambda} (\boldsymbol{U}^T \hat{\boldsymbol{Q}}_L + \boldsymbol{I} - \boldsymbol{I}) - \boldsymbol{\Lambda}\| \\ &\leq \|(\hat{\boldsymbol{Q}}_L^T \boldsymbol{U} - \boldsymbol{I}) \boldsymbol{\Lambda}\| + \|\boldsymbol{\Lambda} (\boldsymbol{U}^T \hat{\boldsymbol{Q}}_L - \boldsymbol{I})\| + \|(\hat{\boldsymbol{Q}}_L^T \boldsymbol{U} - \boldsymbol{I}) \boldsymbol{\Lambda} (\boldsymbol{U}^T \hat{\boldsymbol{Q}}_L - \boldsymbol{I})\| \\ &\leq \lambda_1 (2\|\boldsymbol{I} - \hat{\boldsymbol{Q}}_L^T \boldsymbol{U}\| + \|\boldsymbol{I} - \hat{\boldsymbol{Q}}_L^T \boldsymbol{U}\|^2) \\ &\leq \lambda_1 (2(1 - \sigma_r(\hat{\boldsymbol{Q}}_L^T \boldsymbol{U})) + (1 - \sigma_r(\hat{\boldsymbol{Q}}_L^T \boldsymbol{U}))^2)\end{aligned}$$

and since $\mathrm{dist}^2(\hat{\boldsymbol{Q}}_L, \boldsymbol{U}) = 1 - \sigma_r^2(\hat{\boldsymbol{Q}}_L^T \boldsymbol{U}) \leq \epsilon^2$ and thus we get that $\sigma_r(\hat{\boldsymbol{Q}}_L^T \boldsymbol{U}) \geq \sqrt{1 - \epsilon^2} \geq 1 - \epsilon^2$. Finally, the assumption on the channel noise implies that with high probability, $\|\boldsymbol{W}_L\| \leq C\sqrt{n}\sigma_c \leq 1.5\lambda_r \epsilon$. Thus,

$$\lambda_{\max}(\hat{\boldsymbol{\Lambda}}) \geq \lambda_1(1 - 4\epsilon^2) - \lambda_{r+1}\epsilon^2 - \lambda_r \epsilon$$

We also get

$$\lambda_{\max}(\hat{\boldsymbol{\Lambda}}) \leq \lambda_{\max}(\hat{\boldsymbol{Q}}_L^T \boldsymbol{B}\boldsymbol{B}^T \hat{\boldsymbol{Q}}_L) + \|\boldsymbol{W}_L\| \leq \|\hat{\boldsymbol{Q}}_L\|^2 \|\boldsymbol{B}\boldsymbol{B}^T\| + \|\boldsymbol{W}_L\| = \lambda_{\max}(\boldsymbol{B}\boldsymbol{B}^T) + 1.5\lambda_r \epsilon$$

⊠

Proof of Item 3 of Theorem 3.3. The proof follows by application of Theorem B.2, B.3 to a standard normal random matrix, and definition of principal angles. Recall that $(\hat{\boldsymbol{U}}_0)_{ij} \overset{iid}{\sim} \mathcal{N}(0, 1)$ and consider

its reduced QR decomposition, $\hat{\boldsymbol{U}}_0 = \hat{\boldsymbol{Q}}_0 R_0$. We know that

$$\begin{aligned}
\text{dist}^2(\hat{\boldsymbol{U}}_0, \boldsymbol{U}) &= \|(\boldsymbol{I} - \hat{\boldsymbol{Q}}_0\hat{\boldsymbol{Q}}_0^T)\boldsymbol{U}\|^2 = \lambda_{\max}(\boldsymbol{I} - \boldsymbol{U}^T\hat{\boldsymbol{Q}}_0\hat{\boldsymbol{Q}}_0^T\boldsymbol{U}) \\
&= 1 - \lambda_{\min}(\boldsymbol{U}^T\hat{\boldsymbol{Q}}_0\hat{\boldsymbol{Q}}_0^T\boldsymbol{U}) = 1 - \lambda_{\min}(\boldsymbol{U}^T\hat{\boldsymbol{U}}_0 R_0^{-1}(R_0^{-1})^T\hat{\boldsymbol{U}}_0^T\boldsymbol{U}) \\
&\overset{(a)}{\leq} 1 - \lambda_{\min}(\boldsymbol{U}^T\hat{\boldsymbol{U}}_0\hat{\boldsymbol{U}}_0^T\boldsymbol{U})\lambda_{\min}(R_0^{-1}(R_0^{-1})^T)) \\
&= 1 - \frac{\sigma^2_{\min}(\boldsymbol{U}^T\hat{\boldsymbol{U}}_0)}{\|\hat{\boldsymbol{U}}_0\|_2^2}
\end{aligned}$$

where (a) follows from Ostrowski's Theorem (Theorem 4.5.9, [36]) and the last relation follows since reduced qr decomposition preserves the singular values. It is easy to see that $(\boldsymbol{U}^T\hat{\boldsymbol{U}}_0)_{ij} \sim \mathcal{N}(0,1)$. We can apply Theorem B.3 to get that with probability at least $1 - \exp(-cr) - (c/\gamma)$,

$$\sigma_{\min}(\boldsymbol{U}^T\hat{\boldsymbol{U}}_0) \geq c(\sqrt{r} - \sqrt{r-1})/\gamma$$

and we also know that $\sqrt{r} - \sqrt{r-1} = O(1/\sqrt{r})$. Additionally, the denominator term is bounded using Theorem B.2 as done before and thus, with probability $1 - \exp(-\epsilon^2)$,

$$\|\hat{\boldsymbol{U}}_0\| \leq C(\sqrt{n} + \sqrt{r} + \epsilon)$$

and now picking $\epsilon = 0.01\sqrt{n}$ we get that with probability at least $1 - \exp(-cn) - \exp(-cr) - (1/c\gamma)$,

$$\text{dist}^2(\hat{\boldsymbol{U}}_0, \boldsymbol{U}) \leq 1 - \frac{1}{\gamma n r}$$

which completes the proof. ⊠

B.3 Preliminaries

The following result is Theorem 4.4.5, [37]

Theorem B.2 (Upper Bounding Spectral Norm)**.** *Let A be a $m \times n$ random matrix whose entries are independent zero-mean sub-Gaussian r.v.'s and let $K = \max_{i,j} \|A_{i,j}\|_{\psi_2}$. Then for any $\epsilon > 0$ with probability at least $1 - 2\exp(-\epsilon^2)$,*

$$\|A\| \leq CK(\sqrt{m} + \sqrt{n} + \epsilon)$$

The following result (Theorem 1.1, [38]) bounds the smallest singular value of a random rectangular matrix.

Theorem B.3 (Lower Bounding Smallest Singular Value for Rectangular matrices)**.** *. Let A be a $m \times n$ random matrix whose entries are independent zero-mean sub-Gaussian r.v.'s. Then for any $\epsilon > 0$ we have*

$$\sigma_{\min}(A) \geq \epsilon C_K(\sqrt{m} - \sqrt{n-1})$$

with probability at least $1 - \exp(-c_K n) - (c_K\epsilon)^{m-n+1}$. Here, $K = max_{i,j}\|A_{i,j}\|_{\psi_2}$.

Theorem B.4 (Davis-Kahan $\sin\theta$ theorem)**.** *Let $\boldsymbol{D}_0$ be a Hermitian matrix whose span of top r eigenvectors equals* $\mathrm{Span}(\boldsymbol{P}_1)$*. Let $\boldsymbol{D}$ be the Hermitian matrix with top r eigenvectors $\boldsymbol{P}_2$. Then,*

$$\begin{aligned}\mathrm{dist}(\boldsymbol{P}_2, \boldsymbol{P}_1) \leq \frac{\|(\boldsymbol{D}-\boldsymbol{D}_0)\boldsymbol{P}_1\|_2}{\lambda_r(\boldsymbol{D}_0)-\lambda_{r+1}(\boldsymbol{D})} &\leq \frac{\|(\boldsymbol{D}-\boldsymbol{D}_0)\boldsymbol{P}_1\|_2}{\lambda_r(\boldsymbol{D}_0)-\lambda_{r+1}(\boldsymbol{D}_0)-\lambda_{\max}(\boldsymbol{D}-\boldsymbol{D}_0)} \\ &\leq \frac{\|\boldsymbol{D}-\boldsymbol{D}_0\|_2}{\lambda_r(\boldsymbol{D}_0)-\lambda_{r+1}(\boldsymbol{D}_0)-\|\boldsymbol{D}-\boldsymbol{D}_0\|} \end{aligned} \tag{7}$$

as long as the denominator is positive. The second inequality follows from the first using Weyl's inequality.

The following result is the Matrix Bernstein result (Theorem 1.6, [35]).

Theorem B.5 (Matrix Bernstein Concentration)**.** *Given an d-length sequence of $n_1 \times n_2$ dimensional random matrices. Assume the following holds. (i) the matrices $\boldsymbol{Z}_t$ are mutually independent, (ii) $\mathbb{P}(\|\boldsymbol{Z}_t\| \leq R) = 1$, and (iii) $\max\left\{\left\|\frac{1}{d}\sum_t \mathbb{E}\left[\boldsymbol{Z}_t^T\boldsymbol{Z}_t\right]\right\|, \left\|\frac{1}{d}\sum_t \mathbb{E}\left[\boldsymbol{Z}_t\boldsymbol{Z}_t^T\right]\right\|\right\} \leq \sigma^2$. Then, for an $\epsilon > 0$*

$$\mathbb{P}\left(\left\|\frac{1}{d}\sum_t \boldsymbol{Z}_t - \frac{1}{d}\sum_t \mathbb{E}[\boldsymbol{Z}_t]\right\| \leq \epsilon\right) \geq 1-(n_1+n_2)\exp\left(\frac{-d\epsilon^2}{2\left(\sigma^2+R\epsilon\right)}\right). \tag{8}$$

C Proof of (stronger version of) PCA SDDN

Theorem C.1. *Assume that the data satisfies $\boldsymbol{y}_i = \boldsymbol{\ell}_i + \boldsymbol{e}_i + \boldsymbol{v}_i$ with $\boldsymbol{\ell}_i = \boldsymbol{P}\boldsymbol{a}_i$, $\boldsymbol{e}_i = \boldsymbol{M}_i\boldsymbol{\ell}_i = \boldsymbol{M}_{1,i}\boldsymbol{M}_{2,i}\boldsymbol{\ell}_i$ with $\|\frac{1}{\alpha}\sum_i \boldsymbol{M}_{1,i}\boldsymbol{M}_{1,i}^T\| \leq b$ and $\|\boldsymbol{M}_{2,i}\boldsymbol{P}\| = q \leq 3$. Define $H(\alpha) = C\sqrt{\eta}qf\sqrt{\frac{r\log n}{\alpha}}$ and $G_{den}(\alpha) = c\eta f\sqrt{\frac{r\log n}{\alpha}}$. Furthermore, assume that the data-dependency matrices $\boldsymbol{M}_i$'s satisfy the assumption with constants b, q which satisfy*

$$6\sqrt{b}qf + \frac{\lambda_v^+}{\lambda^-} + H(\alpha) + G_{den}(\alpha) < 1$$

Then, with probability at least $1-10n^{-10}$, the matrix $\hat{\boldsymbol{P}}$ of top-r eigenvectors of the sample covariance matrix, $\frac{1}{\alpha}\sum_i \boldsymbol{y}_i\boldsymbol{y}_i^T$ satisfy the following.

$$\mathrm{dist}(\hat{\boldsymbol{P}}, \boldsymbol{P}) \leq \frac{2\sqrt{b}qf + \frac{\lambda_v^+}{\lambda^-} + H(\alpha)}{1-6\sqrt{b}qf - \frac{\lambda_v^+}{\lambda^-} - H(\alpha) - G_{den}(\alpha)}$$

Proof of Theorem C.1. We will first define matrices in accordance with Theorem B.4. For this example, we define $\boldsymbol{D}_0 = \frac{1}{\alpha}\sum_t \boldsymbol{\ell}_t\boldsymbol{\ell}_t^T$. Notice that this is a Hermitian matrix $\boldsymbol{P}$ as the top r eigenvectors. Next, let $\boldsymbol{D} = \frac{1}{\alpha}\sum_t \boldsymbol{y}\boldsymbol{y}^T$ and let $\hat{\boldsymbol{P}}$ denote the matrix of D's top r eigenvectors. Observe

$$\begin{aligned}\boldsymbol{D}-\boldsymbol{D}_0 = \frac{1}{\alpha}\sum_i(\boldsymbol{y}_i\boldsymbol{y}_i^T - \boldsymbol{\ell}_i\boldsymbol{\ell}_i^T) &= \frac{1}{\alpha}\sum_i \boldsymbol{\ell}_i\boldsymbol{e}_i^T + \boldsymbol{e}_i\boldsymbol{\ell}_i^T + \boldsymbol{e}_i\boldsymbol{e}_i^T + \boldsymbol{v}_i\boldsymbol{v}_i^T + \boldsymbol{v}_i\boldsymbol{e}_i^T + \boldsymbol{e}_i\boldsymbol{v}_i^T + \boldsymbol{\ell}_i\boldsymbol{v}_i^T + \boldsymbol{v}_i\boldsymbol{\ell}_i^T \\ &:= \mathrm{cross}_{\boldsymbol{\ell},\boldsymbol{e}} + \mathrm{cross}_{\boldsymbol{\ell},\boldsymbol{e}}^T + \mathrm{noise}_{\boldsymbol{e}} + \mathrm{noise}_{\boldsymbol{v}} + \mathrm{cross}_{\boldsymbol{\ell},\boldsymbol{v}} + \mathrm{cross}_{\boldsymbol{\ell},\boldsymbol{v}}^T + \mathrm{cross}_{\boldsymbol{v},\boldsymbol{e}} + \mathrm{cross}_{\boldsymbol{v},\boldsymbol{e}}^T \\ &= \mathrm{cross} + \mathrm{cross}^T + \mathrm{noise}\end{aligned}$$

Also notice that $\lambda_{r+1}(\boldsymbol{D}_0) = 0$, $\lambda_r(\boldsymbol{D}) = \lambda_{\min}\left(\frac{1}{\alpha}\sum_t \boldsymbol{a}\boldsymbol{a}^T\right)$. Now, applying Theorem B.4,

$$\operatorname{dist}(\hat{\boldsymbol{P}}, \boldsymbol{P}) \leq \frac{2\|\text{cross}\| + \|\text{noise}\|}{\lambda_{\min}\left(\frac{1}{\alpha}\sum_t \boldsymbol{a}\boldsymbol{a}^T\right) - \text{numerator}}$$

Now, we can bound $\|\text{cross}\| \leq \|\mathbb{E}[\text{cross}]\| + \|\text{cross} - \mathbb{E}[\text{cross}]\|$ and similarly for the noise term. We use the Cauchy-Schwartz inequality for bounding the expected values of cross, noise as follows.

Recall that $\boldsymbol{M}_i = \boldsymbol{M}_{2,i}\boldsymbol{M}_{1,i}$ with $b := \|\frac{1}{\alpha}\sum_i \boldsymbol{M}_{2,i}\boldsymbol{M}_{2,i}^T\|$ and $q := \max_i \|\boldsymbol{M}_{1,i}\boldsymbol{P}\| \leq q < 1$. Thus,

$$\begin{aligned}
\|\mathbb{E}[\text{noise}]\| &\leq \left\|\frac{1}{\alpha}\sum_i \boldsymbol{M}_i\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^T\boldsymbol{M}_{1,i}^T\boldsymbol{M}_{2,i}^T\right\|_2 + \|\Sigma_{\boldsymbol{v}}\|_2 \qquad (9)\\
&\leq \sqrt{\left\|\frac{1}{\alpha}\sum_i \boldsymbol{M}_i\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^T\boldsymbol{M}_{1,i}^T\boldsymbol{M}_{1,i}\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^T\boldsymbol{M}_i^T\right\|_2 \left\|\frac{1}{\alpha}\sum_i \boldsymbol{M}_{2,i}\boldsymbol{M}_{2,i}^T\right\|_2} + \lambda_v^+\\
&\leq \sqrt{\max_i \|\boldsymbol{M}_i\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^T\boldsymbol{M}_{1,i}^T\|_2^2\, b} + \lambda_v^+ \leq \sqrt{b}q\lambda^+ + \lambda_v^+ \qquad (10)
\end{aligned}$$

Similarly,

$$\begin{aligned}
\|\mathbb{E}[\text{cross}_{\boldsymbol{\ell},\boldsymbol{e}}]\|^2 = \left\|\frac{1}{\alpha}\sum_i \boldsymbol{M}_{2,i}\boldsymbol{M}_{1,i}\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^T\right\|_2^2 &\leq \left\|\frac{1}{\alpha}\sum_i \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^T\boldsymbol{M}_{1,i}^T\boldsymbol{M}_{1,i}\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^T\right\|_2 \left\|\frac{1}{\alpha}\sum_i \boldsymbol{M}_{2,i}\boldsymbol{M}_{2,i}^T\right\|_2\\
&\leq \max_i \|\boldsymbol{M}_{1,i}\boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^T\|_2^2\; b \leq (q\lambda^+)^2 b. \qquad (11)
\end{aligned}$$

And it is easy to see that $\mathbb{E}[\text{cross}_{\boldsymbol{\ell},\boldsymbol{v}}] = 0$ and $\mathbb{E}[\text{cross}_{\boldsymbol{e},\boldsymbol{v}}] = 0$. We now lower bound $\lambda_{\min}\left(\frac{1}{\alpha}\sum_i \boldsymbol{a}_i\boldsymbol{a}_i^T\right)$ as

$$\begin{aligned}
\lambda_{\min}\left(\frac{1}{\alpha}\sum_i \boldsymbol{a}_i\boldsymbol{a}_i^T\right) &= \lambda_{\min}\left(\boldsymbol{\Lambda} - \left(\frac{1}{\alpha}\sum_i \boldsymbol{a}_i\boldsymbol{a}_i^T - \boldsymbol{\Lambda}\right)\right)\\
&\geq \lambda_{\min}(\boldsymbol{\Lambda}) - \lambda_{\max}\left(\frac{1}{\alpha}\sum_i \boldsymbol{a}_i\boldsymbol{a}_i^T - \boldsymbol{\Lambda}\right)\\
&\geq \lambda^- - \left\|\frac{1}{\alpha}\sum_i \boldsymbol{a}_i\boldsymbol{a}_i^T - \boldsymbol{\Lambda}\right\|
\end{aligned}$$

and thus we have

$$\operatorname{dist}(\hat{\boldsymbol{P}}, \boldsymbol{P}) \leq \frac{3\sqrt{b}q\lambda^+ + \lambda_v^+ + 2\|\text{cross} - \mathbb{E}[\text{cross}]\| + \|\text{noise} - \mathbb{E}[\text{noise}]\|}{\lambda^- - \left\|\frac{1}{\alpha}\sum_i \boldsymbol{a}_i\boldsymbol{a}_i^T - \boldsymbol{\Lambda}\right\| - \text{numerator}}$$

Bounding the "Statistical Errors". We use concentration bounds from the Lemma C.2. Notice

that

$$\begin{aligned}
&\|\text{noise} - \mathbb{E}[\text{noise}]\| + 2\|\text{cross} - \mathbb{E}[\text{cross}]\| \\
&\leq \left\|\frac{1}{\alpha}\sum_i (\boldsymbol{e}_i\boldsymbol{e}_i^T - \mathbb{E}[\boldsymbol{e}_i\boldsymbol{e}_i^T])\right\| + \left\|\frac{1}{\alpha}\sum_i (\boldsymbol{v}_i\boldsymbol{v}_i^T - \mathbb{E}[\boldsymbol{v}_i\boldsymbol{v}_i^T])\right\| + 2\left\|\frac{1}{\alpha}\sum_i (\boldsymbol{\ell}_i\boldsymbol{e}_i^T - \mathbb{E}[\boldsymbol{\ell}_i\boldsymbol{e}_i^T])\right\| \\
&\qquad + 2\left\|\frac{1}{\alpha}\sum_i \boldsymbol{\ell}_i\boldsymbol{v}_i^T\right\| + 2\left\|\frac{1}{\alpha}\sum_i \boldsymbol{e}_i\boldsymbol{v}_i^T\right\| \\
&\leq c\sqrt{\eta}q^2 f\sqrt{\frac{r\log n}{\alpha}}\lambda^- + c\sqrt{\eta}q\frac{\lambda_v^+}{\lambda^-}\sqrt{\frac{r\log n}{\alpha}}\lambda^- + c\sqrt{\eta}qf\sqrt{\frac{r\log n}{\alpha}}\lambda^- + c\sqrt{\eta}q^2\frac{\lambda_v^+}{\lambda^-}\sqrt{\frac{r\log n}{\alpha}}\lambda^- \\
&\qquad + c\sqrt{\eta}q\frac{\lambda_v^+}{\lambda^-}\sqrt{\frac{r\log n}{\alpha}}\lambda^- \\
&\leq C\sqrt{\eta}qf\sqrt{\frac{r\log n}{\alpha}}\lambda^- := H(\alpha)\lambda^-
\end{aligned}$$

where the last line follows from using $q \leq 1$ and $\lambda_v^+ \leq \lambda^+$. The bound on $\|\frac{1}{\alpha}\sum_t \boldsymbol{a}\boldsymbol{a}^T - \boldsymbol{\Lambda}\|_2$ follows directly from the first item of Lemma C.2. This completes the proof. ⊠

Lemma C.2. *With probability at least* $1 - 10n^{-10}$*, if* $\alpha > r\log n$*, then,*

$$\begin{aligned}
&\left\|\frac{1}{\alpha}\sum_i \boldsymbol{a}_i\boldsymbol{a}_i^T - \boldsymbol{\Lambda}\right\| \leq c\eta f\sqrt{\frac{r\log n}{\alpha}}\lambda^- := G_{den}(\alpha)\lambda^-, \\
&\left\|\frac{1}{\alpha}\sum_i \boldsymbol{\ell}_i\boldsymbol{e}_i^T - \frac{1}{\alpha}\mathbb{E}\left[\sum_i \boldsymbol{\ell}_i\boldsymbol{e}_i^T\right]\right\|_2 \leq c\sqrt{\eta}qf\sqrt{\frac{r\log n}{\alpha}}\lambda^- := H(\alpha)\lambda^-, \\
&\left\|\frac{1}{\alpha}\sum_i \boldsymbol{e}_i\boldsymbol{e}_i^T - \frac{1}{\alpha}\mathbb{E}\left[\sum_i \boldsymbol{e}_i\boldsymbol{e}_i^T\right]\right\|_2 \leq c\sqrt{\eta}q^2 f\sqrt{\frac{r\log n}{\alpha}}\lambda^- := H(\alpha)q\lambda^- \\
&\left\|\frac{1}{\alpha}\sum_i \boldsymbol{v}_i\boldsymbol{v}_i^T - \frac{1}{\alpha}\mathbb{E}\left[\sum_i \boldsymbol{v}_i\boldsymbol{v}_i^T\right]\right\|_2 \leq c\sqrt{\eta}q\frac{\lambda_v^+}{\lambda^-}\sqrt{\frac{r\log n}{\alpha}}\lambda^- \\
&\left\|\frac{1}{\alpha}\sum_i \boldsymbol{e}_i\boldsymbol{v}_i^T\right\|_2 \leq c\sqrt{\eta}q^2\frac{\lambda_v^+}{\lambda^-}\sqrt{\frac{r\log n}{\alpha}}\lambda^- \\
&\left\|\frac{1}{\alpha}\sum_i \boldsymbol{\ell}_i\boldsymbol{v}_i^T\right\|_2 \leq c\sqrt{\eta}q\frac{\lambda_v^+}{\lambda^-}\sqrt{\frac{r\log n}{\alpha}}\lambda^-
\end{aligned}$$

Proof of Lemma C.2. 1. $\boldsymbol{a}_i\boldsymbol{a}_i^T$ *term* . Let $\tilde{\boldsymbol{Z}}_i := \boldsymbol{a}_i\boldsymbol{a}_i^T$ and we apply Theorem B.5 to $\boldsymbol{Z}_i = \tilde{\boldsymbol{Z}}_i - \mathbb{E}[\tilde{\boldsymbol{Z}}_i]$. with $s = \epsilon\alpha$. Now it is easy to see that $\|\boldsymbol{Z}_i\| \leq 2\|\boldsymbol{a}_i\boldsymbol{a}_i^T\| \leq 2\|\boldsymbol{a}_i\|_2^2 \leq 2\eta r\lambda^+ := R$ and similarly,

$$\frac{1}{\alpha}\|\sum_i \mathbb{E}[\boldsymbol{Z}_i^2]\| = \frac{1}{\alpha}\|\sum_i \mathbb{E}[\|\boldsymbol{a}_i\|_2^2\boldsymbol{a}_i\boldsymbol{a}_i^T]\| \leq \cdot\max_{\boldsymbol{a}_i}\|\boldsymbol{a}_i\|_2^2 \cdot \max_i \mathbb{E}[\boldsymbol{a}_i\boldsymbol{a}_i^T] \leq \eta r(\lambda^+)^2 := \sigma^2$$

and thus, w.p. at most $2r\exp\left(-c\min\left(\frac{\epsilon^2\alpha}{r(\lambda^+)^2}, \frac{\epsilon^2\alpha}{r\lambda^+\epsilon}\right)\right)$. Now we set $\epsilon = \epsilon_5\lambda^-$ with $\epsilon_5 = c\eta f\sqrt{\frac{r\log n}{\alpha}}$ so that with with probability at most $2n^{-10}$,

$$\left\|\frac{1}{\alpha}\sum_i(\boldsymbol{a}_i\boldsymbol{a}_i^T - \mathbb{E}\left[\boldsymbol{a}_i\boldsymbol{a}_i^T\right])\right\| \geq c\eta f\sqrt{\frac{r\log n}{\alpha}}\lambda^-$$

2. $\boldsymbol{\ell}_i\boldsymbol{e}_i^T$ *term.* Let $\boldsymbol{Z}_i := \boldsymbol{\ell}_i\boldsymbol{e}_i^T$. We apply this result to $\tilde{\boldsymbol{Z}}_i := \boldsymbol{Z}_i - \mathbb{E}[\boldsymbol{Z}_i]$. To get the values of R and σ^2 in a simple fashion, we use the facts that (i) if $\|\boldsymbol{Z}_i\|_2 \leq R_1$, then $\|\tilde{\boldsymbol{Z}}_i\| \leq 2R_1$; and (ii) $\sum_i \mathbb{E}[\tilde{\boldsymbol{Z}}_i\tilde{\boldsymbol{Z}}_i^T] \preccurlyeq \sum_i \mathbb{E}[\boldsymbol{Z}_i\boldsymbol{Z}_i^T]$. Thus, we can set R to two times the bound on $\|\boldsymbol{Z}_i\|_2$ and similary for σ^2

   It is easy to see that $R = 2\sqrt{\eta r\lambda^+}\sqrt{\eta rq^2\lambda^+} = 2\eta rq\lambda^+$. To get σ^2, observe that

$$\begin{aligned}\frac{1}{\alpha}\left\|\sum_i \mathbb{E}[\boldsymbol{e}_i\boldsymbol{\ell}_i^T\boldsymbol{\ell}_i\boldsymbol{e}_i^T]\right\|_2 &\leq (\max_{\boldsymbol{\ell}_i}\|\boldsymbol{\ell}_i\|^2)\cdot\max_i\|\mathbb{E}[\boldsymbol{e}_i\boldsymbol{e}_i^T]\| \\ &\leq \eta r\lambda^+\cdot q^2\lambda^+ = \eta rq^2(\lambda^+)^2.\end{aligned}$$

   Repeating the above steps, we get the same bound on $\|\sum_i \mathbb{E}[\boldsymbol{Z}_i\boldsymbol{Z}_i^T]\|_2$. Thus, $\sigma^2 = rq^2(\lambda^+)^2$. Thus, we conclude that,

$$\frac{1}{\alpha}\left\|\sum_i \boldsymbol{\ell}_i\boldsymbol{e}_i^T - \mathbb{E}[\sum_i \boldsymbol{\ell}_i\boldsymbol{e}_i^T]\right\|_2 \geq \epsilon \tag{12}$$

   w.p. at most $2n\exp\left(-c\min\left(\frac{\epsilon^2\alpha}{\eta rq^2(\lambda^+)^2}, \frac{\epsilon\alpha}{\eta rq\lambda^+}\right)\right)$. Set $\epsilon = \epsilon_0\lambda^-$ with $\epsilon_0 = cqf\sqrt{\frac{r\log n}{\alpha}}$ so that (12) hold w.p. at most $2n^{-10}$.

3. $\boldsymbol{e}_i\boldsymbol{e}_i^T$ *term.* We again apply matrix Bernstein and proceed as above. In this case, $R = 2\eta rq^2\lambda^+$ and $\sigma^2 = \eta rq^4(\lambda^+)^2$. Set $\epsilon = \epsilon_2\lambda^-$ with $\epsilon_2 = c\sqrt{\eta}q^2f\sqrt{\frac{r\log n}{\alpha}}$. Then again, the probability of the bad event is bounded by $2n^{-10}$.

4. $\boldsymbol{v}_i\boldsymbol{v}_i^T$ *term.* We again apply matrix Bernstein. In this case, $R = 2Cr\lambda_v^+$ and $\sigma^2 = 2Cr(\lambda_v^+)^2$. Set $\epsilon = \epsilon_2\lambda^-$ with $\epsilon_2 = c\sqrt{\eta}f\sqrt{\frac{r\log n}{\alpha}}$. Then again, the probability of the bad event is bounded by $2n^{-10}$.

5. $\boldsymbol{\ell}_i\boldsymbol{v}_i^T$, *and* $\boldsymbol{e}_i\boldsymbol{v}_i^T$ *terms.* We again apply matrix Bernstein as done before.

⊠