Large-scale Wireless Fingerprints Prediction for Cellular Network Positioning

Xinyu Wu, Xiaohua Tian, Xinbing Wang
School of Electronic Information and Electrical Engineering,
Shanghai Jiao Tong University, China.
Email: {wuxinyu,xtian,xwang8}@sjtu.edu.cn.

***Abstract*—Cellular network positioning is a mandatory requirement for localizing emergency callers, such as E911 in North America. Although smartphones are normally with GPS modules, there are still a large number of users with cell phones only as basic devices, and GPS could be ineffective in urban canyon environments. To this end, the fingerprinting positioning mechanism is incorporated into LTE architecture by 3GPP, where the major challenge is to collect geo-tagged wireless fingerprints in vast areas. This paper proposes to utilize the subspace identification approach for large-scale wireless fingerprints prediction. We formulate the problem into the problem of finding the optimal subspace over Stiefel manifold, and redesign the Stiefel-manifold optimization method with fast convergence rate. Moreover, we propose a sliding window mechanism for the practical large-scale fingerprints prediction scenario, where fingerprints are unevenly distributed in the vast area. Combining the two proposed mechanisms enables an efficient method of large-scale fingerprints prediction in the city level. Further, we validate our theoretical analysis and proposed mechanisms by conducting experiments with real mobile data, which shows that the resulted localization accuracy and reliability with our predicted fingerprints exceed the requirement of E911.**

I. Introduction

Cellular network positioning is mainly driven by the governments' mandatory requirement for operators to localize the caller in emergency situations, such as E911 in North America and E112 in Europe [1], [2], [3]. This is because most of the emergency callers (e.g., 60% in the Europe Union in 2013 [4], [5]) are unable to provide their current positions accurately. To this end, the 3rd Generation Partnership Project (3GPP) has made the positioning methods such as Cell ID (CID) mandatory since Release 8 [6], and specified the architecture of fingerprinting based positioning for LTE networks in Release 9 [7]. The positioning capability also can be leveraged for network planning, troubleshooting [13], and location based services such as event recommendation and location-aware advertising [2], [8], [14].

The past decades have witnessed a large body of work devoted to indoor positioning [9], [10], [11], [12], [20], where it is largely believed that the global navigation satellite system (GNSS) such as GPS has satisfied the need of localization in outdoor spaces; however, the fact is that solely relying on GNSS is unable to meet the positioning requirement of E911 or E112 even in outdoor spaces. First, a large number of mobile devices without GPS functionality still remain in use. It is found that more than 90% of Americans have cell phones, but the smartphone adoption level is only 64% in 2015 [24], and the level for the group of senior Americans (60+) is merely around 18% in 2014 [23]. Second, even for those smartphone users, it has been verified in the operator's practice that the performance of GPS is unacceptable in urban canyon environments. Some locations of such kind are unable to have a single satellite visible, and a notable of locations have less than 3 satellites visible, which is the basic requirement for GPS localization [2], [5], [8].

Although not 100% adopted, considerable widespread use of smartphones with the GPS module can facilitate fingerprinting based localization, which particularly suits cellular networks [13], [14], [16], [17], [18], [19], [21], [22]. LTE smartphones regularly report the user measurement data (UMD) to the database at the network edge in the network control and management process [2], [5], [6], [7], [13], [14], [16], where the RF measurements contained in the UMD such as the reference signal receiving power (RSRP) can be regarded as a kind of wireless fingerprint of the observed location. Leveraging the natural process, the network operator can construct a comprehensive and up-to-date fingerprints database in a crowdsourcing manner [13]. As the basic mobile device still needs to report the RSRP to the network periodically, then the device's current location can be estimated by comparing the reported RSRP with the fingerprints database.

However, the challenge for fingerprinting localization in the cellular network is that vast areas still need to be surveyed. The regular UMD contains no GPS location information if some special software is not installed in the user's mobile device [13], [14]; therefore, the war-driving method is still needed to geo-tag the UMD [14], [16]. The expensive war-driving process motivates the idea of fingerprints prediction based on the particular radio propagation model [16]; the mobile trajectory tracking method is also utilized to match a time series of the UMD to a route [14], [15], [17], so that the location information of the continuously-tracked UMD can be derived. While such interesting ideas can be helpful handling particular cases, a systematical approach to performing large-scale outdoor fingerprints prediction in the city level is still an open issue.

In this paper, we propose a large-scale fingerprints prediction approach to facilitate cellular network positioning, where we redesign the subspace identification mechanism to fully exploit the intrinsic connections among wireless fingerprints.

Our contributions are as following:

First, we formulate the fingerprints prediction problem into the problem of finding the optimal subspace over the Stiefel manifold [26], and propose a streamlined Stiefel-manifold optimization algorithm for fingerprints prediction. The d-dimensional Stiefel manifold is the set of all orthogonal d linearly independent vectors in the m-dimensional space, where each element in the set can span a d-dimensional subspace. The basic idea of the classic Stiefel-manifold optimization algorithm is similar to the gradient descent method, where the difference is that the decision variable of the former lies in the Stiefel manifold instead of the real number domain. We streamline the classic Stiefel-manifold optimization algorithm to accommodate the characteristics of fingerprints prediction (§**Section IV-A**), and prove convergence of the proposed algorithm (§**Section IV-B**); moreover, we reveal the fundamental reason why our design converges faster than an alternative approach to optimizing decision variables over Grassmann manifold [31], [32], [33].

Second, we propose a dynamic sliding window mechanism to deal with the practical fingerprints prediction scenario, where the fingerprints are unevenly distributed in the vast area. The proposed mechanism scans the entire area multiple times with a sliding window, where the window size increases in each new round of scanning as the predicted fingerprints obtained in the previous round increase the density of available fingerprints (§**Section V-A**). The mechanism is highly efficient with completing fingerprints prediction over $69.8km^2$ area within 7 rounds of scanning. The crux of the mechanism design is to determine the dimension of the subspace d for the matrix in the sliding window. We propose to sample a complete sub-matrix in each window matrix, and determine d through applying singular value decomposition (SVD) to the sub-matrix; we theoretically prove that the subspace dimension determination method incurs tractable information loss (§**Section V-B**).

Third, we validate our theoretical analysis and proposed mechanism with a real data set, which contains around $8,820,000$ RSRP data records collected from a 69.8 km^2 area in a city. Our experimental results show that the proposed scheme provides satisfactory accuracy of fingerprints prediction. We conduct positioning experiments with the predicted data, and the results show that the user can be localized in the $100m$ and $300m$ neighborhood of the real location at 70.8% and 98.67% respectively, which exceeds the E911 network based localization requirement regulated by the federal communication comission (FCC): "within $100m$ for 67% and $300m$ for 90%" (§**Section VI-C**). Moreover, the results verify that the proposed streamlined Stiefel-manifold optimization algorithm converges faster than the Grassmann manifold alternative (§**Section VI-D**).

Third, we validate our theoretical analysis and proposed mechanism with real data sets sampled in two cities. The two data sets contain RSRP data covering 2.2 km^2 and 67.3 km^2 areas in two cities with around $60,000$ and $8,820,000$ data records, respectively. Our experimental results show that the proposed scheme provides higher accuracy of fingerprints prediction compared with other frequently used methods (§**Section VI-A**). We conduct positioning experiments with the predicted data, and the results show that the user can be localized in the $100m$ and $300m$ neighborhood of the real location at 71% and 98% respectively, which exceeds the requirement of E911 regulated by the federal communication comission (FCC): "within $100m$ for 67% and $300m$ for 90%" (§**Section VI-B**). Moreover, the results verify that the proposed streamlined Stiefel-manifold optimization algorithm converges faster than the Grassmann manifold alternative (§**Section VI-B**).

II. Related Work

Multiple mechanisms are supported by the LTE network positioning architecture by 3GPP [6], [7] including CID, TOA, TDoA and fingerprinting, among which the fingerprinting approach draws much attention in the research community; because the CID performance is highly dependent on the density of the BS, and the information in the practical UMD can be insufficient for TOA and TDOA [13], [14], [16], [17], [18], [19], [21], [22].

The 3GPP specifies the architecture of fingerprinting based positioning for the long term evolution (LTE) networks in Release 9 [7], as illustrated in Fig. 1. The LTE positioning architecture contains three main elements: the location service client (LCS), the location server, and the target device. The target device is normally installed the service that reports measurements or location to the location server. The location server processes request from the LCS, which not only collects measurements and other location information from the device and eNodeBs (eNBs), but also provides assisting measurements information for the target device to estimate its position. The LCS could send the network initiated location request (NILR) for emergency positioning, where the network instructs the target device to provide the position with unsolicited assistance data; the LCS could also send the mobile terminated location request (MTLR) for positioning the target device, which contains privacy features and can be rejected by the target device. The target device also could actively send mobile originated location request (MOLR), which is relayed by the mobility management entity (MME) to the location server.

The positioning data can be carried by both the control plane and the data plane protocols, where the control plane positioning are mostly used in emergency services. The LTE positioning protocol (LPP) is used for point-to-point communication between a target device and a location server over both the control plane and data plane. The LPP Annex (LPPa) protocol is for communication between an eNB and a location server for control plane positioning, which also can assist user-plane positioning by querying eNBs for information and measurements. User plane positioning uses the data channel to exchange positioning information, which is enabled by SUPL 2,0 protocol. SUPL 2.0 provides a common user plane pipe service for all air interfaces including 2G, 3G and LTE.

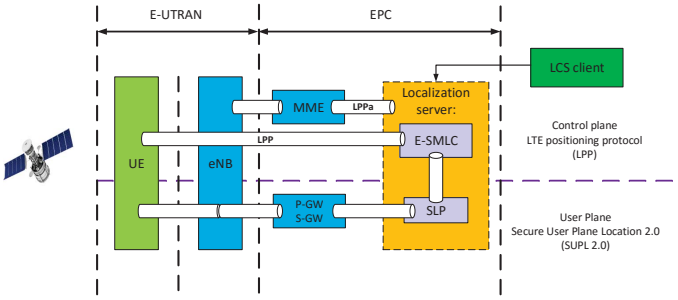


Fig. 1. Fingerprinting localization architecture in the LTE network.

The major challenge for fingerprinting positioning in the cellular network is to construct a wide-area radio map. Margolies *et al.* develop a fingerprinting based cellular network positioning testbed, where the radio map is constructed with crowdsourced data from 4 million unique users whose mobile devices are installed with a proprietary software [13]. Comprehensive evaluations are performed with the testbed, but there are still wide areas not covered by the crowd workers. Ray *et al.* utilize the user mobility to derive the location information of the continuously sampled UMD by matching the UMD time series to a physical route [14], where the predicted fingerprints are basically along the main roads of the city. The idea of utilizing the user mobility is also adopted by other work on cellular network measurement [15], [17]. Chakraborty *et al.* propose a geo-tag method based on Gaussian Mixture Model (GMM) [16], where the RF signal characteristics are modeled with a Gaussian distributed random variable.

Fingerprints prediction schemes based on matrix theory have been applied to indoor localization systems for saving the cost of site survey [9], [10], [11], [12], where the matrix completion algorithms are utilized [27], [28], [29], [30]. Although also adopting the matrix completion model, our work in this paper for the first time formulate the fingerprints prediction problem into a Stiefel manifold optimization problem to the best of our knowledge.

Edelman *et al.* present the framework to use the gradient descent method on the Grassmann and Stiefel manifold [26]. While the Grassmann manifold optimization problem is studied and applied in the field of image processing and remote sensing [31], [32], [33], the Stiefel manifold is under studied. Our work in this paper redesign the original Stiefel manifold optimization algorithm in [26] to accommodate the fingerprints prediction scenario, where the theoretical issues such as convergence rate analysis and step size design are resolved in contrast to [26].

III. Problem Formulation

A. Fingerprints Prediction: A Subspace Identification Perspective

The fingerprints prediction problem can be formulated into a matrix completion problem [9], [10], [11], [12]. The area needs localization service is first divided into grids, and the fingerprints sampled in grids are like elements in a matrix. The purpose of fingerprints prediction is essentially to complete the entire matrix by deriving the unknown elements based on those available ones. A number of mathematical tools for matrix completion are available, such as the singular value thresholding (SVT) [27], singular value partition (SVP) [28], forward-backward algorithm for matrix completion (FBMC) [29] and iterative reweighted least squares (sIRLSp) [30]. Though with different implementation details, those algorithms are generally based on singular value decomposition (SVD).

In particular, given an incomplete $m \times n$ matrix induced from the incomplete radio map, if we assign the values of all unknown elements to be 0s, then we have a complete radio map matrix $A = U\Lambda V^T$ after SVD, where U is an $m \times m$ real unitary matrix, Λ is an $m \times n$ rectangular diagonal matrix with non-negative real numbers on the diagonal and V^T is an $n \times n$ real unitary matrix. It is usually assumed that A is a low-rank matrix, which means that all the column vectors in A are linearly dependent to each other; this is based on the in-practice observation that fingerprints are correlated within a certain area. To exploit the linear dependency, we could keep d greatest singular values lying on the diagonal of Λ making it a $d \times d$ matrix, and make the corresponding parts in U and V^T $m \times d$ and $d \times n$ matrices, respectively. Then multiplying the three parts results in a new $m \times n$ matrix $\hat{A}$, which contains estimations to those unknown elements originally assigned values of 0s in A.

The essence of the SVD method is actually to find a lower dimensional subspace that contains all the column vector in $\hat{A}$. Consider an m-dimensional space that contains all m-dimensional column vectors in A, if most of those vectors are linearly dependent with each other, then most of them should belong to a lower dimensional subspace of the m-dimensional space. For example, imagine that there are some points in the 3-D space, if most of the points are linearly dependent with each other, then those points should be in a 2-D plane or a straight line. In SVD, the residual $m \times d$ matrix U is such a lower (d) dimensional subspace induced by the greatest d singular vectors. If such a lower dimensional subspace is found, any vector residing in the subspace are available, which is the fundamental reason the unknown elements can be estimated.

The accuracy of the elements estimation is highly dependent on whether the obtained subspace indeed contains most of the vectors in A. There are infinite number of possible subspaces that can be induced by A; however, the SVD method factually always finds one specific type of the subspace. This is because the input of the SVD method always assigns the values of unknown elements to be 0s. Assigning different values to those unknown elements results in different subspaces, but there are infinite number of possible situations, which makes SVD method unable to guarantee that the found subspace is always optimal. In the following, we are to show how to find the optimal subspace in the whole set of possible subspaces.

B. Problem Formulation

The fingerprints prediction problem can be formulated into the following matrix completion problem:

$$\begin{aligned} \min_{\Omega,\hat{A}} \quad & ||P_\Omega(A) - P_\Omega(\hat{A})||, \\ & s.t. |\Omega| \leq |\Omega_m|, \end{aligned} \tag{1}$$

where $||\cdot||$ represents any suitable norm and A is the matrix representing the radio map. Since some elements in A have not been measured thus unavailable, we use $P_\Omega(A)$ to denote those available fingerprints in A. The fingerprints prediction mechanism results in $\hat{A}$; this is a complete estimation of A, which contains estimations to those unmeasured fingerprints in corresponding postions. The physical meaning of the problem is to find $\hat{A}$ that minimizes the deviation from the available observations denoted by $P_\Omega(A)$, given the limited number of observations $|\Omega| \leq |\Omega_m|$.

The j-th column of A can be regarded as an m-dimensional vector denoted by a_j, and all the column vectors are in an m-dimensional space denoted by $\mathscr{M}$. Since a_js are correlated to each other, then we can assume that the matrix A is a rank-d matrix, which means that all vectors of A belong to a d-dimensional subspace of $\mathscr{M}$. However, as some elements in A are unknown, to obtain a prediction of those elements, we want to find a rank-d matrix A_d based on the known elements in each a_j. The A_d found must be a complete matrix and $A_d = U_d \Lambda_d V_d^T$ after SVD, where the matrix U_d contains d m-dimensional orthogonal vectors, which can span an d-dimensional subspace of $\mathscr{M}$. Finding A_d directly could be challenging, but due to the property of the low rank matrix, if we can find U_d, then A_d can be derived.

The problem is now transformed into a subspace identification problem in the following form:

$$\min_{\substack{U_d:m\times d \\ w_j:d\times 1}} \sum_{j=1}^{n} ||[U_d w_j]_\Omega - [a_j]_\Omega||_2^2, \tag{2}$$

where $U_d w_j$ represents the column vector in A_d corresponding to the column vector in A denoted by a_j. Since a_j could be incomplete, we use $[a_j]_\Omega$ to denote the available elements in a_j; the corresponding elements in A_d is denoted by $[U_d w_j]_\Omega$, as the matrix A_d can be transformed into $A_d = U_d W$ with $W = \Lambda_d V_d^T$, where w_j is the jth column of W.

The problem (2) distinguishes itself from other commonly seen optimization problems in that the decision variable U_d is a subspace in the form of a matrix. Recall the m-dimensional space where all vectors of A are in, all d-tuples of orthogonal m-dimensional vectors form a d-dimensional Stiefel manifold; therefore, the problem becomes to find a point in the Stiefel manifold considering the objective function in (2). This is a Stiefel manifold optimization problem [26].

It is worth mentioning that multiple points in the Stiefel manifold can possibly form the same subspace, because one subspace may have multiple sets of basis. All those lower-dimensional subspaces in the m-dimensional space form another kind of manifold known as Grassmann manifold [31], [32], [33], which is to be used for convergence proof in the discussion later. We will theoretically prove that the convergence rate of our proposed mechanism based on Stiefel manifold can be higher than performing optimization over the Grassmann manifold, which especially suits fingerprints prediction in extremely large-scale areas.

IV. Streamlined Stiefel Manifold Optimization

A. Algorithm Design

The basic idea of the Stiefel manifold optimization approach is similar to the gradient descent method, which is frequently used in resolving optimization problems. The gradient descent method starts with a given point on a curve representing the objective function, and iteratively takes steps proportional to the negative of the gradient of the function at the current point. The method can be extended to the Stiefel manifold optimization problem [26]; however, the practical scenario of fingerprints prediction problem can not fit in the general framework presented in [26]. Moreover, it is not mentioned in [26] whether the approach will converge, and how to choose important parameters to guarantee performance, which makes it necessary to streamline the existing Stiefel manifold optimization approach. We are to go through the classic Stiefel-manifold optimization approach and show the challenge to be confronted in resolving the fingerprints prediction problem, and then present the streamlined design of the approach for dealing with the challenges.

Challenge 1: Determine the direction of iteration. The first step in [26] is to determine the direction of iterations, which is realized by finding the Hessian matrix H with respect to the objective function $F = \sum_{j=1}^{n} ||[U_d w_j]_\Omega - [a_j]_\Omega||_2^2$, and then finding the inverse of H. However, we find that H for the fingerprints prediction problem may not have full rank, thus the H^{-1} may be unavailable. To this end, we propose to replace H^{-1} with the gradient of the objective function ∇F, where the intuition is that ∇F also represents a possible iteration direction.

Challenge 2: Complex objective function. The second step is to find the common iteration equation:

$$U_{d,t+1} = U_{d,t} M_t + Q N_t,$$

where Q satisfies the QR decomposition of $(I - U_{d,t} U_{d,t}^T)\nabla F$, and M_t and N_t satisfy

$$\begin{bmatrix} M_t \\ N_t \end{bmatrix} = \left(\exp\left(t \begin{bmatrix} U_{d,t}^T \nabla F & -R^T \\ R & 0 \end{bmatrix} \right) \right) \begin{bmatrix} I_d \\ 0 \end{bmatrix}, \tag{3}$$

where we replace H^{-1} with ∇F. However, it is noted that U_d and w_j are factually dependent to each other, since $A_d = U_d W$ with $W = \Lambda_d V_d^T$ and w_j is the jth column of W. Moreover, there is a matrix exponential function in the iteration, which we find could hinder finding the iteration equation due to the tedious form involving infinite matrix series.

We propose to replace the original objective function with

$$F(U_d) = \min_{x_j} ||[U_d x_j]_\Omega - [a_j]_\Omega||_2^2,$$

which decouples the dependence between U_d and w_j. It is straightforward that $w_j = x_j^* = argmin_{x_j}||[U_d x_j]_\Omega - [a_j]_\Omega||_2^2$. The new objective function is factually the item of the summation in the objective function in problem (2). A natural question is: will the solution with the new objective function be the same as that with the original objective function?

It is straightforward to verify that the second order derivative of $F(U_d)$ with respect to U_d is a semi-definite matrix, which means that the new objective function is convex thus definitely can achieve a unique minimum value. However, the solution for optimizing different items to make each item achieve the minimum value may be different. Recall the nature of A that the column vectors a_js of A fall in a d-dimensional subspace due to their correlation. In the process of optimizing each item, the solution must make the corresponding a_j fall in the same d-dimensional subspace. However, even if solutions of optimizing all those items can make those a_js fall in the subspace, they are not necessarily the same, because each solution is factually a set of basis of the d-dimensional subspace according to the definition of Stiefel manifold. It is possible that the solutions of optimizing different items vary, because the subspace can have multiple sets of basis. Nevertheless, the interesting point is that, since each solution is a set of basis of the same subspace, then any element vector in a given set of basis definitely can be represented by any other set of basis. This means that if an item achieves the minimum value, the corresponding solution can also make other items achieve the minimum value; therefore, the solution of the transformed problem is indeed the solution of the original problem.

Challenge 3: Obtain the common iteration equation. Based on the revision mentioned above, we now try to obtain the common iteration equation, which requires to perform QR decomposition to the matrix $(I - U_t U_t^T)\nabla F$ [1].

Since $\nabla F = -2r_t w_t^T - U_t(-2r_t w_t^T)^T U_t = -2r_t w_t^T$ with $r_t = P_\Omega(U_d w_j - a_j)$, we have $(I - U_t U_t^T)\nabla F = 2r_t w_t^T$ with $U_t^T r_t = 0$, where $w_t = x_j^*$ for the current vector a_j we are considering. This means that the rank of the matrix is 1 thus it is impossible to perform QR decomposition to the matrix. To deal with this challenge, we propose to replace QR decomposition with SVD, so that we can still obtain an orthonormal matrix Q. We relax the constraint in the classic approach and allow the matrix R to be singular, then we have $Q = [\frac{r_t}{||r_t||} \quad q_2 \quad q_3 \quad \cdots \quad q_n]$ and $R = [2||r_t||w_t \quad 0 \quad 0 \quad \cdots \quad 0]^T$, where $q_2, q_3, ...q_n$ are all orthonormal singular vectors and orthogonal to $\frac{r_t}{||r_t||}$[2]. Then we have $M_t = I_d$ and $N_t = \eta_t R$ with the consequent common iteration equation:

$$U_{t+1} = U_t + 2\eta_t \frac{r_t w_t^T}{||r_t||||w_t||}, \tag{4}$$

where η_t is the step size parameter.

[1]Note that we use F to denote $F(U_d)$ and U_t to denote $U_{d,t}$ respectively for the convenience of presentation, and such denotations are also to be adopted in the following discussions.

[2]Hereinafter we use $||\cdot||$ to represent $||\cdot||_2$, the 2-norm of a vector.

Challenge 4: Determine step size of iteration. Before executing the iteration, we must determine the step size first. Imagine the subspace identification process, we can first estimate a subspace U_t and see if the a_j we are considering is in the subspace. If a_j is not in U_t, the angle between the projection of a_j on U_t and a_j itself denoted by θ must be unequal to zero. Then we need to rotate the subspace to decrease θ to find a new estimation of the subspace. The degree of the rotation is factually the step size and it is straightforward that the step size should just make $\theta = 0$. Then we continue to rotate the previously estimated subspace U_t in the same way to make it contain other a_js, and the resulted U_t is the subspace we are finding. More precisely, we are trying to find the set of basis that spans a subspace containing all a_js according to the definition of Stiefel manifold.

Define a $d \times d$ matrix W_t in the t-th iteration, where $W_t = [\frac{w_t}{||w_t||} \quad C_t]$. Note that C_t is a $d \times (d-1)$ matrix, whose columns are unit vectors and all orthogonal to $\frac{w_t}{||w_t||}$. According to Gram-Schmidt transformation, C_t can be turned into a new matrix with all columns orthogonal to each other. Here we assume the columns of C_t are orthogonal to each other for the convenience of presentation. Multiply W_t in both sides of equation (4), we have $U_{t+1}W_t = U_t W_t + 2\eta_t \frac{r_t}{||r_t||}\begin{bmatrix}1 & 0 & 0 & \cdots & 0\end{bmatrix}$. The physical meaning of such operations is to perform rotation to the estimated subspace U_t, and the equation above can be further transformed into $U_{t+1}\frac{w_t}{||w_t||} = \frac{p_t}{||w_t||} + 2\eta_t \frac{r_t}{||r_t||}$, where $||p_t|| = ||w_t||$. Note that p_t is the projection of a_j on U_t, $\frac{r_t}{||r_t||}$ is a unit vector that is orthogonal to U_t; therefore, the degree U_t to be rotated is determined by η_t. We need to take an appropriate value of η_t so that a_j can fall in the resulted U_{t+1}, which results in the step size in iteration t:

$$\eta_t = \frac{1}{2}\frac{||r_t||}{||w_t||}. \tag{5}$$

After overcoming the challenges above, we present the streamlined Stiefel-manifold optimization algorithm (SSOA) as in Algorithm 1, where we use U_Ω to denote the rows of U whose index is in Ω.

B. Convergence Analysis

The challenge to prove the convergence of the proposed SSOA in Algorithm 1 is that the elements in each a_j are not completely known. In particular, recall that the problem (2) we study is to find the U_d that minimizes the objective function, we can claim Algorithm 1 converges if we can indeed find the U_d, which spans a subspace containing all vectors a_js in A, but a_j is incomplete itself.

We use $\beta_t(U, U_t) = 1 - \delta_t(U, U_t) = 1 - |U^T U_t|^2$ as the metric of measuring the distance between the estimated subspace in the t-th iteration U_t and its true value U [33], where $|U^T U_t|$ means the determinant of $U^T U_t$. Recall that we assume A is a rank-d matrix, where the subspace U contains all the vectors in A. According to the definition, if U and U_t span the same subspace, then $\beta_t(U, U_t) = 0$; if U and U_t are orthogonal to each other (any column in U is orthogonal to

Algorithm 1: Streamlined Stiefel-manifold optimization algorithm (SSOA)

Input:
An initial column-orthonormal $m \times d$ matrix U_0; sample set Ω, $m \times n$ sample matrix $P_\Omega(A)$; maximum number of iteration T.

Output:
Estimated matrix A_d.

$t = 0$;
while $t < T$ **do**
  Randomly choose a column index $q \in \{1, 2, ..., n\}$, get $[a_q]_\Omega$;
  $w_t = ([U_t]_\Omega^T [U_t]_\Omega)^{-1} [U_t]_\Omega [a_q]_\Omega$;
  $p_t = U_t w_t$;
  $r_t = P_\Omega(v_t - p_t)$;
  $U_{t+1} = U_t + \frac{r_t w_t^T}{||w_t||^2}$;
  $t = t + 1$;
end while
$U = U_t$;
for each $i \in \{1, 2, ..., n\}$ **do**
  $\hat{a}_i = U([U]_\Omega^T [U]_\Omega)^{-1} [U]_\Omega [a_i]_\Omega$;
end for
$A_d = [\hat{a}_1, \hat{a}_2, ..., \hat{a}_n]$.

all columns in U_t), then $\beta_t(U, U_t) = 1$ meaning that U and U_t have the largest distance.

With the metric, we first prove in Lemma 1 that SSOA converges at least as fast as Grassmann-manifold optimization algorithm with step size η_t. Then we prove in Theorem 1 that SSOA converges faster than Grassmann-manifold optimization algorithm with appropriately choice of η_t. Faster convergence rate is very meaningful especially for large-scale fingerprints prediction.

Lemma 1. *Let $\eta_t = \frac{||r_t||}{2||w_t||}$, then SSOA algorithm converges at least as fast as Grassmann-manifold based mechanism.*

Proof. Define $\bar{U}_t = U_t W_t = \begin{bmatrix} \frac{p_t}{||p_t||} & C_t \end{bmatrix}$ and $\check{U}_{t+1} = U_t W_t = \begin{bmatrix} \frac{q_t}{||w_t||} & C_t \end{bmatrix}$. Since the first column of $\check{U}_{t+1}$ is not a unit vector, we define $\bar{U}_{t+1} = \begin{bmatrix} \frac{q_t}{||q_t||} & C_t \end{bmatrix}$. With the step size η_t, we have $q_t = p_t + 2\eta_t \frac{||w_t||}{||r_t||} r_t = a_t$. Since W_t is an orthogonal matrix, then $R(\bar{U}_t) = R(U_t)$ and $R(\bar{U}_{t+1}) = R(U_{t+1})$, where $R(U)$ denotes the image of a matrix U, i.e., $R(U) = \{UA | \forall A \in \mathbb{R}^{m \times n}\}$.

Then we define $\bar{U} = UW_t = \begin{bmatrix} \frac{a_t}{||a_t||} & \bar{C} \end{bmatrix}$, where C is a $d \times (d-1)$ matrix with orthonormal columns, and these columns are all orthogonal to $\frac{a_t}{||a_t||}$. Hence $\bar{U}$ is an orthogonal matrix and $R(\bar{U}) = R(U)$. Then there exists an orthogonal matrix Y_t such that $\bar{U} = UY_t$, and $\frac{\delta_{t+1}}{\delta_t} = \frac{|\bar{U}_{t+1}^T \bar{U}|^2}{|\bar{U}_t{}^T \bar{U}|^2} = \frac{|U_{t+1}^T U|^2}{|U_t^T U|^2} = \frac{|C_t^T C|^2}{|C_t^T C|^2} (\frac{||p_t|| ||a_t||}{p_t^T a_t})^2 = \frac{||a_t||^2}{||p_t||^2}$ which is the convergence rate of the Grassmann-manifold optimization algorithm [33]. Note that p_t is the projection of a_t over the subspace U_t, thus the physical meaning of the equation above means that the distance between U_t and U decreases faster in each iteration. □

Theorem 1. *Denote $\sigma_i(A)$ as the i-th largest singular value of A and $\lambda_i(A)$ as the i-th largest eigenvalue of A. If we set the step size η_t such that*

$$\frac{\lambda_1(U_t^T U_t)}{\lambda_1(U_t^T U_t) + 4\eta_t^2} \frac{\lambda_d(U_t^T U_t)}{\lambda_2(U_{t+1}^T U_{t+1})} (1 + 2\eta_t \frac{||r_t||}{||p_t||})^2 > \frac{||a_t||^2}{||p_t||^2}, \tag{6}$$

then the convergence rate of the SSOA is strictly greater than $\frac{||a_t||^2}{||p_t||^2}$, which is known as the convergence rate of Grassmann-manifold optimization algorithm [33].

Proof. Note that U_{t+1} and U_t are not necessarily with orthonormal columns with the step size η_t now. We first apply the QR decomposition to U_{t+1} and U_t denoted by $U_{t+1} = U_{t+1}^Q R_{t+1}$ and $U_t = U_t^Q R_t$, respectively. Similar to the derivation in Lemma 1, we can derive

$$\frac{\delta_{t+1}}{\delta_t} = \frac{|(U_{t+1}^Q)^T U|^2}{|(U_t^Q)^T U|^2} = \frac{|R_{t+1}^{-1}|^2}{|R_t^{-1}|^2} \frac{((p_t + 2\eta_t \frac{||p_t||}{||r_t||} r_t)^T a_t)^2}{(p_t^T a_t)^2}. \tag{7}$$

Note that (i) U_{t+1} and R_{t+1} share the same singular values since multiplying an orthogonal matrix does not alter the singular values; (ii) R_{t+1} is a diagonal matrix, thus we have $|R_t^{-1}| = \frac{1}{\prod_{i=1}^d \sigma_i(U_t)}$ and $|R_{t+1}^{-1}| = \frac{1}{\prod_{i=1}^d \sigma_i(U_{t+1})}$ where

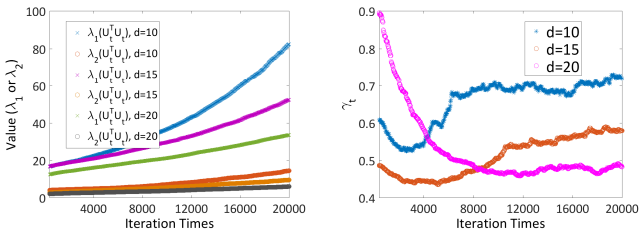
$$\sigma_i(U_{t+1}) = \sqrt{\lambda_i(U_{t+1}^T U_{t+1})} = \sqrt{\lambda_i(U_t^T U_t + 4\eta_t^2 \frac{w_t w_t^T}{||w_t||^2})},$$

based on the fact that $U_t^T r_t = 0$. Since $w_t w_t^T$ is a rank-1 matrix, with the only non-zero eigenvalue $||w_t||^2$, therefore according to Weyl's inequality [36], we have

$$\lambda_i(U_t^T U_t + 4\eta_t^2 \frac{w_t w_t^T}{||w_t||^2}) \geq \begin{cases} \lambda_1(U_t^T U_t) + 4\eta_t^2, & i = 1; \\ \lambda_{i-1}(U_t^T U_t), & i \geq 2. \end{cases} \tag{8}$$

With Eqn. (7) and Inequality (8), it is easy to derive $\frac{\delta_{t+1}}{\delta_t} \geq \frac{\lambda_1(U_t^T U_t)}{\lambda_1(U_t^T U_t) + 4\eta_t^2} \frac{\lambda_d(U_t^T U_t)}{\lambda_2(U_{t+1}^T U_{t+1})} (1 + 2\eta_t \frac{||r_t||}{||p_t||})^2$. Since RHS of the inequality is greater than $\frac{||a_t||^2}{||p_t||^2}$, which is the convergence rate of Grassmann-manifold optimization algorithm as presented in [33], the convergence rate of Stiefel-manifold optimization algorithm is faster. Proved. □

Theorem 1 presents the general condition where SSOA outperforms Grassmann-manifold optimization algorithm in convergence rate. In fact, based on experiments on the real big datasets in Section VI-B we find that (i) $\lambda_1(U_t^T U_t)$ surges rapidly and becomes much greater than $\lambda_2(U_{t+1}^T U_{t+1})$ and $\lambda_d(U_t^T U_t)$; (ii) $\lambda_d(U_t^T U_t)/\lambda_2(U_{t+1}^T U_{t+1}) \in [c, 1]$ with the constant $c > 0$. The verifications are illustrated in Fig. 2(a) and Fig. 2(b). Fig. 2(a) verifies observation (i) since under different dimension of subspaces ($d = 10, 15, 20$), we find that $\lambda_1(U_t^T U_t)$ is always greater than $\lambda_2(U_t^T U_t)$, and as more iterations are conducted, the gap becomes greater. Fig. 2(b) confirms observation (ii) since under $d = 10, 15, 20$, the ratio



(a) $\lambda_1(U_t^TU_t)$ vs $\lambda_2(U_t^TU_t)$ (b) $\gamma_t = \frac{\sigma_d(U_t^TU_t)}{\sigma_2(U_{t+1}^TU_{t+1})}$

Fig. 2. Experimental verifications of observations about Theorem 1.

$\lambda_d(U_t^TU_t)/\lambda_2(U_{t+1}^TU_{t+1})$ is always less than 1 and in our situation we can set $c = 0.45$. Meanwhile, combining both figures we verify $\lambda_1(U_t^TU_t)$ is always greater than $\lambda_d(U_t^TU_t)$.

Therefore we present a more practical condition in Proposition 1 that can facilitate determining the step size in practice, which is an approximation to the general Theorem 1.

Proposition 1. *We set* $\gamma_t = \frac{\lambda_d(U_t^TU_t)}{\lambda_2(U_{t+1}^TU_{t+1})} \in [c, 1]$. *If we choose the step size* η_t *such that* $\frac{1}{2}\frac{||a_t||-||p_t||}{\sqrt{c}||r_t||} < \eta_t << \frac{1}{2}\sigma_1(U_t)$, *then the convergence rate of the SSOA is strictly higher than that of Grassmann-manifold optimization algorithm.*

Proof. Since $\lambda_1(U_t^TU_t) >> \lambda_2(U_{t+1}^TU_{t+1}) = \gamma_t\lambda_d(U_t^TU_t)$, we can approximate Eqn. (6) in Theorem 1 as $\gamma_t(1 + 2\eta_t\frac{||r_t||}{||p_t||})^2 > \frac{||a_t||^2}{||p_t||^2}$. Then we can easily obtain

$$\eta_t > \frac{1}{2}\frac{||a_t|| - ||p_t||}{\sqrt{c}||r_t||}.$$

On the other hand, since $\sigma_1(U_t) = \sqrt{\lambda_1(U_t^TU_t)}$we need to ensure that $\eta_t << \frac{1}{2}\sigma_1(U_t)$ to let $\frac{\lambda_1(U_t^TU_t)}{\lambda_1(U_t^TU_t)+4\eta_t^2} \to 1$. Thus when $\frac{1}{2}\frac{||a_t||-||p_t||}{\sqrt{c}||r_t||} < \eta_t << \frac{1}{2}\sigma_1(U_t)$, the convergence rate of SSOA is higher than that of Grassmann-manifold optimization algorithm. □

Note that sometimes when $||r_t||$ is small enough, especially when the algorithm is close to convergence, $\frac{||a_t||-||p_t||}{\sqrt{c}||r_t||} > \lambda_1(U_t^TU_t)$. However, in this situation we just use $\eta_t = \frac{1}{2}\frac{||r_t||}{||w_t||}$. Combining the ways above to determine step sizes, we will show that the convergence speed of SSOA outperforms significantly than that of Grassmann-manifold based algorithm under real big datasets in Section VI-B.

C. Discussions

The fundamental reason that the Stiefel-manifold optimization approach converges faster than the Grassmann-manifold counterpart is that the physical meaning of a point on the Stiefel-manifold is a set of basis of a d-dimensional subspace and that on the Grassmann-manifold is a d-dimensional subspace itself. It is supposed that our optimization algorithm should measure the distance between a_j and the estimated set of basis in each iteration; however, such kind of iteration will definitely incur high computational complexity due to the finer-granularity of distance metric. Since a set of basis can span a subspace, we use the angle between a_j and its projection on the subspace as the distance metric in our algorithm design. In this case, the solution of our algorithm is factually the subspace spanned by the set of basis, instead of the particular set of basis itself the traditional Stiefel-manifold optimization mechanism is finding. This is because a subspace can have multiple sets of basis.

However, this means that there may exist multiple solutions can be obtained by our proposed SSOA, and any one of the solutions can satisfy the requirement. Recall that we transformed the objective function in the algorithm design (Section 3.1), and the new objective function is convex, then the situation of optimizing over the Stiefel manifold is like that illustrated in the right part of the Fig.3, and the left part of the figure shows the situation of optimizing over the Grassmann manifold. There could be multiple solutions on the Stiefel manifold, but only one solution on the Grassmann manifold, because a d-dimensional subspace is just regarded as a point on the Grassmann manifold according to the definition. Consequently, it is easier to find a solution on the Stiefel manifold and the convergence rate is higher.

We present numerical analysis results in Fig. 4 showing the average number of iterations for completing 500 randomly generated matrix on the Stiefel and Grassmann manifold, respectively. The two sub-figures show the number of iterations needed to achieve different levels of prediction accuracy (average error) for the cases that 40% and 60% of the real data are available. It is clear that the proposed SSOA requires less numbers of iterations compared with the Grassmann manifold optimization algorithm. The experimental results with the real data to be presented in Section V-C also corroborates our analysis.

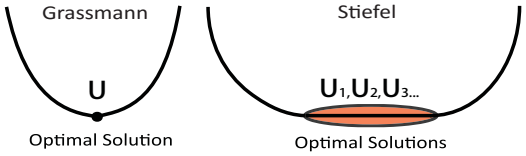


Fig. 3. Convergence over Stiefel and Grassmann manifold.

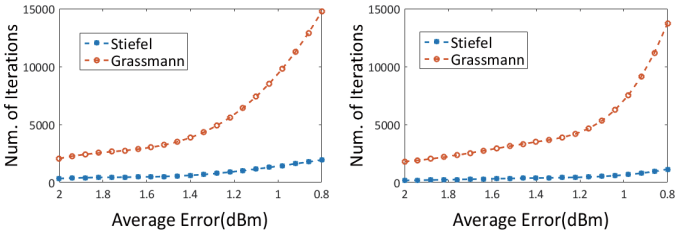


Fig. 4. Numerical analysis to the convergence rate (Left: 40%; Right: 60%).

V. Fingerprints Prediction with Sliding Window

A. Sliding Window Mechanism Design

In practice, the sampled wireless fingerprints for cellular networks are unevenly distributed in the extremely vast area, as to be shown in Section V. It is impossible to complete the matrix of the entire area in one shot, since the sampled data are sparsely distributed in some subareas. To deal with this issue,

we can create a sliding window to scan the entire area in a row-by-row manner, where the crux is to determine the size of the window. In particular, we first grid the entire area into square cells, where the edge length of each cell is dependent on the accuracy requirement (normally in tens of meters). We then let the sliding window cover a number of such cells and move from left to right and top to bottom so that the entire area can be scanned. The sliding window's movement step size in both horizontal and vertical direction is randomly assigned, so that the predicted fingerprints obtained in the window's previous location can be utilized to predict the fingerprints in the current area covered by the window.

Moreover, the window size must be set small enough initially to make sure there are enough amount of data available within the window for prediction. After a round of scanning, we have predicted fingerprints available in some cells, which can be regarded as the training data thus the density of available fingerprints increases. Then we could enlarge the window and scan the entire area again still in a row-by-row manner. In this way, we scan the area multiple times, so that fingerprints in most of the area can be predicted.

The crux of the sliding window design is to determine the window size, which is essentially to determine the dimension of the subspace d. This is because the length and width of the sliding window must be greater than d, or it is impossible to predict the fingerprints in the unsampled cells. However, the challenge is that the matrix corresponding to the sliding window itself is incomplete. To deal with the challenge, we propose to determine d of the sliding window matrix by sampling a complete sub-matrix within the window. The rationale is that if the vectors in the window matrix are correlated, the correlation should be reflected by any sub-matrix within. The question is how good we can predict the fingerprints in the window matrix if we determine d in this way.

The corner stone assumption of the subspace identification approach is that all the vectors in the window matrix denoted by A is in the same subspace denoted by U_d; however, although the fingerprints are correlated, some vectors indeed are not in U_d thus incur prediction errors. Note that a large-valued d can decrease such error since more vectors can be included in U_d, but requires more elements in A to be available; in contrast, a small-valued d can increase the error, but it can accommodate more-sparsely-sampled A.

In particular, $A = U\Lambda V^T$ after SVD, and we use $\sigma_i = \Lambda_{ii}, i = 1, 2, ..., m$ (set $m < n$) to denote the i-th largest singular value of A. Suppose we had known d, then we approximate A with $A_d = U_d\Lambda_d V_d^T$, where U_d and V_d are the first d columns of U and V respectively, and Λ_d is the sub-matrix comprised of the first d columns and rows of Λ. We use $l_d = \sqrt{\frac{\sum_{i=1}^{d}\sigma_i^2}{\sum_{i=1}^{m}\sigma_i^2}}$ to denote the remained information after the approximation which only includes the largest d singular values in Λ. The rationale is that most of the information of the matrix lies in the largest singular values of the matrix after SVD. It is straightforward that a greater d leads to a greater l_d.

In the following discussion we will theoretically prove that the gap between the remained information of the complete sub-matrix we sample within the window, $\tilde{l}_d$, and that of the whole sliding window matrix, l_d, is small enough when the correlation of the columns of the sliding window matrix is strong. This validates that we can determine d of the sliding window matrix by conducting SVD on the complete sub-matrix within the window, with similar remained information.

B. Remained Information Analysis

We first present some lemmas to prepare for Theorem 2, the remained information analysis.

Lemma 2. *If real numbers $a, b, c, d > 0$, $b << a$, $d << c$, $|a - c| \le \epsilon_1$ and $|b - d| \le \epsilon_2$, then we have*

$$|\frac{b}{a} - \frac{d}{c}| \le \max\{\frac{\min\{b, d\}}{ac}\epsilon_2, \frac{\epsilon_1}{\max\{a, c\}}\}.$$

Proof. This lemma can be proved in a case-by-case manner as following:

Case 1: If $\frac{b}{a} \ge \frac{d}{c}$ and $a \ge c$ (note that now $b \ge d$), then $|\frac{b}{a} - \frac{d}{c}| = \frac{b}{a} - \frac{d}{c} \le \frac{|b-d|}{a} \le \frac{\epsilon_1}{a}$;

Case 2: If $\frac{b}{a} \ge \frac{d}{c}$, $a < c$ and $b < d$, then $|\frac{b}{a} - \frac{d}{c}| = \frac{bc-ad}{ac} \le \frac{d}{ac}|c - a| \le \frac{d}{ac}\epsilon_2$;

Case 3: If $\frac{b}{a} \ge \frac{d}{c}$ and $a < c$, $b \ge d$, then $|\frac{b}{a} - \frac{d}{c}| = \frac{bc-ad}{ac} \le \frac{b}{ac}|c - a| \le \frac{b}{ac}\epsilon_2$;

Similarly we can conduct the other 3 cases when $\frac{b}{a} < \frac{d}{c}$. Combining all the 6 cases, we finish the proof. □

Lemma 3. [The Hoffman-Wielandt inequality] *Given two $k \times k$ real symmetric matrices X and Y, if the eigenvalues of X are $\mu_1,\mu_2,...,\mu_k$ and that of Y are $\nu_1,\nu_2,...,\nu_k$, with $\mu_1 \ge \mu_2 \ge ... \ge \mu_k \ge 0$, $\nu_1 \ge \nu_2 \ge ... \ge \nu_k \ge 0$, then we have $\sum_{i=1}^{k}(\mu_i - \nu_i)^2 \le ||X - Y||_F^2$.*

Lemma 4. *[34] If U is an orthonormal matrix and Ω is a random set of row indexes. Then there exists a positive constant C, such that $\boldsymbol{E}||\frac{m}{|\Omega|}U_\Omega^T U_\Omega - I_n|| \le Cm\sqrt{\frac{\log n}{|\Omega|}}\max_{1\le k\le m}||u^k|| \le 1$, where u^k denotes the k-th row of U.*

Lemma 5. *If $\hat{A} = \tilde{A}V_p^T$, where V_p is an $n \times t$ matrix with orthonormal columns. Then there exists a constant $C > 0$, such that $\boldsymbol{E}||\frac{m}{|\Omega|}\hat{A}^T\hat{A} - A^T A|| \le Cm\sqrt{\frac{\log n}{|\Omega|}}\max_{1\le k\le m}$ $||u^k||||\Lambda||^2 \le |||\Lambda||^2$*

Proof. Note that $A = U\Lambda V^T$ after SVD, and $\hat{A}$ is a sub-matrix of A concatenated by the rows whose indices belonging

to Ω. Then we have $\hat{A} = U_\Omega \Lambda V^T$, and

$$\begin{aligned}
&\mathbf{E}||\frac{m}{|\Omega|}\hat{A}^T\hat{A} - A^TA|| \\
&= \mathbf{E}||\frac{m}{|\Omega|}V\Lambda U_\Omega^T U_\Omega \Lambda V^T - \frac{1}{|\Omega|}V\Lambda U^T U \Lambda V^T|| \\
&= \mathbf{E}||V\Lambda(\frac{m}{|\Omega|}U_\Omega^T U_\Omega - I_m)\Lambda V^T|| \\
&\leq \mathbf{E}||\frac{m}{|\Omega|}U_\Omega^T U_\Omega - I_m||||\Lambda||^2 \\
&\leq Cm\sqrt{\frac{\log n}{|\Omega|}}\max_{1\leq k\leq m}||u^k||||\Lambda||^2.
\end{aligned} \tag{9}$$

The last inequality holds based on Lemma 4. □

Theorem 2. *We use l_d and $\tilde{l}_d$ to denote the remained information of the incomplete $p \times q$ window matrix A and a complete $s \times t$ sub-matrix $\tilde{A}$ within, where we set $p \leq q$ and $s \leq t$, and d is the dimension of the subspace obtained by performing SVD to the sub-matrix; if the linear correlation of fingerprints is strong enough in the window matrix A, then $|l_d - \tilde{l}_d| \to 0$.*

Proof. Denote the following SVDs: $A = U\Lambda V^T$, $\tilde{A} = \tilde{U}\tilde{\Lambda}\tilde{V}^T$, $A_d = U_d\Lambda_d V_d^T$ and $\tilde{A}_d = \tilde{U}_d\tilde{\Lambda}_d\tilde{V}_d^T$. The window matrix A is highly linearly correlated, which means that almost all the information is contained within the subspace spanned by several principal axes while the other ones can be reasonably neglected. Then we have $\frac{\sum_{i=d+1}^{p}\sigma_i^2}{\sum_{i=1}^{p}\sigma_i^2} \to 0$. Based on Taylor's expansion, we obtain

$$l_d = \sqrt{1 - \frac{\sum_{i=d+1}^{p}\sigma_i^2}{\sum_{i=1}^{p}\sigma_i^2}} \approx 1 - \frac{1}{2}\frac{\sum_{i=d+1}^{p}\sigma_i^2}{\sum_{i=1}^{p}\sigma_i^2} \approx 1 - \frac{1}{2}\frac{||\Lambda - \Lambda_d||_F^2}{||\Lambda||_F^2}.$$

Similarly, $\tilde{l}_d \approx 1 - \frac{1}{2}\frac{||\tilde{\Lambda} - \tilde{\Lambda}_d||_F^2}{||\tilde{\Lambda}||_F^2}$, thus according to Lemma 2, we obtain

$$\begin{aligned}
|l_d - \tilde{l_d}| &= \frac{1}{2}|\frac{||\Lambda - \Lambda_d||_F^2}{||\Lambda||_F^2} - \frac{||\tilde{\Lambda} - \tilde{\Lambda_d}||_F^2}{||\tilde{\Lambda}||_F^2}| \\
&= \frac{1}{2}|\frac{||\Lambda - \Lambda_d||_F^2}{||\Lambda||_F^2} - \frac{\frac{p}{s}||\tilde{\Lambda} - \tilde{\Lambda}_d||_F^2}{\frac{p}{s}||\tilde{\Lambda}||_F^2}| \leq \frac{1}{2}\max\{S, T\},
\end{aligned} \tag{10}$$

where

$$S = \frac{\min\{||\Lambda - \Lambda_d||_F^2, \frac{p}{s}||\tilde{\Lambda} - \tilde{\Lambda_d}||_F^2\}}{\frac{p}{s}||\tilde{\Lambda}||_F^2||\Lambda||_F^2}|||\Lambda||_F^2 - \frac{p}{s}||\tilde{\Lambda}||_F^2|,$$

$$T = \frac{1}{\max\{||\Lambda||_F^2, \frac{p}{s}||\tilde{\Lambda}||_F^2\}}|||\Lambda - \Lambda_d||_F^2 - \frac{p}{s}||\tilde{\Lambda} - \tilde{\Lambda}_d||_F^2|.$$

For S, set $\breve{A} = \tilde{A}V_p^T$, where V_p is an $q \times t$ matrix with orthonormal columns and $\breve{A}$ is an $s \times q$ matrix. Let $Z = \frac{p}{s}\breve{A}^T\breve{A} - A^TA$, which is an $q \times q$ matrix. Note that (i) $|||\Lambda||_F^2 - \frac{p}{s}||\tilde{\Lambda}||_F^2| = |\sum_{i=1}^{p}\sigma_i^2 - \frac{p}{s}\sum_{i=1}^{s}\tilde{\sigma}_i^2|$; (ii) $\frac{p}{s}\breve{A}^T\breve{A}$ and A^TA are both symmetric matrices; (iii) $\breve{A}$ and $\tilde{A}$ share the same singular values. Thus according to Lemma 3 and 5,

$$\begin{aligned}
||Z||_F^2 &= \sum_{i=1}^{p}\lambda_i^2(Z) \geq \sum_{i=1}^{p}(\lambda_i(A^TA) - \frac{p}{s}\lambda_i(\breve{A}^T\breve{A}))^2 \\
&\geq \frac{1}{p}(\sum_{i=1}^{p}\sigma_i^2 - \frac{p}{s}\sum_{i=1}^{s}\tilde{\sigma}_i^2)^2 = \frac{1}{p}|||\Lambda||_F^2 - \frac{p}{s}||\tilde{\Lambda}||_F^2|^2.
\end{aligned}$$

Then $S \leq \frac{\min\{||\Lambda - \Lambda_d||_F^2, \frac{p}{s}||\tilde{\Lambda} - \tilde{\Lambda}_d||_F^2\}}{\frac{p}{s}||\tilde{\Lambda}||_F^2}\sqrt{n}$.

Now we focus on T. If $||\Lambda||_F^2 \geq \frac{p}{s}||\tilde{\Lambda}||_F^2$, then

$$T \leq \frac{||\Lambda - \Lambda_d||_F^2}{||\Lambda||_F^2} + \frac{p}{s}\frac{||\tilde{\Lambda} - \tilde{\Lambda}_d||_F^2}{||\Lambda||_F^2} \leq \frac{||\Lambda - \Lambda_d||_F^2}{||\Lambda||_F^2} + \frac{||\tilde{\Lambda} - \tilde{\Lambda}_d||_F^2}{||\tilde{\Lambda}||_F^2}. \tag{11}$$

Similarly, if $||\Lambda||_F^2 < \frac{p}{s}||\tilde{\Lambda}||_F^2$, we also have Inequality (11). Then if the linear correlation of fingerprints in A is strong, $||\Lambda - \Lambda_d||_F^2$ and $||\tilde{\Lambda} - \tilde{\Lambda}_d||_F^2$ approach zero, which makes both S and T approach zero. According to Eqn. (10), we prove that $|l_d - \tilde{l_d}| \to 0$, which means that using $\hat{A}$ to estimate the subspace dimension d for A does not incur much deviation in remained information (information loss). □

VI. Experimental Results

We do experiments with real data sampled by a network operator in two cities, where the data sets are sampled within 48 hours, covering 2.2 km^2 and 69.8 km^2 areas in the two cities and containing around $60,000$ and $8,820,000$ data records, respectively. Each data record contains the location information in terms of latitude and longitude, the corresponding RSRP, the time the measurement is performed, and some other irrelevant parameters.

We will first show the results with the smaller data set, in order to verify that the proposed subspace identification approach outperforms other frequently used matrix completion algorithms [27], [28], [29], [30]. Then we will show the results with the larger data set to examine the performance of our proposed SSOA and dynamic sliding window mechanism in fingerprints prediction in the large-scale scenario. To verify our prediction mechanism, we perform localizations with the predicted fingerprints and see if the accuracy and reliability can meet the requirement of E911, compared with Cell ID (CID) mechanism and Gaussian Mixture Model (GMM) based method [16]. Additionally, we validate our theoretical result that the convergence rate of our proposed SSOA outstrips the Grassmann approach with similar methodology by the larger data set.

A. Experiments on Small Data Set

Overview of Fingerprints Prediction Results: We first illustrate the fingerprints prediction results obtained by our proposed mechanism and then compare our mechanism with others in terms of different metrics. The top sub-figure in the first column of Fig. 5 shows the data sampled on the main road of the target area. This data set also contains fingerprints obtained from branches of those main roads. We use the proposed mechanisms to predict fingerprints on those

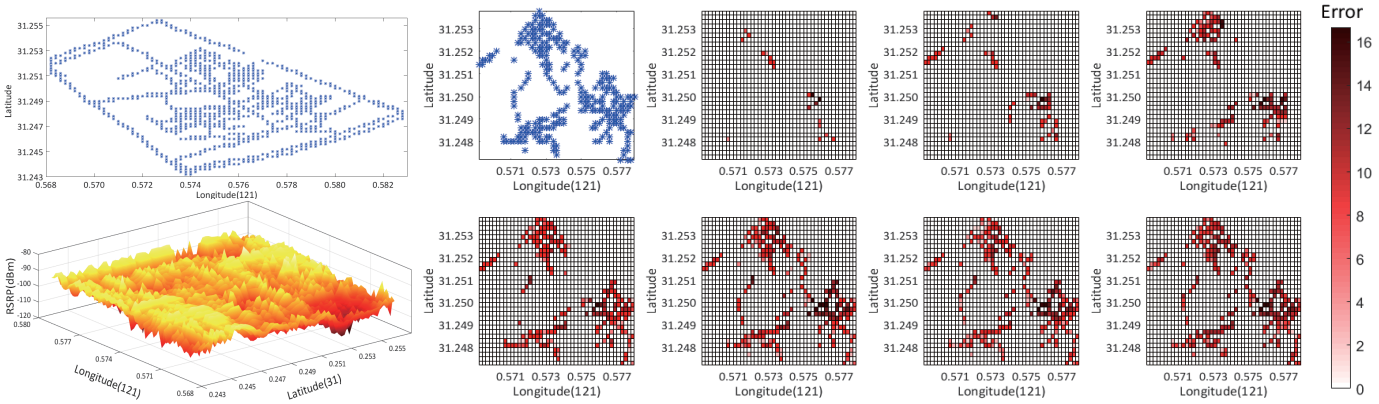


Fig. 5. Completion process with small data set.

branches, and compare the predictions with the real data value. The distribution of RSRP fingerprints is shown in the bottom sub-figure in the first column. The rest of the sub-figures illustrate the process of matrix completion with our proposed SSOA combined with the dynamic sliding window scanning. The top sub-figure in the second column shows the locations on those branches of the main roads where real data are sampled. The following sub-figures show the distribution of the completion errors in dBm after each round of scanning with the sliding window. It can be seen from the results that the fingerprints can be predicted, and the average error and standard deviation are $5.1dBm$ and $3.5dBm$ respectively.

Algorithms in Comparison: We compare the performance of the SSOA with that of the following matrix completion methods: *Singular value thresholding (SVT)* is a variant of SVD, where there is a threshold τ to determine the rank of the rectangular diagonal matrix; *Singular value partition (SVP)* is another variant of SVD, which partitions available observations into subsets and perform completion on each subset; *Forward-backward algorithm for matrix completion(FBMC)* scheme formulates the matrix completion problem into a convex optimization problem, with minimizing the objective function that is a combination of completion error and the rank of the estimated matrix; *Iterative reweighted least squares(sIRLSp)* is a family of algorithms, which conducts a least square minimization problem with the decision variable as a matrix.

Performance Metrics: To evaluate the performance of each mechanism, we use a subset of the whole data set to form the training set, based on which the fingerprints in other areas will be predicted, and the rest of the data form a test set denoted by Ω_{test}, where the data in the set are the ground truth for testing the predicted fingerprints. All the predicted fingerprints form a set denoted by Ω^c, but note that Ω_{test} and Ω^c are not necessarily the same, since we may predict fingerprints of some area that is not surveyed by the technicians. The performance of the mechanisms mentioned above is evaluated with the following metrics.

Error expectation defined as

$$ErExp(A) = \frac{\sum_{(i,j)\in\Omega^c\cap\Omega_{test}} |\hat{A}(i,j) - A(i,j)|}{|\Omega^c \cap \Omega_{test}|};$$

Error standard deviation defined as

$$ErStd(A) = \sqrt{\frac{\sum_{(i,j)\in\Omega^c\cap\Omega_{test}} (|\hat{A}(i,j) - A(i,j)| - ErExp(A))^2}{|\Omega^c \cap \Omega_{test}|}};$$

Normalized square of error defined as

$$NSE(A) = \frac{\sum_{(i,j)\in\Omega^c\cap\Omega_{test}} |\hat{A}(i,j) - A(i,j)|^2}{\sum_{(i,j)\in\Omega^c\cap\Omega_{test}} |\hat{A}(i,j)|^2};$$

In-threshold ratio defined as

$$ITR(A) = \frac{|\Omega^c \cap \Omega_{test} \cap \Omega_\xi|}{|\Omega^c \cap \Omega_{test}|}$$

with $\Omega_\xi = \{(i,j)||\hat{A}(i,j) - A(i,j)| < \xi\}$ (We set $\xi = 8$ in our experiments).

Prediction Performance 1: Fig. 6 shows the performance under different proportions of real data in the test set Ω_{test}, which is termed as the sampling rate. If we just use the data obtained from the main road to perform prediction, the sampling rate is 0, if we take 50% of the data in Ω_{test} out also as the training set then the sampling rate is 50%. It is straightforward that the subspace identification approach no matter over the Stiefel manifold or the Grassmann manifold outperforms other mechanisms, except for the $ErStd$ metric. Although the subspace identification approach results in a higher standard deviation, the expectation of the errors *ErExp(A)* is smaller than that of others. This is because the real data set itself fluctuates dramatically, but the results by other mechanisms are unable to reflect such variation.

Prediction Performance 2: Fig. 7 shows the performance under different ways of griding. We divide the region into cells with different edge length, which is termed as resolution. Based on the operator's localization accuracy requirement, the target area may be divided into grids with different resolutions, which in essence is to change the size of the matrix A. In Fig. 7, the resolution 5000 means that the target area is divided into

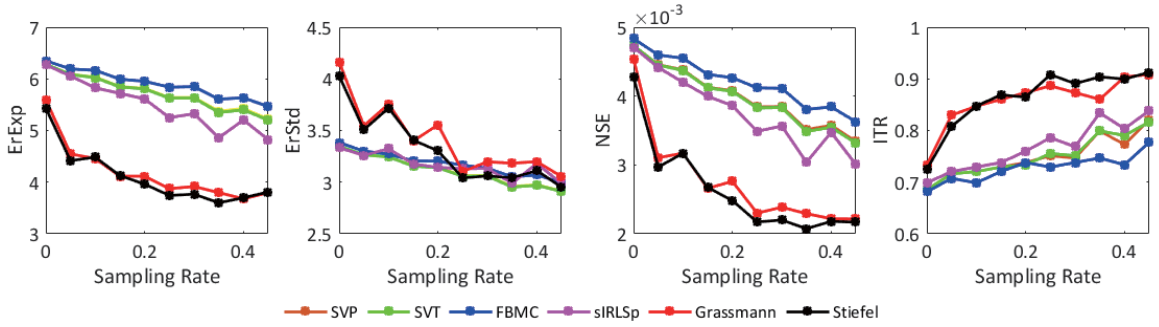


Fig. 6. Metrics with different sampling rates on branch roads.

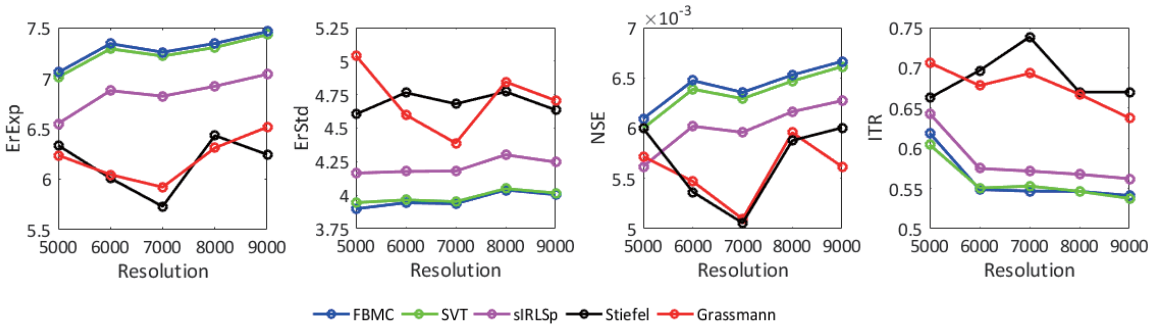


Fig. 7. Metrics with different resolutions of the area.

5000 equally-sized small cells over each edge. We only use the main-road data as the training set to predict fingerprints in the branch roads. The results in Fig. 7 shows that our proposed mechanisms work well under different ways of griding.

Remark: Note that both Stiefel-manifold and Grassmann-manifold based mechanism perform well, with faint difference in performance metrics above. However, we will show the significantly higher efficiency of Stiefel-manifold based mechanism in larger data set, which coincides with its faster convergence rate shown in Theorem 2 and promises its higher practicality in large-scale fingerprints prediction than Grassmann-manifold based mechanism.

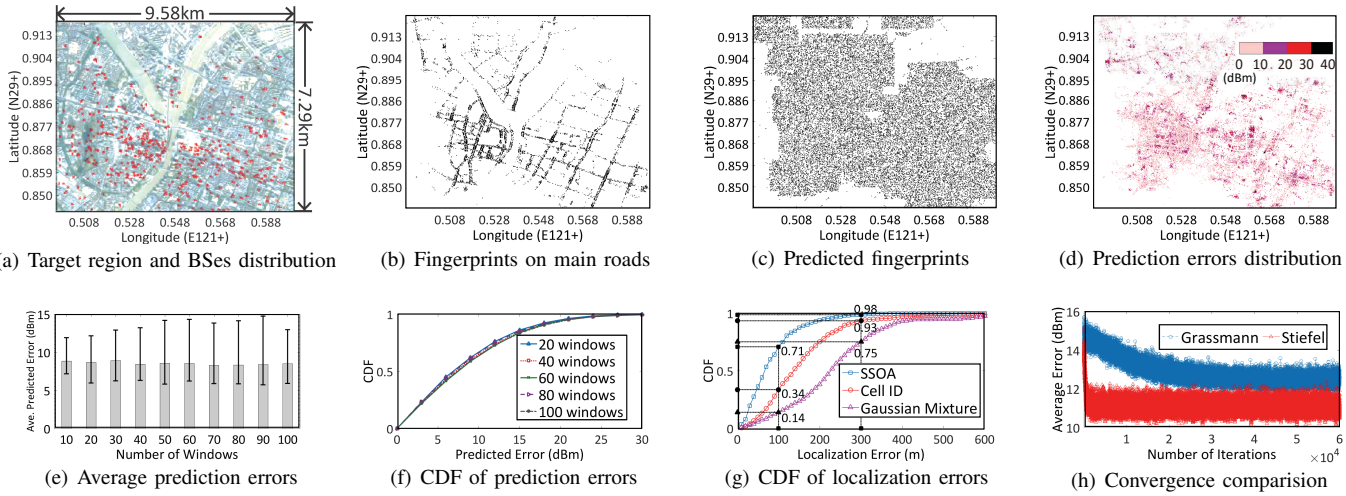
B. Experiments on Large Data Set

Overview of Fingerprints Prediction Results: The map of the city where the data were sampled is shown in Fig. 8(a); the red dots on the map represent the location of the BSes. We use fingerprints collected along the main roads of the city to predict the fingerprints on those branching roads. The spatial distribution of fingerprints on main road are shown in Fig. 8(b), which accounts for only 6.7% of the whole region. After 7 iterations of sliding window based prediction mechanism with SSOA, we obtain Fig. 8(c) shown the prediction result. The predicted region accounts for 73.2% of the whole region, having fingerprints in most of the locations predicted. To examine the prediction accuracy, we compare the predicted results with the ground true, and show corresponding error of each prediction in Fig. 8(d), where different colors represent different levels of errors in dBm. We find that the average and median predicting errors are 8.46 and 7.09 respectively.

Local Performance of Fingerprints Prediction: We here show the local performance of fingerprints prediction in Fig. 8(d). In particular, we randomly sample a $250m \times 300m$ sub-area over the city region as shown in Fig. 8(d) multiple times, and examine the prediction performance within the window each time. Then the average and distribution of the errors can be obtained.

Figure 8(e) shows the average, maximum and minimum prediction errors when we select different numbers of sub-areas to examine. It can be found that the average predicting error (the bars) fluctuates slightly around $8.5dBm$ in different number of windows varying from 10 to 100, with the standard deviation $0.17dBm$. This indicate the stability of average predicting performance in different sub-areas. We can also see that the minimum value is generally much closer to the average error than the maximum value, indicating that the good predictions are more than the bad ones. Figure 8(f) shows the cumulative distribution function (CDF) of prediction errors. It can be seen that the CDFs under different numbers of samplings approximately overlap with each other, indicating that the prediction performance in each sub-area is stable.

Positioning Results with Predicted Fingerprints: We here validate that the predicted fingerprints can be utilized for location estimation with the accuracy and reliability satisfying E911 requirement. We first grid the entire area into 871×663 square cells with each edge length to be $11m$. As mentioned above, the data set contains data from 611 BSes, but a number



(a) Target region and BSes distribution (b) Fingerprints on main roads (c) Predicted fingerprints (d) Prediction errors distribution

(e) Average prediction errors (f) CDF of prediction errors (g) CDF of localization errors (h) Convergence comparision

Fig. 8. Experimental results on $69.8km^2$ dataset.

of base stations are only observed at a couple of locations. Thus we first sort the BSes according to the frequency they are observed at all locations of the area, and select the top 135 BSes. Then the corresponding data account for 96% of the entire data set. To perform localization, we choose those cells that both have the measured and the predicted fingerprints. We construct a fingerprints database with the predicted fingerprints, and use the real data as the user's reported data for localization. According to the statics of the data set, a user's mobile device normally can observe 1 to 12 BSes, and our preliminary experimental results show that the localization accuracy will be unacceptable if the user just report the fingerprint with respect to only one BS; therefore, we just consider the cells that can observe at least two BSes.

With our predicted fingerprints, we compare the performance of fingerprinting localization performance with that of Cell ID (CID) and Gaussian Mixture Model (GMM) based method [16]. The basic idea of the CID approach is to estimate the user's location to be the geometric center of all BSes the user can observe; GMM method is to estimate the location of a reported fingerprint using the GMM model constructed based on the Gaussian radio propagation model, which also can be regarded as a method to predict a given fingerprint's location.

We perform localizations for around 4500 times by three methods respectively, and draw the CDF of localization errors for each method, as shown in Fig. 8(g). The localization error is the Euclidean distance between the user's estimated location and the ground truth. We use the E911's localization requirement benchmark to evaluate the three localization methods, which is "within $100m$ for 67% and within $300m$ for 90%". We can see that the fingerprinting method using our predicted fingerprints by SSOA achieves "within $100m$ for 71% and within $300m$ for 98%", CID method achieves "within $100m$ for 34% and within $300m$ for 93%", and the GMM method achieves "within $100m$ for 14% and within $300m$ for 75%". This is because CID's performance is impacted by the unbalanced distribution of BSes, and GMM's assumption that the received signal strength at a given location is a multivariate Gaussian distributed random variable [16] is not always realistic especially in urban environment with more serious shadowing and multipath effects.

Convergence Rate: Our convergence analysis reveals that the proposed SSOA mechanism converges faster than the Grassmann-manifold optimization algorithm, and we now provide experimental results to validate this claim. We consider the entire area as a giant matrix, and use 40% of the data as the training set to predict the rest of the data. We let the SSOA and the Grassmann-manifold optimization algorithm iterate $60,000$ times and examine the prediction error after each iteration. The prediction error is found by comparing the predicted data and the real data in the other 60% of the data set, and each error is represented as a point in Fig. 8(h). It shows that the average prediction error using SSOA reaches around $11dBm$ within 1000 times, while the error using Grassmann method only reaches around $12.5dBm$ after 30000 times. It takes $75min$ for the Grassmann method to reach $12.5dBm$ error, while our proposed SSOA just consumes around $2min$ to achieve $11dBm$ error.

VII. Conclusion

This paper has proposed to utilize the subspace identification approach to predict fingerprints in unsurveyed areas with available fingerprints sampled in the nearby areas. We have formulated the fingerprints prediction problem into the problem of finding the optimal subspace over the Stiefel manifold, and proposed a streamlined Stiefel-manifold optimization algorithm with fast convergence rate for the fingerprints prediction scenario. Moreover, we have proposed a sliding window mechanism to deal with the practical fingerprints prediction scenario, where the fingerprints are unevenly distributed in the vast area. Combining the two proposed mechanisms enables an efficient method to predict large-scale fingerprints prediction in the km^2 level. Further, we have validated our theoretical analysis and proposed mechanisms by conducting experiments

with real mobile data sets sampled in two cities; it has been shown that the localization accuracy and reliability exceed the requirement of E911 by FCC, moreover, the convergence rate of the proposed mechanism outperforms the Grassmann approach with the similar methodology.

REFERENCES

[1] Enhanced 911 - Wireless Services Online: https://www.fcc.gov/general/enhanced-9-1-1-wireless-services.

[2] Ericsson White Paper, Postioning with LTE, Online:http://www.sharetechnote.com/Docs/WP-LTE-positioning.pdf, Sept. 2011.

[3] European Global Navigation Satellite Systems Agency How to Enable Better Location for Emergency Calls: Galileo and 112 Online:https://www.gsa.europa.eu/news/how-enable-better-location-emergency-calls-galileo-and-112

[4] Rohde & Schwarz, ,LTE Location Based Services Technology Introduction White paper, Online:http://www.rohde-schwarz-wireless.com/documents/LTELBSWhitePaper_RohdeSchwarz.pdf , Apr. 2013

[5] Spirent, White Paper: An Overview of LTE Positioning, Online:https://www.spirent.com/~/media/White%20Papers/Mobile/LTE_LBS_White_Paper_2012.pdf,Feb. 2012

[6] 3GPP Release 8, Online:http://www.3gpp.org/specifications/releases/72-release-8

[7] 3GPP Release 9, Online:http://www.3gpp.org/specifications/releases/71-release-9

[8] Huawei, The second phase of LTE-Advanced LTE-B : 30-fold capacity boosting to LTE, Online:www.huawei.com/ilink/en/download/HW_259010 Apr. 2013

[9] X. Liu, S. Aeron, V. Aggarwal, X. Wang and M. Wu, "Adaptive Sampling of RF Fingerprints for Fine-Grained Indoor Localization," *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2411–2423, Oct. 2016.

[10] D. Milioris, G. Tzagkarakis and A. Papakonstantinou, "Low dimensional signal-strength fingerprint-based positioning in wireless LANs," *Ad Hoc Netw.*, vol. 12, pp. 100–114, 2014.

[11] S. Nikitaki, G. Tsagkatakis and P. Tsakalides, "Efficient multichannel signal strength based localization via matrix completion and Bayesian sparse learning," *IEEE Trans. Mobile Comput.*, vol. 14, no. 11, pp. 2244–2256, Nov. 2015.

[12] D. Milioris, M. Bradonjic and P. Muhlethaler, "Building complete training maps for indoor location estimation," in *Proc. IEEE INFOCOM*, 2015, pp. 75–76

[13] R. Margolies, R. Becker, S. Byers, S. Deb, R. Jana, S. Urbanek and C. Volinsky, "Can you find me now? Evaluation of network-based localization in a 4G LTE network," in *Proc. IEEE INFOCOM*, 2017.

[14] A. Ray, S. Deb and P. Monogioudis, "Location of LTE measurement records with missing information, in *Proc. IEEE INFOCOM*, 2016.

[15] R. Margolies, A. Sridharan, V. Aggarwal, R. Jana, N. Shankaranarayanan, V. A. Vaishampayan, and G. Zussman, "Exploiting mobility in proportional fair cellular scheduling: Measurements and algorithms," *IEEE/ACM Trans. on Netw.*, vol. 24, no. 1, pp. 355–367, 2016.

[16] A. Chakraborty, L. E. Ortiz and S. R. Das, "Network-side positioning of cellular-band devices with minimal effort, in *Proc. IEEE INFOCOM*, 2015.

[17] S. C. Ergen, H. S. Tetikol, M. Kontik, R. Sevlian, R. Rajagopal, and P. Varaiya, "RSSI-fingerprinting-based mobile phone localization with route constraints," *IEEE Trans. Veh. Technol.*, vol. 63, no. 1, pp. 423–428, 2014.

[18] M. Y. Chen, T. Sohn, D. Chmelev, D. Haehnel, J. Hightower, J. Hughes, A. LaMarca, F. Potter, I. Smith and A. Varshavsky, "Practical metropolitan-scale positioning for GSM phones," in *Proc. ACM UbiComp*, 2006.

[19] P. Nurmi, S. Bhattacharya, and J. Kukkonen "A grid-based algorithm for on-device GSM positioning ," in *Proc. ACM UbiComp*, 2010.

[20] G. Sun, J. Chen, W. Guo and K. J. R. Liu, "Signal processing techniques in network-aided positioning: a survey of state-of-the-art positioning designs," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 12–23, 2005.

[21] T. Wigren, " Adaptive Enhanced Cell-ID Fingerprinting Localization by Clustering of Precise Position Measurements, *IEEE Transactions on Vehicular Technology*, vol. 56, no. 5, pp. 3199–3209, Sept. 2007.

[22] L. Shi and T. Wigren, "AECID Fingerprinting Positioning Performance, in *Proc. IEEE GLOBECOM*, 2009.

[23] Pew Research Center, "Older Adults and Technology Use, Research Report, Online:http://www.pewinternet.org/2014/04/03/older-adults-and-technology-use/.

[24] Pew Research Center, "U.S. Smartphone Use in 2015, Research Report, Online:http://www.pewinternet.org/2015/04/01/us-smartphone-use-in-2015/.

[25] Pew Research Center, " Smartphone Ownership and Internet Usage Continues to Climb in Emerging Economies, Research Report, Online:http://www.pewglobal.org/2016/02/22/smartphone-ownership-and-internet-usage-continues-to-climb-in-emerging-economies/.

[26] A. Edelman, T. A. Arias and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *Siam Journal on Matrix Analysis and Applications*, vol. 20, no. 2, pp.303–353, 1998.

[27] J. F. Cai, E. J. Cands and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp.1956–1982, 2010

[28] P. Jain and P. Netrapalli, "Fast Exact Matrix Completion with Finite Samples ," Online:https://arxiv.org/pdf/1411.1087.pdf

[29] D. Lazzaro, "A nonconvex approach to low-rank matrix completion using convex optimization," *Numerical Linear Algebra with Applications*, vol. 23, no. 5, pp.801–824, 2016.

[30] K. Mohan and M. Fazel, "Iterative reweighted algorithms for matrix rank minimization," *Journal of Machine Learning Research*, vol. 13, no. 1, pp.3441–3473, 2012.

[31] L. Balzano, R. Nowak and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Communication, Control, and Computing*, 2010, pp. 704–711.

[32] L. Balzano and S. J. Wright, "Local convergence of an algorithm for subspace identification from partial data," *Foundations of Computational Mathematics*, vol. 15, no. 5, pp.1279–1314, 2015.

[33] D. Zhang and L. Balzano, "Global convergence of a Grassmannian gradient descent algorithm for subspace estimation," Online:https://arxiv.org/pdf/1506.07405.pdf.

[34] E. J. Candes and J. Romberg, "Sparsity and Incoherence in Compressive Sampling," *Inverse Problems*, vol. 23, no. 3, pp.969–985, 2007.

[35] M. E. Kilmer, K. Braman and N. Hao and R. C. Hoover, "Third-Order Tensors as Operators on Matrices: A Theoretical and Computational Framework with Applications in Imaging," *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 1, pp.148–172, 2013.

[36] S. Fisk, "A Note on Weyl's Inequality," *American Mathematical Monthly*, vol. 104, no. 5, pp.257–258, 1997