Wirtinger Flow for Nonconvex Blind Demixing with Optimal Step Size

Chih-Ho Hsu, Carlos Feres, *Member, IEEE*, Zhi Ding, *Fellow, IEEE*

***Abstract*—The prospect of massive deployment of devices for Internet-of-Things (IoT) motivates grant-free access for simultaneously uplink transmission by multiple nodes. Blind demixing represents a promising technique for recovering multiple such source signals over unknown channels. Recent studies show Wirtinger Flow (WF) algorithm can be effective in blind demixing. However, existing theoretical results on WF step size selection tend to be conservative and slow down convergence rates. To overcome this limitation, we propose an improved WF (WF-OPT) by optimizing its step size in each iteration and expediting the convergence. We provide a theoretical guarantee on the strict contraction of WF-OPT and present the upper bounds of the contraction ratio. Simulation results demonstrate the expected convergence gains.**

***Index Terms*—Blind demixing, blind deconvolution, grant-free access, optimal step size, Wirtinger Flow.**

I. Introduction

BLIND demixing is a well-known fundamental problem that arises in various fields such as wireless communications, image processing, and array processing [1]. The rapidly growing deployment of IoT devices poses challenges to access control and spectrum management. Blind demixing can play a major role to facilitate grant-free access for simultaneous data transmission by multiple uplink nodes over unknown channels [2]. In blind deconvolution of a single source, the task is to recover an unknown signal $\boldsymbol{x}$ from its (noisy) convolution

$$y = \boldsymbol{h} * \boldsymbol{x} + n \tag{1}$$

with an unknown channel $\boldsymbol{h}$ under additive noise n. In the IoT context, $\boldsymbol{x}$ corresponds to the signal transmitted from a sensor/source node whereas $\boldsymbol{h}$ captures its unknown wireless fading channel. Blind deconvolution is feasible by exploiting high-order statistics of $\boldsymbol{x}$ under certain practical conditions [3].

Blind demixing extends the task of *single*-source recovery to simultaneous recovery of *multiple* source signals, where the observed signal y is a noisy superposition of S convolutions between multiple source signals $\{\boldsymbol{x}_i\}_{i=1}^S$ and their corresponding unknown channels $\{\boldsymbol{h}_i\}_{i=1}^S$. Given S active sources, the uplink host node receives

$$y = \sum_{i=1}^{S} \boldsymbol{h}_i * \boldsymbol{x}_i + n. \tag{2}$$

Note that this extension is nontrivial since the incoherence among signals from different sources poses special challenges [4]. Moreover, the bilinear nature of measurements of (2) is incompatible with certain existing demixing methods [5] developed for linear measurements.

A growing number of research works have contributed to address these challenges. One approach is to transform the original bilinear problem into a convex optimization problem by lifting products of unknown signal vectors into rank-one matrices. To overcome the non-convexity of the rank-one condition, such solutions apply semi-definite relaxation to derive a convex optimization problem that can be tackled via semi-definite programming [6] [7] and nuclear norm minimization [8]. Although convex optimization via lifting appears attractive, such solutions significantly expand the solution space that leads to exceedingly high computational cost, and the projection from the high dimensional lifted space back to the required original solution space also introduces uncertainties.

To mitigate the complexity of convex lifting, an alternative is to remain in the original lower dimensional space. Recent studies have reexamined the non-convexity of blind demixing by exploiting manifold geometry of fixed-rank matrices [9] via Riemannian optimization. However, iterative Riemannian optimization still poses challenges to statistical analysis. Alternatively, Ling [10] proposed a regularized Gradient Descent (GD) method for non-convex blind demixing, though its attractive optimality properties require careful tuning.

From another perspective, a connection between phase retrieval and blind source separation was noted at least as early as in 1996 [11]. The Wirtinger Flow (WF) algorithm, originally proposed in [12] for phase retrieval, is a simple but effective method for high dimensional statistical problems. WF is a two-stage regularization-free algorithm that consists of spectral initialization followed by a standard GD procedure. Previous studies in [9], [13] show that WF can also tackle the challenging blind deconvolution and demixing problems with linear convergence rates in both noiseless and noisy scenarios.

However, theoretical results thus far [9], [13] only guarantee convergence to ground-truth signals when the step size is a constant selected within an approximate range. In this sense, the WF step size may be set conservatively to ensure convergence, resulting in a possibly slow convergence rate. To attain faster convergence rates, the algorithm proposed in [14] adapts the GD step size by exploiting the geometry of blind demixing. Nonetheless, this algorithm only guarantees asymptotic convergence to ground-truths.

Motivated by these recent works and the challenges of blind demixing, in this letter we propose an efficient and provable WF-based blind demixing procedure: WF-OPT. The proposed WF-OPT analytically obtains an optimized step size for implementing WF at each iteration, which substantially improves the convergence speed with only a marginal increase in computational cost. We further provide convergence analysis on the proposed WF-OPT and derive theoretical performance

This material is based upon work supported by the National Science Foundation under Grants No. 2009001 and No. 2029027.

Chih-Ho Hsu, Carlos Feres, and Zhi Ding are with the Department of Electrical and Computer Engineering, University of California, Davis, CA, 95616, USA (email: lchhsu@ucdavis.edu, cjferes@ucdavis.edu and zding@ucdavis.edu) Corresponding author: Zhi Ding.

bounds. Finally, our simulation results demonstrate the significant performance gains by WF-OPT.

Notations: Throughout this letter, we denote $(\cdot)^*$, $(\cdot)^{\mathsf{T}}$ and $(\cdot)^{\mathsf{H}}$ as conjugate, transpose, and conjugate transpose, respectively. We use small bold letters for vectors, capital bold letters for matrices, and non-bold font for scalars and functions. Finally, $\|\cdot\|$ denotes the ℓ_2 norm.

II. System Model and Problem Formulation

In general, the blind demixing problem becomes intractable without imposing certain structures, and hence we first assume that convoluted signals belong to known subspaces [8], [10]. We consider S sources simultaneously transmitting data to an access point (AP), who observes the mixture of the transmitted signals through their corresponding wireless channels with N samples. In the frequency domain (e.g. using OFDM), we can represent the j-th received signal sample as

$$y_j = \sum_{i=1}^{S} \boldsymbol{b}_j^{\mathsf{H}} \bar{\boldsymbol{h}}_i (\bar{\boldsymbol{x}}_i)^{\mathsf{H}} \boldsymbol{a}_{ij} + n_j, \quad 1 \le j \le N, \tag{3}$$

which are noisy bilinear measurements of the mixture of the S ground-truth transmitted signals $\bar{\boldsymbol{x}}_i \in \mathbb{C}^K$ and their corresponding channels $\bar{\boldsymbol{h}}_i \in \mathbb{C}^L$. Here, K is the length of the signal vectors and L is the maximum length of the channels whereas $\boldsymbol{a}_{ij} \in \mathbb{C}^K$ and $\boldsymbol{b}_j \in \mathbb{C}^L$ are known design vectors.

We assume that $\boldsymbol{a}_{ij}$ are i.i.d. multivariate complex Gaussian, i.e. $\boldsymbol{a}_{ij} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I}_K)$. Without loss of generality, we focus on OFDM transmissions such that $\boldsymbol{b}_j$ depends on the first L columns of discrete Fourier Transform (DFT) matrix $\boldsymbol{F} \in \mathbb{C}^{N \times N}$, via $\boldsymbol{F} \begin{bmatrix} \boldsymbol{I}_L \\ \boldsymbol{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{b}_1 & \cdots & \boldsymbol{b}_N \end{bmatrix}^{\mathsf{H}} \in \mathbb{C}^{N \times L}$ [8], [9]. Furthermore, n_j is i.i.d. circularly symmetric complex AWGN, i.e. $n_j \sim \mathcal{CN}(0, d_0^2/\gamma)$, where $d_0^2 = \sum_{i=1}^S \|\bar{\boldsymbol{h}}_i\|^2 \|\bar{\boldsymbol{x}}_i\|^2 / N^2$ and γ denotes signal-to-noise ratio (SNR).

According to the model of Eq.(3), our goal in blind demixing is to simultaneously recover signals $\{\bar{\boldsymbol{x}}_i\}$ (and possibly channel responses $\{\bar{\boldsymbol{h}}_i\}$) by solving the following problem

$$\min_{\boldsymbol{h}, \boldsymbol{x}} \; G(\boldsymbol{h}, \boldsymbol{x}), \quad G(\boldsymbol{h}, \boldsymbol{x}) = \sum_{j=1}^{N} \left| y_j - \sum_{i=1}^{S} \boldsymbol{b}_j^{\mathsf{H}} \boldsymbol{h}_i \boldsymbol{x}_i^{\mathsf{H}} \boldsymbol{a}_{ij} \right|^2 \tag{4}$$

where we denote $\boldsymbol{h} = [\boldsymbol{h}_1^{\mathsf{T}}, \cdots, \boldsymbol{h}_S^{\mathsf{T}}]^{\mathsf{T}}$, $\boldsymbol{x} = [\boldsymbol{x}_1^{\mathsf{T}}, \cdots, \boldsymbol{x}_S^{\mathsf{T}}]^{\mathsf{T}}$.

III. Wirtinger Flow with Optimal Step Size

The original WF is a two-stage algorithm that consists of a spectral initialization followed by iterative GD. In this letter, we further propose to determine the optimal step size at each GD iteration to accelerate convergence speed and improve performance. We label this approach WF-OPT, summarized in Algorithm 1. We present our derivation in the following.

1) Spectral Initialization: We first define auxiliary matrices

$$\boldsymbol{M}_i = \sum_{j=1}^{N} y_j \boldsymbol{b}_j \boldsymbol{a}_{ij}^{\mathsf{H}}, \qquad 1 \le i \le S, \tag{5}$$

and let $\sigma_1(\boldsymbol{M}_i)$, $\breve{\boldsymbol{h}}_i$ and $\breve{\boldsymbol{x}}_i$ be the leading singular value, left singular vector and right singular vector of $\boldsymbol{M}_i$, respectively. Next, we set the initial point of the WF algorithm to

$$\boldsymbol{h}_i^0 = \sqrt{\sigma_1(\boldsymbol{M}_i)} \breve{\boldsymbol{h}}_i, \quad \boldsymbol{x}_i^0 = \sqrt{\sigma_1(\boldsymbol{M}_i)} \breve{\boldsymbol{x}}_i. \quad 1 \le i \le S. \tag{6}$$

2) Gradient Descent Procedure: After initialization, the algorithm will iteratively update the variables $\boldsymbol{h}^t$, $\boldsymbol{x}^t$ for $t \ge 1$ using Wirtinger derivatives [15], where t denotes the iteration index. Let $\nabla_{\boldsymbol{h}_i} G$ and $\nabla_{\boldsymbol{x}_i} G$ denote the Wirtinger gradient of the cost function $G(\boldsymbol{h}, \boldsymbol{x})$ with respect to $\boldsymbol{h}_i$ and $\boldsymbol{x}_i$ respectively, which are computed as

$$\nabla_{\boldsymbol{h}_i} G(\boldsymbol{h}, \boldsymbol{x}) = \sum_{j=1}^{N} \left(\sum_{i=1}^{S} \boldsymbol{b}_j^{\mathsf{H}} \boldsymbol{h}_i \boldsymbol{x}_i^{\mathsf{H}} \boldsymbol{a}_{ij} - y_j \right) \boldsymbol{b}_j^{\mathsf{H}} \boldsymbol{a}_{ij}^{\mathsf{H}} \boldsymbol{x}_i, \tag{7a}$$

$$\nabla_{\boldsymbol{x}_i} G(\boldsymbol{h}, \boldsymbol{x}) = \sum_{j=1}^{N} \left(\sum_{i=1}^{S} \boldsymbol{b}_j^{\mathsf{H}} \boldsymbol{h}_i \boldsymbol{x}_i^{\mathsf{H}} \boldsymbol{a}_{ij} - y_j \right) \boldsymbol{a}_{ij}^{\mathsf{H}} \boldsymbol{b}_j^{\mathsf{H}} \boldsymbol{h}_i, \tag{7b}$$

for $1 \le i \le S$. Then the GD of the original WF algorithm uses a constant step size η as follows:

$$\boldsymbol{h}_i^{t+1} = \boldsymbol{h}_i^t - \frac{\eta}{\|\boldsymbol{x}_i^t\|^2} \nabla_{\boldsymbol{h}_i^t} G(\boldsymbol{h}_i^t, \boldsymbol{x}_i^t), \; 1 \le i \le S, \tag{8a}$$

$$\boldsymbol{x}_i^{t+1} = \boldsymbol{x}_i^t - \frac{\eta}{\|\boldsymbol{h}_i^t\|^2} \nabla_{\boldsymbol{x}_i^t} G(\boldsymbol{h}_i^t, \boldsymbol{x}_i^t), \; 1 \le i \le S. \tag{8b}$$

In Eq. (8), gradients with respect to $\boldsymbol{h}_i$ are normalized by the norm of signal vectors, and vice versa, which ensures both channels and signals are bounded in a practical way regardless of the scalar invariance of (4) [13].

3) Optimizing Step Size: Instead of using a fixed step size for both channels and signals, WF-OPT shall separately compute optimized step sizes for $\boldsymbol{h}^t$ and $\boldsymbol{x}^t$ at each iteration $t \ge 0$, denoted as η_h^{opt} and η_x^{opt}, respectively. To that end, we derive from the cost function of (4) and consider the following optimization problems:

$$\eta_h^{\text{opt}} = \arg\min_{\eta_h} G\big(\boldsymbol{h}^t - \eta_h \nabla_{\boldsymbol{h}^t} G(\boldsymbol{h}^t, \boldsymbol{x}^t), \boldsymbol{x}^t\big), \tag{9a}$$

$$\eta_x^{\text{opt}} = \arg\min_{\eta_x} G\big(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t - \eta_x \nabla_{\boldsymbol{x}^t} G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t)\big). \tag{9b}$$

where, for convenience, we stack normalized gradients with respect to the i-th channel and signal

$$\nabla_{\boldsymbol{h}^t} G = \left[\|\boldsymbol{x}_1^t\|^{-2} (\nabla_{\boldsymbol{h}_1^t} G)^{\mathsf{T}} \quad \cdots \quad \|\boldsymbol{x}_S^t\|^{-2} (\nabla_{\boldsymbol{h}_S^t} G)^{\mathsf{T}} \right]^{\mathsf{T}},$$

$$\nabla_{\boldsymbol{x}^t} G = \left[\|\boldsymbol{h}_1^{t+1}\|^{-2} (\nabla_{\boldsymbol{x}_1^t} G)^{\mathsf{T}} \quad \cdots \quad \|\boldsymbol{h}_S^{t+1}\|^{-2} (\nabla_{\boldsymbol{x}_S^t} G)^{\mathsf{T}} \right]^{\mathsf{T}}.$$

Note from (9) that we obtain the optimal step size for $\boldsymbol{x}^t$ (i.e. η_x^{opt}) based on the updated channel $\boldsymbol{h}^{t+1}$, i.e. computing sequential gradients separately with transient iterates, instead of optimizing the step size simultaneously for both channel and signal iterates and their corresponding gradients. We justify this choice after presenting our analytical derivation.

We first focus on solving (9a). We define vectors $\boldsymbol{\Phi}_h, \boldsymbol{\Psi}_h \in \mathbb{C}^N$ with respect to their j-th element, $1 \le j \le N$, given by

$$[\boldsymbol{\Phi}_h^t]_j = \sum_{i=1}^{S} \boldsymbol{b}_j^{\mathsf{H}} \frac{\nabla_{\boldsymbol{h}_i^t} G}{\|\boldsymbol{x}_i^t\|^2} (\boldsymbol{x}_i^t)^{\mathsf{H}} \boldsymbol{a}_{ij}, \tag{10a}$$

$$[\boldsymbol{\Psi}_h^t]_j = \sum_{i=1}^{S} \boldsymbol{b}_j^{\mathsf{H}} \boldsymbol{h}_i^t (\boldsymbol{x}_i^t)^{\mathsf{H}} \boldsymbol{a}_{ij} - y_j. \tag{10b}$$

Here, $\boldsymbol{\Phi}_h^t$ can be interpreted as the perturbation vector in the cost value caused by descending in direction $\nabla_{\boldsymbol{h}^t} G$ in each sample, while $\boldsymbol{\Psi}_h^t$ is the vector that collects the cost value contributed by each sample, and satisfies $\|\boldsymbol{\Psi}_h^t\|^2 = G(\boldsymbol{h}^t, \boldsymbol{x}^t)$.

Therefore, we can expand Eq.(4) as

$$\begin{aligned} G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t) &= \min_{\eta_h} G(\boldsymbol{h}^t - \eta_h \nabla_{\boldsymbol{h}^t} G, \boldsymbol{x}^t) \\ &= \min_{\eta_h} \sum_{j=1}^{N} \Big| y_j - \sum_{i=1}^{S} \boldsymbol{b}_j^{\mathsf{H}} \Big(\boldsymbol{h}_i^t - \eta_h \frac{\nabla_{\boldsymbol{h}_i^t} G}{\|\boldsymbol{x}_i^t\|^2} \Big) (\boldsymbol{x}_i^t)^{\mathsf{H}} \boldsymbol{a}_{ij} \Big|^2 \\ &= \min_{\eta_h} \sum_{j=1}^{N} \Big| [\boldsymbol{\Phi}_h^t]_j \eta_h - [\boldsymbol{\Psi}_h^t]_j \Big|^2 \\ &= \min_{\eta_h} \|\boldsymbol{\Phi}_h^t\|^2 \Big(\eta_h - \frac{\mathrm{Re}\big\{ {\boldsymbol{\Phi}_h^t}^{\mathsf{H}} \boldsymbol{\Psi}_h^t \big\}}{\|\boldsymbol{\Phi}_h^t\|^2} \Big)^2 + G(\boldsymbol{h}^t, \boldsymbol{x}^t) \\ &\qquad - \frac{\mathrm{Re}^2\big\{ {\boldsymbol{\Phi}_h^t}^{\mathsf{H}} \boldsymbol{\Psi}_h^t \big\}}{\|\boldsymbol{\Phi}_h^t\|^2}, \end{aligned} \tag{11}$$

which is a quadratic function of η_h. From the last equality of (11), selecting η_h to optimal value η_h^{opt} maximizes $G(\boldsymbol{h}^t, \boldsymbol{x}^t) - \|\boldsymbol{\Phi}_h^t\|^{-2} \mathrm{Re}^2\{ {\boldsymbol{\Phi}_h^t}^{\mathsf{H}} \boldsymbol{\Psi}_h^t \}$. Moreover, the optimal η_h^{opt} for $\boldsymbol{h}^t$ is

$$\eta_h^{\mathrm{opt}} = \frac{\mathrm{Re}\big\{ {\boldsymbol{\Phi}_h^t}^{\mathsf{H}} \boldsymbol{\Psi}_h^t \big\}}{\|\boldsymbol{\Phi}_h^t\|^2} \in \mathbb{R}. \tag{12}$$

Note that if $\|\boldsymbol{\Phi}_h\| = 0$, then $\mathrm{Re}\big\{ {\boldsymbol{\Phi}_h^t}^{\mathsf{H}} \boldsymbol{\Psi}_h^t \big\} = 0$ and Eq.(11) reduces to a constant term $G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t) = \|\boldsymbol{\Psi}_h^t\|^2 = G(\boldsymbol{h}^t, \boldsymbol{x}^t)$. Moreover, if $\mathrm{Re}\big\{ {\boldsymbol{\Phi}_h^t}^{\mathsf{H}} \boldsymbol{\Psi}_h^t \big\} = 0$, Eq.(11) reduces to $G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t) = \min_{\eta_h} \|\boldsymbol{\Phi}_h^t\|^2 \eta_h^2 + \|\boldsymbol{\Psi}_h^t\|^2$ with $\eta_h^{\mathrm{opt}} = 0$. In both cases, the optimized GD step leads to $G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t) = G(\boldsymbol{h}^t, \boldsymbol{x}^t)$, which corresponds to a stable equilibrium of the algorithm, indicating that the GD has already converged. Thus, we can omit these two cases in our analysis. Finally, we can solve (9b) by defining $\boldsymbol{\Phi}_x^t, \boldsymbol{\Psi}_x^t \in \mathbb{C}^N$ with respect to their j-th element, $1 \le j \le N$ in a similar way:

$$[\boldsymbol{\Phi}_x^t]_j = \sum_{i=1}^{S} \boldsymbol{b}_j^{\mathsf{H}} \boldsymbol{h}_i^{t+1} \Big(\frac{\nabla_{\boldsymbol{x}_i^t} G}{\|\boldsymbol{h}_i^{t+1}\|^2} \Big)^{\mathsf{H}} \boldsymbol{a}_{ij}, \tag{13a}$$

$$[\boldsymbol{\Psi}_x^t]_j = \sum_{i=1}^{S} \boldsymbol{b}_j^{\mathsf{H}} \boldsymbol{h}_i^{t+1} (\boldsymbol{x}_i^t)^{\mathsf{H}} \boldsymbol{a}_{ij} - y_j. \tag{13b}$$

We then can obtain

$$\eta_x^{\mathrm{opt}} = \frac{\mathrm{Re}\big\{ {\boldsymbol{\Phi}_x^t}^{\mathsf{H}} \boldsymbol{\Psi}_x^t \big\}}{\|\boldsymbol{\Phi}_x^t\|^2} \in \mathbb{R}. \tag{14}$$

The WF-OPT algorithm will repeatedly implement GD on $\boldsymbol{h}^t$ and $\boldsymbol{x}^t$ by computing the optimal step sizes until convergence, which we define by setting a small tolerance ϵ for the computed step sizes, i.e. $\eta_h^{\mathrm{opt}} < \epsilon$ or $\eta_x^{\mathrm{opt}} < \epsilon$.

Now we can justify our choice for optimizing the step size using sequential gradients with transient variables instead of the full gradient containing both $\nabla_{\boldsymbol{h}^t} G$ and $\nabla_{\boldsymbol{x}^t} G$. First, the optimal step size for the full gradient is a real root of a third-order polynomial without a closed form. Second, and more importantly, our tests show that the performance of the algorithm using the full-gradient optimal step size is only marginally better than using sequential gradients, despite higher computational complexity. Choosing sequential gradients with transient variables offers an attractive tradeoff, with faster computation and similar convergence speed.

Algorithm 1: WF with Optimal Step Size (WF-OPT)

```
Input: known design vectors $\{\boldsymbol{a}_{ij}\}_{i=1,j=1}^{S,N}$, $\{\boldsymbol{b}_j\}_{j=1}^{N}$,
       and bilinear measurements $\{y_j\}_{j=1}^{N}$
Output: recovered signals $\boldsymbol{h}_i^*, \boldsymbol{x}_i^*$, for $1 \le i \le S$
Compute $\boldsymbol{M}_i = \sum_{j=1}^{N} y_j \boldsymbol{b}_j \boldsymbol{a}_{ij}^{\mathsf{H}}$ for $1 \le i \le S$ and
  obtain its largest singular value and singular vectors
Set $\boldsymbol{h}_i^0 = \sqrt{\sigma_1(\boldsymbol{M}_i)} \breve{\boldsymbol{h}}_i, 1 \le i \le S$
Set $\boldsymbol{x}_i^0 = \sqrt{\sigma_1(\boldsymbol{M}_i)} \breve{\boldsymbol{x}}_i, 1 \le i \le S$
while not converged do
    Compute optimal step size $\eta_h^{\mathrm{opt}}$ for $\boldsymbol{h}^t$ using (12)
    Update $\boldsymbol{h}^{t+1} = \boldsymbol{h}^t - \eta_h^{\mathrm{opt}} \nabla_{\boldsymbol{h}} G(\boldsymbol{h}^t, \boldsymbol{x}^t)$
    Compute optimal step size $\eta_x^{\mathrm{opt}}$ for $\boldsymbol{x}^t$ using (14)
    Update $\boldsymbol{x}^{t+1} = \boldsymbol{x}^t - \eta_x^{\mathrm{opt}} \nabla_{\boldsymbol{x}} G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t)$
end
Compute scaling factors $\alpha_i = x_{i,1}^{\mu} / \widehat{x}_i$, for $1 \le i \le S$
Return $\boldsymbol{h}_i^{\mathrm{opt}} = (\alpha_i^*)^{-1} \boldsymbol{h}_i^{\mu}$ and $\boldsymbol{x}_i^{\mathrm{opt}} = \alpha_i \boldsymbol{x}_i^{\mu}$, $1 \le i \le S$
```

4) Signal Recovery: It is clear that the channel $\boldsymbol{h}$ and signal $\boldsymbol{x}$ in blind demixing are only identifiable up to global scaling [13]. That is, for any nonzero constant $\alpha \in \mathbb{C}$, we have

$$\boldsymbol{h}_i \boldsymbol{x}_i^{\mathsf{H}} = \big[(\alpha^*)^{-1} \boldsymbol{h}_i \big] \big(\alpha \boldsymbol{x}_i \big)^{\mathsf{H}}, \qquad 1 \le i \le S, \tag{15}$$

which means that the result of WF is inherently invariant to a scaling and phase rotation of the ground-truth values. Therefore, after GD convergence at iteration μ, WF-OPT applies proper scaling to the obtained final iterate $(\boldsymbol{h}^{\mu}, \boldsymbol{x}^{\mu})$. This is easily done by, e.g., transmitting one known pilot symbol $\widehat{x}_i$ to estimate the scalar/phase factor of each source at the AP. Another practical approach is to leverage a CRC. If data is modulated using QAM, then by checking the recovered data sequence for each user against an embedded CRC/FEC code for each of the four phase ambiguities, we can determine the correct phase when the packet passes the CRC or FEC parity check. Without loss of generality, assuming that the first symbol of each signal is the pilot, the scaling factors are $\alpha_i = x_{i,1}^{\mu} / \widehat{x}_i, 1 \le i \le S$, and the outputs of the WF-OPT are

$$\boldsymbol{h}_i^{\mathrm{opt}} = (\alpha_i^*)^{-1} \boldsymbol{h}_i^{\mu}, \qquad \boldsymbol{x}_i^{\mathrm{opt}} = \alpha_i \boldsymbol{x}_i^{\mu}, \qquad 1 \le i \le S. \tag{16}$$

Let us consider the computation complexity of WF-OPT in terms of N and S. From Eq. (5), the spectral initialization step exhibits a complexity of $\mathcal{O}(NS)$. Jointly, Eqs. (7) and (8) have a cost of $\mathcal{O}(NS)$ to compute gradients and update iterates. Further, from the definition of $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ and the closed form of η, step size optimization has a complexity of $\mathcal{O}(NS)$. If convergence takes T iterations, the overall complexity of the gradient descent process will be $\mathcal{O}(TNS)$, and recovering the signals has complexity of $\mathcal{O}(S)$, according to (15) and (16). Therefore, the overall complexity of WF-OPT is of order $\mathcal{O}(TNS)$, which is equivalent to WF.

IV. Performance Analysis

The convergence properties and sample complexity of non-regularized WF have been studied in [9], [13], [16] and WF-OPT inherits these properties, namely convergence guarantees with a given sample complexity and signal/channel incoherence, and computational complexity. Hence, we omit theoretical convergence analysis of WF-OPT by assuming that the number of samples is large enough to guarantee convergence [9, Theorem 1]. In this section, then, we focus on demonstrating that WF-OPT iterates provide a strict contraction of the cost function. We achieve this goal by stating the following lemmas.

Lemma 1. *At each WF-OPT iteration $t \geq 0$ before convergence (i.e. $\eta^{\text{opt}} \neq 0$), the cost function G will strictly decrease. Alternatively, the contraction ratios at iteration t satisfy*

$$0 \leq \zeta_h^t = \frac{G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t)}{G(\boldsymbol{h}^t, \boldsymbol{x}^t)} < 1, \tag{17a}$$

$$0 \leq \zeta_x^t = \frac{G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^{t+1})}{G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t)} < 1. \tag{17b}$$

Proof. Recall that $G(\cdot,\cdot)$ is always positive, and thus $\zeta_h^t, \zeta_x^t \geq 0$. We then focus on (17a). From (11), the difference between numerator and denominator of ζ_h^t is $G(\boldsymbol{h}^t, \boldsymbol{x}^t) - G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t) = \|\boldsymbol{\Psi}_h^t\|^2 - \min_{\eta_h} \|\boldsymbol{\Phi}_h^t \eta_h - \boldsymbol{\Psi}_h^t\|^2 = -\max_{\eta_h} \|\boldsymbol{\Phi}_h^t\|^2 \big(\eta_h - \eta_h^{\text{opt}}\big)^2 + \|\boldsymbol{\Phi}_h^t\|^2 (\eta_h^{\text{opt}})^2$, which is a quadratic function of η_h. If $\eta_h^{\text{opt}} \neq 0$, the maximum of the difference will be positive and hence, $0 \leq \zeta_h^t < 1$. (17b) follows similarly. □

Lemma 2. *At each iteration $t \geq 0$ the cost value of WF-OPT is bounded by the cost value of the previous iteration. i.e.*

$$G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t) \leq \big(1 - r^2(\boldsymbol{\Phi}_h^t, \boldsymbol{\Psi}_h^t)\big)\, G(\boldsymbol{h}^t, \boldsymbol{x}^t), \tag{18a}$$

$$G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^{t+1}) \leq \big(1 - r^2(\boldsymbol{\Phi}_x^t, \boldsymbol{\Psi}_x^t)\big)\, G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t), \tag{18b}$$

where $r(\boldsymbol{u}, \boldsymbol{v})$ represents the correlation coefficient of two vectors $\boldsymbol{u}, \boldsymbol{v}$, defined by $r(\boldsymbol{u}, \boldsymbol{v}) = \frac{\sum_{j=1}^N |u_j||v_j|}{\|\boldsymbol{u}\|\|\boldsymbol{v}\|}$ and vectors $\boldsymbol{\Phi}_h^t$, $\boldsymbol{\Psi}_h^t$, $\boldsymbol{\Phi}_x^t$ and $\boldsymbol{\Psi}_x^t$ defined in Eq.(10) and (13).

Proof. Starting from (18a), we arrange the definition of ζ_h^t and expand the equation as follows:

$$\begin{aligned} G(\boldsymbol{h}^{t+1}, \boldsymbol{x}^t) &= \zeta_h^t G(\boldsymbol{h}^t, \boldsymbol{x}^t) = \frac{\|\boldsymbol{\Phi}_h^t \eta_h^{\text{opt}} - \boldsymbol{\Psi}_h^t\|^2}{\|\boldsymbol{\Psi}_h^t\|^2} G(\boldsymbol{h}^t, \boldsymbol{x}^t) \\ &\overset{\text{(i)}}{=} \left(1 - \frac{\text{Re}^2\{\boldsymbol{\Phi}_h^{t\,\mathsf{H}} \boldsymbol{\Psi}_h^t\}}{\|\boldsymbol{\Phi}_h^t\|^2 \|\boldsymbol{\Psi}_h^t\|^2}\right) G(\boldsymbol{h}^t, \boldsymbol{x}^t) \\ &\overset{\text{(ii)}}{\leq} \left(1 - \frac{\left(\sum_{j=1}^N \big|[\boldsymbol{\Phi}_h^t]_j\big| \cdot \big|[\boldsymbol{\Psi}_h^t]_j\big|\right)^2}{\|\boldsymbol{\Phi}_h^t\|^2 \|\boldsymbol{\Psi}_h^t\|^2}\right) G(\boldsymbol{h}^t, \boldsymbol{x}^t) \\ &\overset{\text{(iii)}}{=} \big(1 - r^2(\boldsymbol{\Phi}_h^t, \boldsymbol{\Psi}_h^t)\big) G(\boldsymbol{h}^t, \boldsymbol{x}^t) \end{aligned} \tag{19}$$

where equality (i) follows from using the result of Eq. (12), inequality (ii) holds by applying the AM-GM inequality, and inequality (iii) comes from the definition of correlation coefficient $r(\boldsymbol{u}, \boldsymbol{v})$. We can prove (18b) in a similar manner. □

Remark: Alternatively, $r(\boldsymbol{u}, \boldsymbol{v})$ can be interpreted as a measure of cosine similarity between vectors $\boldsymbol{u}$ and $\boldsymbol{v}$. Therefore, the more correlated $\boldsymbol{\Phi}_h^t$ and $\boldsymbol{\Psi}_h^t$ are, i.e., the gradient is more consistent with the magnitude of cost reduction, the tighter the contraction bound will be.

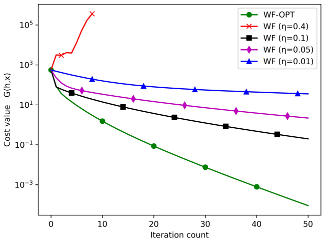

Fig. 1: Convergence for different step sizes.

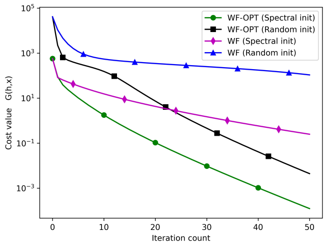

Fig. 2: Convergence for different initialization methods.

Finally, by invoking Lemmas 1 and 2, the cost value obtained after the GD step of WF-OPT is

$$\begin{aligned} G(\boldsymbol{h}^\mu, \boldsymbol{x}^\mu) &= C_1 G(\boldsymbol{h}^0, \boldsymbol{x}^0) \\ &\leq \Big(\prod_{t=1}^{\mu} \big(1 - r^2(\boldsymbol{\Phi}_h^t, \boldsymbol{\Psi}_h^t)\big)\big(1 - r^2(\boldsymbol{\Phi}_x^t, \boldsymbol{\Psi}_x^t)\big)\Big) G(\boldsymbol{h}^0, \boldsymbol{x}^0) \end{aligned}$$

where $C_1 = \prod_{t=1}^{\mu} \zeta_h^t \zeta_x^t < 1$ is a constant, and μ is the iteration index when the GD stage is completed.

V. Numerical Experiments

In this section, we provide simulation tests to validate the performance gain of the proposed WF-OPT algorithms.

In our simulations, we use the original WF as a baseline for comparison. Unless otherwise stated, we use the following settings throughout our tests. We assume there are $S = 10$ sources simultaneously transmitting signals $\{\bar{\boldsymbol{x}}_i\}_{i=1}^S \in \mathbb{C}^K$ of 64 QPSK symbols plus one known pilot symbol to an AP (i.e. $K = 65$) and further normalize each source signal vector. The sample size is set to $N = 50K$ which varies with K while the step size of WF is chosen by trial and error to an appropriate value $\eta = 0.1$. The known design vectors $\{\boldsymbol{a}_{ij}\}_{i=1,j=1}^{S,N}$ and $\{\boldsymbol{b}_j\}_{j=1}^N$ are defined according to the descriptions in Section II. The channel ground-truths $\{\bar{\boldsymbol{h}}_i\}_{i=1}^S \in \mathbb{C}^L$ are i.i.d. Rayleigh fading, i.e. $\bar{\boldsymbol{h}}_i \sim \mathcal{N}(\boldsymbol{0}, \frac{1}{2}\boldsymbol{I}_L) + i\mathcal{N}(\boldsymbol{0}, \frac{1}{2}\boldsymbol{I}_L)$, and we set $L = K$. We also consider noiseless scenarios first, to then study the effect of channel noise.

A. Effect of Algorithm Settings

Fig. 1 shows that WF-OPT significantly outperforms WF for several choices of step size. WF-OPT attains the highest cost reduction along directions of the alternating gradients with the help of optimized step sizes. In contrast, the performance of the original WF varies with the step size selection. For example, if η is too small, convergence is rather slow; if η is too large, the algorithm may even diverge, as in the case of $\eta = 0.4$.

Fig. 2 compares spectral initialization with random initialization, where the initial iterate $(\boldsymbol{h}^0, \boldsymbol{x}^0)$ are generated randomly. Spectral initialization significantly lowers the cost at the first iteration compared with the random initialization, further improving convergence and performance at the expense of additional computation. However, after a few iterations, the convergence rate (slope) is similar for either initialization, thanks to the convergence properties of WF, which

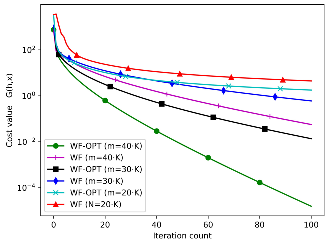

Fig. 3: Convergence for different number of samples.

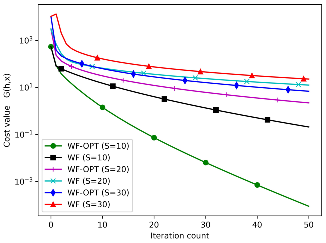

Fig. 4: Convergence for different number of sources.

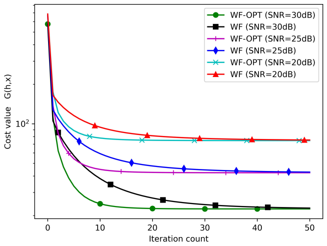

Fig. 5: Convergence for different SNR values.

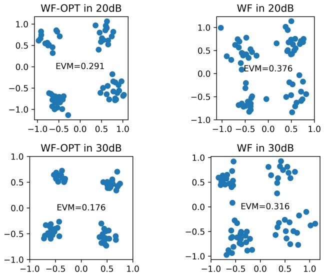

Fig. 6: Constellations at 10-th iteration for different SNR values.

enjoys geometrically well-behaved regions reachable after a few iterations with the random initialization, given sufficient measurements [16].

B. Performance with Different System Parameters

Fig. 3 shows performance for different sample sizes in noiseless scenarios. Here, both WF-OPT and WF reduce cost G in each iteration, but at different convergence rates. However, when data samples are shorter (e.g. $m = 20K$), the cost drops more slowly, requiring a significant number of iterations to converge. This is because both algorithms require enough statistical richness from samples to recover the underlying signals. In other words, given sufficient data samples without noise, both WF-OPT and WF will converge to near zero cost and recover near perfect signals. Fig. 4 shows that more sources S makes the problem more challenging and degrade the performance for both WF-OPT and WF. Additionally, we notice that, a larger number of sources S requires a larger number of samples to converge at a given rate for signal recovery by successful blind demixing.

Finally, Fig. 5 illustrates performance under channel noise at different SNR values. As expected, WF-OPT converges much faster than WF. We also observe that in the presence of noise, both WF-OPT and WF converge to the same non-diminishing residual cost value depending on SNR. It is clear that the residual cost value decreases with growing SNR, and can be interpreted as the magnitude of distortion. In addition, it is worth noting that at any given iteration, WF-OPT reaches a lower cost at the marginal expense of modest computation required for optimizing step sizes. Fig. 6 further corroborates this analysis, by showing recovered signal constellations of both WF-OPT and WF at iteration 10 and different SNR values. It shows that constellations recovered by WF-OPT exhibit better quality in earlier iterations in terms of Error Vector Magnitude (EVM), which measures the average magnitude of deviation from the ground-truth value of all symbols. As expected, recovery quality improves with increasing SNR.

VI. Conclusion

In this letter, we propose an efficient and practical blind demixing procedure called WF-OPT by leveraging the original WF algorithm. To accelerate convergence and enhance the performance of WF, we optimize the step sizes for the gradient descent stage of WF-OPT. Theoretical analysis shows that our proposed WF-OPT strictly decreases the cost value in each iteration with a bounded contraction ratio. Finally, our simulation results show that WF-OPT significantly outperforms WF in terms of convergence speed at a modest computation cost, presenting an improved WF solution to the blind demixing problem with overall lower computational complexity.

References

[1] H. Bristow, A. Eriksson, and S. Lucey, “Fast Convolutional Sparse Coding,” in *2013 IEEE Conf. Comput. Vision Pattern Recognit.*, 2013, pp. 391–398.

[2] G. Wunder, H. Boche, T. Strohmer, and P. Jung, “Sparse Signal Processing Concepts for Efficient 5G System Design,” *IEEE Access*, vol. 3, pp. 195–208, 2015.

[3] Z. Ding and Y. G. Li, *Blind Equalization and Identification*. New York, NY, USA: CRC Press, 2001.

[4] P. Jung, F. Krahmer, and D. Stöger, “Blind Demixing and Deconvolution at Near-Optimal Rate,” *IEEE Trans. Inf. Theory*, vol. 64, no. 2, pp. 704–727, 2018.

[5] M. B. McCoy and J. A. Tropp, “Sharp Recovery Bounds for Convex Demixing, with Applications,” *Found. Comput. Math.*, vol. 14, p. 503–567, 2014.

[6] B. Mariere, Z.-Q. Luo, and T. N. Davidson, “Blind Constant Modulus Equalization via Convex Optimization,” *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 805–818, 2003.

[7] A. Ahmed, B. Recht, and J. Romberg, “Blind Deconvolution Using Convex Programming,” *IEEE Trans. Inf. Theory*, vol. 60, no. 3, pp. 1711–1732, 2014.

[8] S. Ling and T. Strohmer, “Blind Deconvolution Meets Blind Demixing: Algorithms and Performance Bounds,” *IEEE Trans. Inf. Theory*, vol. 63, no. 7, pp. 4497–4520, 2017.

[9] J. Dong and Y. Shi, “Nonconvex Demixing from Bilinear Measurements,” *IEEE Trans. Signal Process.*, vol. 66, no. 19, pp. 5152–5166, 2018.

[10] S. Ling and T. Strohmer, “Regularized Gradient Descent: A Non-convex Recipe for Fast Joint Blind Deconvolution and Demixing,” *Inf. Inference: Journal of IMA*, vol. 8, no. 1, pp. 1–49, 03 2018.

[11] A.-J. van der Veen and A. Paulraj, “An Analytical Constant Modulus Algorithm,” *IEEE Trans. Signal Process.*, vol. 44, no. 5, pp. 1136–1155, 1996.

[12] E. J. Candes, X. Li, and M. Soltanolkotabi, “Phase Retrieval via Wirtinger Flow: Theory and Algorithms,” *IEEE Trans. Inf. Theory*, vol. 61, no. 4, p. 1985–2007, apr 2015.

[13] C. Ma, K. Wang, Y. Chi, and Y. Chen, “Implicit Regularization in Nonconvex Statistical Estimation: Gradient Descent Converges Linearly for Phase Retrieval, Matrix Completion and Blind Deconvolution,” *Found. Comput. Math.*, vol. 20, no. 3, pp. 451–632, aug 2019.

[14] Y. Liu, “An Efficient Method for Non-Convex Blind Deconvolution,” *IEEE Access*, vol. 7, pp. 113 663–113 674, 2019.

[15] K. Kreutz-Delgado, “The Complex Gradient Operator and the CR-Calculus,” June 2009. [Online]. Available: arXiv:0906.4835v1[math.OC]

[16] J. Dong and Y. Shi, “Blind Demixing via Wirtinger Flow with Random Initialization,” in *Proc. 22nd Int. Conf. Artif. Intell. Statist.*, vol. 89. PMLR, 2019, pp. 362–370.