# MISO Channel Estimation and Tracking from Received Signal Strength Feedback

Tianyu Qiu, Xiao Fu, *Member, IEEE*, Nicholas D. Sidiropoulos, *Fellow, IEEE*, and Daniel P. Palomar, *Fellow, IEEE*

***Abstract*—Downlink channel estimation is an important task in any wireless communication system, and 5G massive multiple-input multiple-output in particular—because the receiver must estimate and feed back to the transmitter a high-dimensional multiple-input single-output (MISO) vector channel for each receiving element. This is a serious burden in terms of mobile computation and power, as well as uplink communication overhead. The starting point of this paper is that all existing and emerging wireless communication systems provide basic Received Signal Strength (RSS) / Channel Quality Indicator (CQI) feedback to compensate for temporal channel variations. Is it possible to estimate and track the vector MISO channel from RSS/CQI feedback alone? This paper shows that the answer is affirmative, if one employs time-varying beamforming and phase modulation together with *phase retrieval* ideas from optics and crystallography. Three efficient algorithms that cover different model assumptions are proposed to track the vector MISO channel on the transmitter's side using only RSS/CQI feedback. Numerical simulation results under various settings validate the efficacy of the proposed algorithms in tracking a slowly time-varying vector MISO channel. Interestingly, this is the first application of phase retrieval where assuming independent and identically distributed Gaussian measurement vectors can be practically justified.**

***Index Terms*—Massive MIMO, channel estimation, channel tracking, received signal strength, channel quality indicator, limited feedback, phase retrieval.**

## I. Introduction

Channel estimation is a key task for any digital communication system. Channel knowledge is always needed for decoding at the receiver; higher rates and quality of service can be provided when the channel state can be made available to the transmitter, because then the transmission can be matched to the propagation channel, e.g., via appropriate beamforming or precoding. Acquiring accurate channel state information (CSI) is particularly important for massive multiple-input multiple-output (MIMO) systems that use a very large number of base station transmit antennas to enable highly selective spatial multiplexing [1], [2]. In the past several years, much effort has been invested into the area of channel estimation for massive MIMO—a variety of approaches such as those in [3]–[12] have been developed.

In frequency division duplex (FDD) systems, information about the channel state can only be acquired at the receiver, but should somehow be communicated to the transmitter in order to enable transmit beamforming and spatial multiplexing. Channel estimation and feedback is much more challenging in FDD massive MIMO systems, because the receiver must estimate and feed back to the transmitter a high-dimensional multiple-input single-output (MISO) vector channel for each receiving element. This creates a serious burden in terms of mobile computation and power, as well as uplink communication and signaling overhead [13]. Limited feedback schemes based on vector quantization (VQ) [7]–[9] use pilots to estimate the channel state, then quantize it using a previously designed vector codebook that is shared between the transmitter and the receiver, and feed back the quantization index to the transmitter. VQ-based limited feedback schemes [7]–[9] work well for conventional MIMO systems, but require codebook re-design when the channel characteristics change, and are not appealing for massive MIMO systems because their feedback overhead must scale linearly with the number of transmit antennas for a given performance level.

Another line of work uses "closed-loop" training to select a training sequence so that the downlink training overhead can be reudced [6]. This approach was later combined with a sparse channel model to further reduce feedback (uplink) overhead in [4]. Specifically, the downlink massive MIMO channel is mod-

Manuscript received July 2, 2017; revised October 12, 2017; accepted November 6, 2017. Date of publication January 11, 2018; date of current version February 13, 2018. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Adel Belouchrani. This work was supported in part by the University of Minnesota and the Hong Kong University of Science and Technology via bilateral collaboration funds, in part by the United States NSF Grants ECCS-1608961 and CIF-1525194, and in part by the Sponsorship Scheme for Targeted Strategic Partnership FP602 of the Hong Kong University of Science and Technology. *(Corresponding author: Nicholas D. Sidiropoulos.)*

T. Qiu was with the University of Minnesota, Minneapolis, MN 55455 USA. He is now with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong (e-mail: tqiu@ust.hk).

X. Fu was with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA. He is now with the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR 97331 USA (e-mail: xiao.fu@oregonstate.edu).

N. D. Sidiropoulos was with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA. He is now with the Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22904 USA (e-mail: nikos@virginia.edu).

D. P. Palomar is with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong (e-mail: palomar@ust.hk).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the author. This includes details of the three baseline algorithms in the main text, i.e., Kalman filter with perfect phase (KF-P), Kalman filter with alternating optimization (KF-A), and Kalman filter with smoothing (KF-S). The total size of the file is 167 KB.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2018.2791974

1053-587X © 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

eled using direction-of-departure (DOD), direction-of-arrival (DOA), and path loss for a few dominant paths [3], [4], [12]. This allows one to build up a dictionary that contains many possible combinations of DODs and DOAs in space. Then, sparse optimization approaches are employed to identify the "active" combinations, thereby achieving channel acquisition. The upshot is that the feedback can be quite ecomnonical since the receivers only need to send back the column indices of the constructed dictionary, which correspond to the active DODs and DOAs. The downside is that this approach only works for this specific channel model.

A very different channel estimation and tracking approach that shifts the burden of acquiring channel state information to the transmitter's side has been proposed in [10]. The idea of [10] is to exploit temporal channel correlation using locked tracking loops at both ends of the link. The receiver coarsely quantizes the innovation sequence down to one bit per real/imaginary part and feeds this pair of bits back to the transmitter. Both ends use a sign-of-innovation Kalman filter, maximum a posteriori probability, or minimum mean square error tracking loop to estimate the analog baseband channel from such 'frugal' feedback. The approach works very well, but requires knowledge of the temporal channel decorrelation rate / coherence time, as well as the spatial channel covariance matrix at the transmitter; a tracking loop at the receiver; and synchronization between the two ends. The scheme in [10] also uses (beamformed) pilots.

A method to estimate and track the spatial channel correlation matrix using very low-rate (again, 1-bit) feedback from the receiver has been recently proposed in [11]. The idea behind [11] is to exploit binary above/below target signal-to-interference-plus-noise ratio (SINR) feedback together with 'exploratory' beamforming to acquire increasingly accurate information about the downlink channel vector from the feedback bitstream. This requires adaptive design of the beamforming vector sequence and the target SINR thresholds used by the receiver, in order to enable channel correlation matrix identification, which is subsequently used for so-called *long-term* transmit beamforming.

The starting point of this paper is that all existing and emerging wireless communication systems provide basic Received Signal Strength (RSS) / Channel Quality Indicator (CQI) feedback, to compensate for temporal channel variations. RSS/CQI service is provided independently from and parallel to any channel estimation, for power control, rate adaptation, routing, and other network functions. But is it possible to estimate the vector MISO channel from RSS/CQI feedback alone? If the answer is affirmative, the upshot is very clear: first, since most wireless communication systems are already equipped with RSS/CQI feedback hardware and protocols, no drastic system-level modifications are needed to implement the algorithms; second, RSS/CQI feedback costs very little overhead, and thus is ideal for estimating and tracking downlink channels in *massive* MIMO systems.

*Contributions:* In this work, a novel channel estimation and tracking framework based on RSS/CQI feedback is proposed. The problem formulation leverages time-varying beamforming, phase modulation, and *phase retrieval* [14], [15]—a classical tool in optics and crystallography, which has recently drawn renewed interest in machine learning and statistical signal processing. Three effective algorithms are proposed to handle the problem of interest. The detailed contributions are as follows:

• *Novel Problem Formulation:* Our first major contribution lies in discovering the intriguing connection between phase retrieval and RSS/CQI feedback-based channel estimation. As mentioned, RSS/CQI-based channel estimation, if possible, entails low-overhead operations and convenient implementation. However, it is *a priori* unclear if this idea is fundamentally feasible. Our work shows that estimating and even tracking channels from RSS/CQI feedback alone is viable—using (pseudo-)random transmit beamforming vectors and building upon identifiability of random measurements-based phase retrieval [15]. Interestingly, using randomized (as opposed to Fourier) measurements is considered impractical in optics and crystallography, which are the major applications of phase retrieval. The situation is very different in our context, because the choice of transmit beamforming vectors is entirely up to the communication system designer.

• *Lightweight Tracking Algorithm:* Based on the connection between phase retrieval and static channel estimation, we take a step further to consider channel tracking. A forgetting factor-based formulation is proposed to take previous channel information into consideration while discounting its importance according to temporal distance to the current time slot. Consequently, a tracking algorithm reminiscent of recursive least squares (RLS) filtering [16] is obtained. To update the channel, the proposed algorithm mainly consists of a simple rank-one update of the pseudoinverse of a matrix and several matrix-vector multiplications, and thus is very lightweight. Convergence properties of the algorithm are also studied.

• *Model-Based High-Performance Tracking Algorithm:* We also consider the case where the channel progression model is known. Specifically, under a first-order auto-regressive (AR) model [17] that is widely employed for modeling the temporal correlation of wireless channels in consecutive time slots, a generalized maximum likelihood estimator (GMLE) is proposed to estimate and track the downlink channel. Two algorithms are proposed to handle the formulated GMLE, which exhibit even better performance compared to the RLS-based algorithm since they exploit prior information of the channel progression model.

The rest of this paper is organized as follows. Section II briefly introduces the phase retrieval problem and algorithms for handling it. The channel estimation and tracking problem is introduced in Section III. The first proposed algorithm, namely recursive phase retrieval, is presented in Section IV. Two more algorithms are derived in Section V based on an AR channel progression model. Simulations are provided in Section VI, and conclusions are drawn in Section VII.

*Notation:* Boldface upper case letters (e.g., $\mathbf{X}, \mathbf{A}$) denote matrices, boldface lower case letters (e.g., $\mathbf{x}, \mathbf{a}$) denote column vectors, and italics (e.g., $x, a$) denote scalars. $\mathbb{R}$ and $\mathbb{C}$ denote the field of real-valued numbers and the field of complex-valued numbers, respectively. For any complex-valued number $x$, $|x|$ denotes its magnitude, $\arg(x)$ denotes its phase, $\mathrm{Re}\{x\}$ denotes its real part, and $\mathrm{Im}\{x\}$ denotes its imaginary part. The superscripts $(\cdot)^T$, $(\cdot)^*$, and $(\cdot)^H$ denote transpose, conjugate, and conjugate transpose, respectively. $\mathbf{X}^{-1}$ ($\mathbf{X}^{\dagger}$) denotes the inverse (pseudoinverse) of a matrix $\mathbf{X}$. $\mathrm{Diag}(\mathbf{x})$ is a diagonal matrix with the vector $\mathbf{x}$ on its main diagonal. For a vector $\mathbf{x} \in \mathbb{C}^N$, $[\mathbf{x}]_n$ (or $x_n$) denotes its $n$th element, and $\|\mathbf{x}\|_2 := \sqrt{\sum_{n=1}^N |x_n|^2}$ denotes its Euclidean norm. For a matrix $\mathbf{X} \in \mathbb{C}^{M \times N}$, $[\mathbf{X}]_{mn}$ (or $x_{mn}$) denotes its element at the $m$th row and the $n$th column, and $\|\mathbf{X}\|_F := \sqrt{\sum_{m=1}^M \sum_{n=1}^N |x_{mn}|^2}$ denotes its Frobenius norm. As usual, $\mathbf{0}$ is the vector with all elements 0, $\mathbf{I}_N$ is the $N \times N$ identity matrix, and $\lambda_{\max}(\mathbf{X})$ denotes the largest eigenvalue of a Hermitian matrix $\mathbf{X}$.

## II. Preliminaries on Phase Retrieval

In this section, we briefly review the core ideas of phase retrieval [14], [15], which will help understand the channel estimation method to be proposed. Phase retrieval is the problem of retrieving a complex-valued signal $\mathbf{x}_o \in \mathbb{C}^N$ from the magnitude of several (noisy) linear measurements:

$$y_m = \left|\mathbf{a}_m^H \mathbf{x}_o\right| + n_m \in \mathbb{R},\ \forall m = 1, \ldots, M, \tag{1}$$

where the measuring vectors $\{\mathbf{a}_m \in \mathbb{C}^N\}_{m=1}^M$ are given and $\{n_m \in \mathbb{R}\}_{m=1}^M$ denotes additive noise. The phase retrieval techniques aim at retrieving the unknown signal $\mathbf{x}_o$ up to a global phase ambiguity—since $\mathbf{x}_o e^{j\phi}$ for any $\phi$ will yield the same magnitude information $\{y_m\}_{m=1}^M$, and thus the global phase ambiguity cannot be removed.

There are many ways of handling the problem. For example, one popular approach proposed by Gerchberg and Saxton [18] finds an estimate of the unknown signal as the solution of the following optimization problem:

$$\underset{\mathbf{x} \in \mathbb{C}^N}{\text{minimize}} \quad \sum_{m=1}^M \left(y_m - \left|\mathbf{a}_m^H \mathbf{x}\right|\right)^2. \tag{2}$$

This problem is not convex. Introducing new variables $\{\phi_m\}_{m=1}^M$ to represent the missing phase information, (2) is equivalent to

$$\underset{\mathbf{x} \in \mathbb{C}^N, \{\phi_m\}_{m=1}^M}{\text{minimize}} \quad \sum_{m=1}^M \left|y_m e^{j\phi_m} - \mathbf{a}_m^H \mathbf{x}\right|^2. \tag{3}$$

Even though (3) is not jointly convex in $\mathbf{x}$ and $\{\phi_m\}_{m=1}^M$, it is quadratic and convex in $\mathbf{x}$ when the phase completion variables $\{\phi_m\}_{m=1}^M$ are fixed. Therefore, the reformulated problem can be handled via alternating optimization. The solution to the $\mathbf{x}$-subproblem given $\{\phi_m\}_{m=1}^M$ is

$$\hat{\mathbf{x}} = \left(\sum_{m=1}^M \mathbf{a}_m \mathbf{a}_m^H\right)^{-1} \cdot \sum_{m=1}^M \mathbf{a}_m y_m e^{j\phi_m}; \tag{4}$$

while the solution to the $\{\phi_m\}_{m=1}^M$-subproblem given $\mathbf{x}$ is

$$\hat{\phi}_m = \arg\left(\mathbf{a}_m^H \mathbf{x}\right) - \pi \cdot 1_{y_m < 0},\ \forall m = 1, \ldots, M, \tag{5}$$

where $1_{y_m < 0}$ is equal to 1 if $y_m < 0$ and 0 otherwise—note that the magnitude information $y_m$ can be negative due to the additive noise $n_m$ according to (1). Together these yield the following simple update of the complex-valued variable $\mathbf{x}$ in Problem (2):

$$\mathbf{x}^{(t+1)} = \left(\sum_{m=1}^M \mathbf{a}_m \mathbf{a}_m^H\right)^{-1} \cdot \sum_{m=1}^M \mathbf{a}_m \left|y_m\right| e^{j \arg\left(\mathbf{a}_m^H \mathbf{x}^{(t)}\right)}, \tag{6}$$

where $\mathbf{x}^{(t)}$ denotes the $t$th iteration of $\mathbf{x}$. The simple algorithm presented in (6) is often referred to as the Gerchberg-Saxton algorithm. There are also variants of this algorithm, e.g., the AltMinPhase algorithm [19].

Phase retrieval is a classical problem in optics and physics, which has been studied since the 1960s [20]. There, $\mathbf{x}_o$ represents an object that an optical device aims to measure, and the measurement vectors $\{\mathbf{a}_m\}_{m=1}^M$ form an oversampled Fourier transform matrix $\mathbf{A} := [\mathbf{a}_1, \ldots, \mathbf{a}_M]^H$, due to the nature of the optical measurement apparatus. Many popular algorithms such as Fienup (or HIO) [21], Gerchberg-Saxton [18], and Holography [22], were proposed during the 1960s and the 1970s, and have served as workhorses for solving this problem in practice ever since. These algorithms have been quite successful—although properties of the algorithms (e.g., convergence and optimality) are not fully understood, especially when $\mathbf{A}$ represents a Fourier transform matrix. In recent years, there has been a surge of renewed interest in phase retrieval, as this problem can be nicely related to semidefinite relaxation and matrix completion, if the elements of $\mathbf{A}$ are i.i.d. and follow the zero-mean unit-variance Gaussian distribution. Using random measurements, many algorithms that guarantee successful phase retrieval with high probability have been proposed [19], [23]–[27]. Interestingly, there has been an (ongoing) debate about how practical the random measurements-based phase retrieval algorithms are—since most optical devices naturally measure magnitude of the Fourier transform of the objects [15].

## III. Problem Statement and Proposed Approach

Let us now turn our attention to the downlink channel estimation problem. Consider a multiple-input single-output (MISO) wireless communication system where the transmitter is equipped with $N$ transmit antennas and the receiver has one receive antenna. This is a typical downlink scenario in wireless communications, where the transmitter is often a base station, which has enough space and power to support many antennas, while the receiver is a mobile device that usually uses much fewer antennas. Under this setting, the wireless channel from the transmitter to a receiver is a complex-valued vector $\mathbf{h} \in \mathbb{C}^N$. In the next generation wireless communication systems, the base stations are expected to be equipped with a large number of antennas (or a massive antenna array). Consequently, $\mathbf{h}$ is a high-dimensional vector. Our goal here is to estimate and track $\mathbf{h}$ at the transmitter using a special type of limited feedback information. Specifically, we aim to estimate $\mathbf{h}$ from the Received Signal Strength (RSS) / Channel Quality Indicator (CQI) feedback, motivated by the fact that most existing communication devices can easily measure RSS/CQI and send it back to the transmitter without modifying the devices and protocols.

To formulate the problem, consider Fig. 1, which illustrates the time-slotted frame structure of the RSS/CQI feedback system. In time slot $m$, the transmitter sends a unit-power (constant-modulus) symbol $s(m) \in \mathbb{C}$ to the receiver, using beamforming vector $\mathbf{w}(m) \in \mathbb{C}^N$, and afterwards the transmitter transmits the data symbols, using different beamforming vectors based on the channel estimation result. Note that any constant modulus

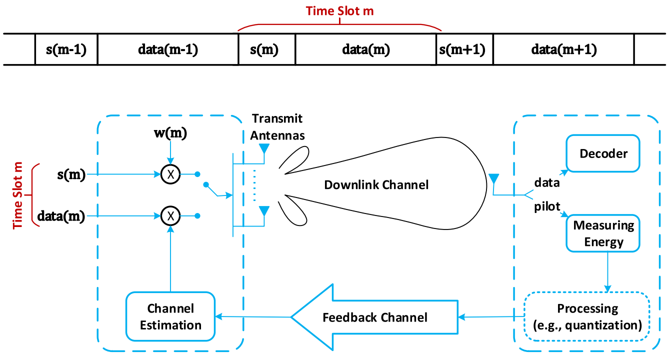


Fig. 1. The time-slotted frame structure of the Received Signal Strength (RSS) / Channel Quality Indicator (CQI) feedback system, adapted from [10].

modulation, such as coherently decoded phase-shift keying (PSK) or incoherently decoded differential PSK (DPSK), or even analog frequency modulation will serve our purposes. The channel for time slot $m$ is $\mathbf{h}(m) \in \mathbb{C}^N$, modeling the vector MISO channel between the $N$ transmit antennas and the single receive antenna. The received signal corresponding to $s(m)$ is then

$$z(m) = \mathbf{w}^H(m)\mathbf{h}(m)s(m) + v(m) \in \mathbb{C}, \tag{7}$$

where $v(m) \sim \mathcal{CN}(0, \sigma_v^2)$ models the additive white Gaussian noise. The receiver feeds back to the transmitter the RSS $|z(m)|^2$, in analog or digital form, through a low-rate analog or digital feedback channel. Note that RSS is related to CQI via $(|z(m)|^2 - \sigma_v^2)/\sigma_v^2$; and that instead of $|z(m)|^2$ the receiver can feed back $|z(m)|$. The goal is then to estimate and track the channel vector $\mathbf{h}(M)$ at each time slot $M = 1, 2, \ldots$ given the current and all previous feedback signal magnitude information $\{|z(m)|\}_{m=1}^M$.

The feedback signal magnitude information satisfies

$$\begin{aligned} y(m) &:= |z(m)| = |z(m)s^*(m)| \\ &= \left|\mathbf{w}^H(m)\mathbf{h}(m) + v(m)s^*(m)\right|, \ \forall m, \end{aligned} \tag{8}$$

where we have used $s(m)s^*(m) = 1$, owing to the PSK modulation. Introducing auxiliary phase completion variables $\{\phi(m)\}_m$, we have

$$y(m)e^{j\phi(m)} = \mathbf{w}^H(m)\mathbf{h}(m) + n(m), \ \forall m. \tag{9}$$

Note that $v(m)$ is i.i.d. circularly symmetric complex Gaussian, and so is $n(m) := v(m)s^*(m)$—phase rotations do not affect the noise statistics. Therefore, we propose to estimate the channel vectors $\{\mathbf{h}(m)\}_{m=1}^M$ via

$$\underset{\{\mathbf{h}(m),\phi(m)\}_{m=1}^M}{\text{minimize}} \ \sum_{m=1}^M \left|y(m)e^{j\phi(m)} - \mathbf{w}^H(m)\mathbf{h}(m)\right|^2. \tag{10}$$

The above is equivalent to

$$\underset{\{\mathbf{h}(m)\}_{m=1}^M}{\text{minimize}} \ \sum_{m=1}^M \left(y(m) - \left|\mathbf{w}^H(m)\mathbf{h}(m)\right|\right)^2, \tag{11}$$

which is a phase retrieval problem with time-varying variables.

From (10), we note that if the channel is static for all of the collected measurements, i.e., $\mathbf{h}(m) = \mathbf{h}, \forall m$, then, Problem (11) is exactly a phase retrieval problem as in (2)—and it is clear that the channel is identifiable (up to a global phase ambiguity) from the RSS feedback alone.[1] This connection is already very interesting, yet we still have two major difficulties in practice. First, phase retrieval techniques typically need more than $4N$ measurements to retrieve an $N$-dimensional complex-valued vector [28], [29], and this is costly in terms of uplink communication overhead since $N$ is the number of transmit antennas. Second, the channel is never static in practice, and thus directly applying the phase retrieval techniques may not return satisfactory results. Fortunately, wireless MISO channels usually exhibit strong temporal correlation, which can be exploited to circumvent these difficulties.[2] In the next two sections, we propose three phase retrieval-inspired channel tracking algorithms. Under the tracking mode, temporal channel variations are taken into consideration—the channel vector is updated using past channel information and the current feedback—which greatly reduces the uplink overhead.

One side comment is that there are various different formulations and algorithms for phase retrieval, such as the semidefinite relaxation based approach [23], Gerchberg and Saxton's formulation [18] (i.e., Problem (10)), and some recently proposed robust formulations [31]. In this work, we will focus on modifying the formulation in (10) to come up with efficient tracking algorithms—since the structure of (10) is particularly friendly for designing computationally lightweight tracking algorithms, as we will see shortly.

*Remark 1:* Before going into the algorithm design part, we would like to make an interesting remark. Unlike the phase retrieval problem where the measurement vectors $\{\mathbf{a}_m\}_m$ are determined by the measurement apparatus (usually, they are rows of the discrete Fourier transform matrix), in our context, the transmitter actually has the freedom of picking the beamforming vectors $\{\mathbf{w}(m)\}_m$ to suit its purposes. One choice is to select pseudo-random Gaussian vectors, i.i.d. across space and time. To the best of our knowledge, this is very likely the only practical application of phase retrieval where the assumption of i.i.d. complex Gaussian measurement vectors is reasonable.

## IV. Recursive Phase Retrieval

### A. Forgetting Factor-Based Formulation

To estimate and track the channel, we start with Problem (10). Note that the problem is extremely ill-posed in general: For every $\mathbf{h}(m) \in \mathbb{C}^N$, only one measurement $y(m) \in \mathbb{R}$ is available. Under such an underdetermined setting, retrieving $\mathbf{h}(m)$ is impossible—even without phase loss. In wireless communications, however, because the channels are strongly correlated

[1] The global ambiguity is easy to resolve in practice; e.g., in the transmission stage, one can always let the first transmit bit to be 1 as part of the protocol. Then the global phase ambiguity can be read out by looking at the first received bit.

[2] A different way of modeling similarity is the low-rank phase retrieval formulation in [30], however linear dependence (channel vectors living in a low-dimensional subspace) neither implies nor is implied by strong temporal correlation, and the latter is far more widely accepted in wireless communication.

over time, channel estimation and tracking are feasible via exploiting information obtained at previous time slots.

Our first idea is to take insights from adaptive filtering, in particular, the recursive least squares (RLS) filter [16]. To be specific, we propose to modify the objective function in (10) to the following estimation criterion at time slot $M$:

$$\underset{\mathbf{h},\{\phi(m)\}_{m=1}^{M}}{\text{minimize}} \sum_{m=1}^{M} \lambda^{M-m} \left| y(m) e^{j\phi(m)} - \mathbf{w}^H(m)\mathbf{h} \right|^2 . \quad (12)$$

Denoting $(\mathbf{h}^\star, \{\phi^\star(m)\}_m^M)$ as the solution of Problem (12), we let $\hat{\mathbf{h}}(M) = \mathbf{h}^\star$ be our estimate of the vector MISO channel at time slot $M$. In (12), $\lambda \in (0, 1]$ is a forgetting factor weighting the squared error terms for past time slots; the closer $m$ is to the current time slot $M$, the larger weight is put on the corresponding squared error term. In practice, a large $\lambda \in (0, 1]$ is used when the channel is believed to be changing slowly; e.g., when the channel is static, $\lambda = 1$ recovers the phase retrieval formulation as in (10). In other words, we effectively treat all the channel vectors as $\mathbf{h}(m) = \mathbf{h}(M)$ for $m = 1, \ldots, M-1$, but discount the 'credibility' of the measurements obtained at time slots that are far away from slot $M$. This way, we have made use of the temporal channel correlation to circumvent the underdetermination issue in (10), while the temporal channel variations are also taken into account—which is inspired by the insight of RLS filtering. The difference is that in our formulation, we have a phase compensation element $e^{j\phi(m)}$ for each time slot, resulting in a *recursive phase retrieval* formulation.

### B. An Efficient Recursive Algorithm

To handle (12), our idea is to alternately solve Problem (12) with respect to (w.r.t.) $\{\phi(m)\}_{m=1}^M$ and $\mathbf{h}$ when fixing the other, respectively. To explain the algorithm, we define $\mathbf{y}_M := [y(1), \ldots, y(M)]^T \in \mathbb{R}^M$, $\boldsymbol{\varphi}_M := [\phi(1), \ldots, \phi(M)]^T \in \mathbb{R}^M$, and $\boldsymbol{\lambda}_M := [\lambda^{\frac{M-1}{2}}, \lambda^{\frac{M-2}{2}}, \ldots, \lambda^{\frac{M-M}{2}}]^T \in \mathbb{R}^M$. Using these notations, (12) can be re-written in a more compact form:

$$\underset{\mathbf{h},\boldsymbol{\varphi}_M}{\text{minimize}} \quad \left\| \mathbf{D}_M^\lambda \mathbf{D}_M^y \mathbf{e}^{j\boldsymbol{\varphi}_M} - \mathbf{D}_M^\lambda \mathbf{W}_M \mathbf{h} \right\|_2^2 , \quad (13)$$

where $\mathbf{D}_M^\lambda := \text{Diag}(\boldsymbol{\lambda}_M)$, $\mathbf{D}_M^y := \text{Diag}(\mathbf{y}_M)$, and $\mathbf{W}_M := [\mathbf{w}(1), \ldots, \mathbf{w}(M)]^H \in \mathbb{C}^{M \times N}$ is a matrix consisting of the beamforming vectors at the $M$th and the previous time slots. The vector $\mathbf{e}^{j\boldsymbol{\varphi}_M}$ is defined as $\mathbf{e}^{j\boldsymbol{\varphi}_M} = [e^{j\phi(1)}, \ldots, e^{j\phi(M)}]^T$.

Denoting the objective function in (13) as $f(\mathbf{h}, \boldsymbol{\varphi}_M)$, our algorithm is a two-block coordinate descent algorithm as follows:

$$\boldsymbol{\varphi}_M^{(t+1)} \leftarrow \arg\min_{\boldsymbol{\varphi}_M} f\left(\mathbf{h}^{(t)}, \boldsymbol{\varphi}_M\right), \quad (14a)$$

$$\mathbf{h}^{(t+1)} \leftarrow \arg\min_{\mathbf{h}} f\left(\mathbf{h}, \boldsymbol{\varphi}_M^{(t+1)}\right), \quad (14b)$$

where $(\mathbf{h}^{(t)}, \boldsymbol{\varphi}_M^{(t)})$ denotes the $t$th iterate of the variables. Let us start with the partial minimization w.r.t. $\boldsymbol{\varphi}_M$, i.e., Problem (14a). The problem is nonconvex, but the optimal solution is easy to find, i.e.,

$$\boldsymbol{\varphi}_M^{(t+1)} = \arg\left(\mathbf{W}_M \mathbf{h}^{(t)}\right), \quad (15)$$

where the $\arg(\mathbf{x})$ is a vector that holds the phases of the elements of the vector $\mathbf{x}$. When fixing $\boldsymbol{\varphi}_M$, the subproblem in (14b) is a least squares problem, and the solution is

$$\mathbf{h}^{(t+1)} = \left(\mathbf{W}_M^H \mathbf{D}_M^\lambda \mathbf{D}_M^\lambda \mathbf{W}_M\right)^{-1} \mathbf{W}_M^H \mathbf{D}_M^\lambda \mathbf{D}_M^\lambda \mathbf{D}_M^y \mathbf{e}^{j\boldsymbol{\varphi}_M^{(t+1)}} . \quad (16)$$

Putting together (15) and (16), the update rule of $\mathbf{h}$ can be expressed as follows:

$$\mathbf{h}^{(t+1)} = \left(\mathbf{W}_M^H \mathbf{D}_M^\lambda \mathbf{D}_M^\lambda \mathbf{W}_M\right)^{-1} \mathbf{W}_M^H \mathbf{D}_M^\lambda \mathbf{D}_M^\lambda \mathbf{D}_M^y \cdot \mathbf{e}^{j \arg\left(\mathbf{W}_M \mathbf{h}^{(t)}\right)} . \quad (17)$$

It is worth mentioning that the algorithm in (17) can be carried out quite efficiently in practice, which is reminiscent of RLS. To see this, let us define the matrix $\mathbf{Q}_M$ as

$$\mathbf{Q}_M := \mathbf{W}_M^H \mathbf{D}_M^\lambda \mathbf{D}_M^\lambda \mathbf{W}_M \in \mathbb{C}^{N \times N}. \quad (18)$$

It is readily seen that

$$\mathbf{Q}_M = \lambda \mathbf{Q}_{M-1} + \mathbf{w}(M)\mathbf{w}^H(M). \quad (19)$$

According to the matrix inversion lemma, $\mathbf{Q}_M^{-1}$ can be updated from $\mathbf{Q}_{M-1}^{-1}$ as

$$\mathbf{Q}_M^{-1} = \lambda^{-1} \left( \mathbf{Q}_{M-1}^{-1} - \frac{\mathbf{Q}_{M-1}^{-1}\mathbf{w}(M)\mathbf{w}^H(M)\mathbf{Q}_{M-1}^{-1}}{\lambda + \mathbf{w}^H(M)\mathbf{Q}_{M-1}^{-1}\mathbf{w}(M)} \right). \quad (20)$$

By updating $\mathbf{Q}_M^{-1}$ using the above, the inversion of an $N \times N$ matrix—which typically takes $\mathcal{O}(N^3)$ flops—can be avoided. The rank-one update of $\mathbf{Q}_M^{-1}$ in (20) takes only $\mathcal{O}(N^2)$ flops, which is one order of magnitude cheaper compared to directly inverting $\mathbf{Q}_M$. Similarly, the term $\mathbf{W}_M^H \mathbf{D}_M^\lambda \mathbf{D}_M^\lambda \mathbf{D}_M^y$ in (17) can also be updated efficiently from the one at the previous time slot; i.e., we have

$$\mathbf{P}_M := \mathbf{W}_M^H \mathbf{D}_M^\lambda \mathbf{D}_M^\lambda \mathbf{D}_M^y = \left[\lambda \mathbf{P}_{M-1},\ y(M)\mathbf{w}(M)\right]. \quad (21)$$

Therefore, the major components for updating $\mathbf{h}$ in (17) can be obtained from the previous time slot using recursive updates.

Note that $\mathbf{Q}_M \in \mathbb{C}^{N \times N}, \forall M$. When $M < N$, $\mathbf{Q}_M$ is rank deficient so $\mathbf{Q}_M^{-1}$ does not exist. Hence the update should not start before the $N$th time slot. The procedure of estimating the channel vector $\mathbf{h}(M)$ at time slot $M$ is summarized in Algorithm 1. The initialization $\mathbf{h}^{(0)}$ is set to be $\hat{\mathbf{h}}(M-1)$, i.e., the channel vector estimated at the previous time slot. As long as the channel changes slowly, such an initialization is very effective, as will be shown in the simulations. One can see that once $\mathbf{Q}_M^{-1}$ and $\mathbf{P}_M$ are updated, the remaining operations are simply taking phases of a vector and multiplying the corresponding phase vector by a matrix (cf. line 7 in Algorithm 1). Therefore, the algorithm is very lightweight and ideal for channel tracking.

### C. Convergence Properties

Algorithm 1 is a two-block alternating optimization algorithm. Therefore, the objective value of $f(\mathbf{h}, \boldsymbol{\varphi}_M)$ decreases at each iteration, and thus the objective sequence $\{f(\mathbf{h}^{(t)}, \boldsymbol{\varphi}_M^{(t)})\}_t$ converges since the cost function is bounded from below. In terms of convergence of the solution sequence $\{(\mathbf{h}^{(t)}, \boldsymbol{\varphi}_M^{(t)})\}_t$,

**Algorithm 1:** Recursive Phase Retrieval (RPR) at time slot $M$.

```
Input: $\lambda$, $\mathbf{W}_M$, $\mathbf{Q}_{M-1}^{-1}$, $\mathbf{P}_{M-1}$, $y(M)$, and $\hat{\mathbf{h}}(M-1)$;
$\mathbf{Q}_M^{-1} \leftarrow \lambda^{-1}\left(\mathbf{Q}_{M-1}^{-1} - \frac{\mathbf{Q}_{M-1}^{-1}\mathbf{w}(M)\mathbf{w}^H(M)\mathbf{Q}_{M-1}^{-1}}{\lambda+\mathbf{w}^H(M)\mathbf{Q}_{M-1}^{-1}\mathbf{w}(M)}\right)$;
$\mathbf{P}_M \leftarrow [\lambda\mathbf{P}_{M-1},\ y(M)\mathbf{w}(M)]$;
$\mathbf{H}_M \leftarrow \mathbf{Q}_M^{-1}\mathbf{P}_M$;
$\mathbf{h}^{(0)} \leftarrow \hat{\mathbf{h}}(M-1)$;
$t \leftarrow 0$;
repeat
    $\mathbf{h}^{(t+1)} \leftarrow \mathbf{H}_M e^{j\arg\left(\mathbf{W}_M\mathbf{h}^{(t)}\right)}$;
    $t \leftarrow t+1$;
until stopping criterion;
Output: $\hat{\mathbf{h}}(M) \leftarrow \mathbf{h}^{(t)}$.
```

however, the classical analytical tools for generic alternating optimization such as those in [32]–[34] cannot be applied to show convergence. The reason is that the analyses in [32]–[34] rely on convexity of the subproblems, but the $\boldsymbol{\varphi}_M$-subproblem in (14a) is not convex. Fortunately, since our algorithm only has two blocks, convergence of $\{(\mathbf{h}^{(t)}, \boldsymbol{\varphi}_M^{(t)})\}_t$ can still be shown, following the argument of the maximum block improvement (MBI) framework [35] that does not require the subproblems to be convex. Formally, we have the following statement:

*Proposition 1:* At each time slot, every limit point of the solution sequence produced by Algorithm 1 is a stationary point of Problem (12).

The idea of proof is straightforward and thus is omitted: as observed in [36], [37], block coordinate descent and MBI are equivalent in the two-block case. Hence, following the convergence results of MBI, Algorithm 1 is readily shown to converge to a stationary point of Problem (13).

## V. Generalized Maximum Likelihood Estimation

The RPR algorithm proposed in the previous section did not make any assumption on the channel evolution model. The upshot is that the RPR algorithm is quite general—it can handle all the scenarios where the channel vector does not change too quickly. On the other hand, if some prior information on the temporal correlation of the channels is known, making use of such information may greatly improve the performance—while the RPR algorithm is not able to exploit such prior information. In this section, we propose channel tracking algorithms that are tailored to the first-order temporal auto-regressive (AR) channel model. The model is widely employed in the literature for modeling the temporal progression of wireless channels and thus is well-motivated [10], [38]–[41].

### A. Problem Formulation Under an AR Model

We consider the following AR model between two consecutive time slots:

$$\mathbf{h}(m) = \alpha\mathbf{h}(m-1) + \mathbf{u}(m),\ \forall m = 1, 2, \ldots, \tag{22}$$

where $\alpha \in (0,1)$ is a constant and $\mathbf{u}(m) \sim \mathcal{CN}(\mathbf{0}, \sigma_u^2\mathbf{I}_N)$ models the spatially uncorrelated process noise [39], [41]. Under the considered AR model, one can easily incorporate previously obtained measurements $y(m)$ for $m = 1, \ldots, M-1$ into estimating $\mathbf{h}(M)$ at time slot $M$. To see this, first notice that the channel vector $\mathbf{h}(m)$ for past time slots indexed by $m < M$ can be written as a function of $\mathbf{h}(M)$:

$$\mathbf{h}(m) = \alpha^{m-M}\mathbf{h}(M) - \sum_{k=0}^{M-m-1}\alpha^{m-M+k}\mathbf{u}(M-k). \tag{23}$$

Therefore, every measurement $y(m)$ for $m = 1, \ldots, M$ can be represented as a function of $\mathbf{h}(M)$; i.e., for $m = 1, \ldots, M$, we have

$$\begin{aligned} y(m) &= \left|\mathbf{w}^H(m)\mathbf{h}(m) + n(m)\right| \\ &= \left|\alpha^{m-M}\mathbf{w}^H(m)\mathbf{h}(M) + \tilde{n}(m)\right|, \end{aligned} \tag{24}$$

in which we define $\tilde{n}(M) := n(M)$ and

$$\tilde{n}(m) := n(m) - \sum_{k=0}^{M-m-1}\alpha^{m-M+k}\mathbf{w}^H(m)\mathbf{u}(M-k) \tag{25}$$

for $m = 1, \ldots, M-1$.

One may notice that the expression of $y(m)$ in (24) looks similar to that in (8) and attempt to construct an estimator as follows:

$$\underset{\mathbf{h}(M),\{\phi(m)\}_{m=1}^M}{\text{minimize}} \sum_{m=1}^{M}\left|y(m)e^{j\phi(m)} - \alpha^{m-M}\mathbf{w}^H(m)\mathbf{h}(M)\right|^2. \tag{26}$$

The above is a reasonable option, but it is likely to be suboptimal (in terms of maximum likelihood estimation) because the noise $\tilde{n}(m)$ is colored across different $m$'s. Fortunately, since the statistics of $\mathbf{u}(m)$ are available, one can easily derive the statistics of $\tilde{n}(m)$—which can be utilized to construct a (generalized) maximum likelihood estimator of $\mathbf{h}(M)$. To be specific, we first consider the mean of $\tilde{n}(m)$, which is readily seen to be zero because $n(m)$, $\mathbf{w}(m)$, and $\mathbf{u}(M-k)$ are all zero-mean and independent of each other. Second, the expected power of $\tilde{n}(m)$ can be expressed as follows:

$$\begin{aligned} &\mathbb{E}\left\{\tilde{n}(m)\tilde{n}^*(m)\right\} \\ &= \mathrm{Var}\left\{n(m)\right\} + \mathrm{Var}\left\{\sum_{k=0}^{M-m-1}\alpha^{m-M+k}\mathbf{w}^H(m)\mathbf{u}(M-k)\right\} \\ &= \sigma_v^2 + \sum_{k=0}^{M-m-1}\alpha^{2(m-M+k)}\mathrm{Var}\left\{\mathbf{w}^H(m)\mathbf{u}(M-k)\right\} \\ &= \sigma_v^2 + \frac{\alpha^{2(m-M)}-1}{1-\alpha^2}\sigma_u^2\mathbf{w}^H(m)\mathbf{w}(m), \end{aligned} \tag{27}$$

where we have used the fact that $n(m)$, $\mathbf{w}(m)$, and $\mathbf{u}(m)$ are independent and $n(m) = v(m)s^*(m) \sim \mathcal{CN}(0, \sigma_v^2)$. Similarly, the cross-correlation of $\tilde{n}(p)$ and $\tilde{n}(q)$ for $1 \leq p \neq q \leq M$ is

$$\mathbb{E}\left\{\tilde{n}(p)\tilde{n}^*(q)\right\} = \frac{\alpha^{p+q-2M} - \alpha^{-|q-p|}}{1-\alpha^2}\sigma_u^2\mathbf{w}^H(p)\mathbf{w}(q). \tag{28}$$

Hence, at time slot $M$, the noise vector $\tilde{\mathbf{n}}(M) = [\tilde{n}(1), \ldots, \tilde{n}(M)]^T \in \mathbb{C}^M$ is a multivariate complex Gaussian vector with zero mean and covariance matrix $\mathbf{C}_M$ with elements

$$[\mathbf{C}_M]_{p,q} = \mathbb{E}\left\{\tilde{n}(p)\tilde{n}^*(q)\right\},\ 1 \leq p, q \leq M. \tag{29}$$

Therefore, given $y(m)$ for $m = 1, \ldots, M$ and the derived statistics of the noise vector $\tilde{\mathbf{n}}(M)$, the *generalized maximum likelihood estimator* (GMLE) of $\mathbf{h}(M)$ is as follows:

$$\begin{aligned}&\left(\hat{\mathbf{h}}(M), \{\phi(m)\}_m\right)\\ &= \arg\min_{\mathbf{h}, \boldsymbol{\varphi}_M} \left\|\mathbf{C}_M^{-1/2}\left(\mathbf{D}_M^y \mathbf{e}^{j\boldsymbol{\varphi}_M} - \mathbf{D}_M^\alpha \mathbf{W}_M \mathbf{h}\right)\right\|_2^2, \end{aligned}\tag{30}$$

where $\mathbf{D}_M^\alpha := \mathrm{Diag}(\boldsymbol{\alpha}_M)$ is a diagonal matrix with main diagonal $\boldsymbol{\alpha}_M := [\alpha^{1-M}, \alpha^{2-M}, \ldots, \alpha^{M-M}]^T \in \mathbb{R}^M$ and $\mathbf{D}_M^y$, $\boldsymbol{\varphi}_M$, and $\mathbf{W}_M$ are defined as before. Note that we call the above estimator GMLE instead of MLE because we have introduced a phase completion term $\mathbf{e}^{j\boldsymbol{\varphi}_M}$, which is different from the classical MLE.

### *B. Alternating Optimization for GMLE*

As in the forgetting factor-based formulation, we also employ alternating optimization to handle the formulated GMLE; that is, we update $\mathbf{h}$ and $\boldsymbol{\varphi}_M$ by the following rules:

$$\boldsymbol{\varphi}_M^{(t+1)} \leftarrow \arg\min_{\boldsymbol{\varphi}_M} g\left(\mathbf{h}^{(t)}, \boldsymbol{\varphi}_M\right), \tag{31a}$$

$$\mathbf{h}^{(t+1)} \leftarrow \arg\min_{\mathbf{h}} g\left(\mathbf{h}, \boldsymbol{\varphi}_M^{(t+1)}\right), \tag{31b}$$

where $g(\mathbf{h}, \boldsymbol{\varphi}_M)$ denotes the objective function in (30), and we solve the two subproblems in (31a) and (31b) alternately until some convergence criterion is reached.

Solving the $\mathbf{h}$-subproblem in (31b) is simple, since it is again an unconstrained quadratic programming problem that has the following analytical solution:

$$\begin{aligned}\mathbf{h}^{(t+1)} &= \left(\mathbf{C}_M^{-1/2}\mathbf{D}_M^\alpha \mathbf{W}_M\right)^\dagger \mathbf{C}_M^{-1/2}\mathbf{D}_M^y \mathbf{e}^{j\boldsymbol{\varphi}_M^{(t+1)}}\\ &= \left(\mathbf{W}_M^H \mathbf{D}_M^\alpha \mathbf{C}_M^{-1}\mathbf{D}_M^\alpha \mathbf{W}_M\right)^{-1}\mathbf{W}_M^H \mathbf{D}_M^\alpha \mathbf{C}_M^{-1}\mathbf{D}_M^y \mathbf{e}^{j\boldsymbol{\varphi}_M^{(t+1)}}\\ &= \tilde{\mathbf{H}}_M \mathbf{e}^{j\boldsymbol{\varphi}_M^{(t+1)}},\end{aligned}\tag{32}$$

where $\tilde{\mathbf{H}}_M := (\mathbf{W}_M^H \mathbf{D}_M^\alpha \mathbf{C}_M^{-1}\mathbf{D}_M^\alpha \mathbf{W}_M)^{-1}\mathbf{W}_M^H \mathbf{D}_M^\alpha \mathbf{C}_M^{-1}\mathbf{D}_M^y$. When $\mathbf{h}$ is fixed, the subproblem in (31a) is equivalent to the following:

$$\underset{\boldsymbol{\varphi}_M}{\text{minimize}} \quad \left(\mathbf{e}^{j\boldsymbol{\varphi}_M}\right)^H \mathbf{F}_M \mathbf{e}^{j\boldsymbol{\varphi}_M} - 2\mathrm{Re}\left\{\left(\mathbf{e}^{j\boldsymbol{\varphi}_M}\right)^H \mathbf{G}_M \mathbf{h}^{(t)}\right\}, \tag{33}$$

where we have $\mathbf{F}_M := \mathbf{D}_M^y \mathbf{C}_M^{-1}\mathbf{D}_M^y$ and $\mathbf{G}_M := \mathbf{D}_M^y \mathbf{C}_M^{-1}\mathbf{D}_M^\alpha \mathbf{W}_M$. Problem (33) has no analytical solution in general. One may employ some standard nonlinear programming algorithms, e.g., gradient descent, to handle it since the objective function is continuously differentiable w.r.t. $\boldsymbol{\varphi}_M$. However, this leads to an iterative process to solve the $\boldsymbol{\varphi}_M$-subproblem, which is undesired in terms of efficiency. In addition, since this subproblem is nonconvex, directly solving it has no guarantee of the solution quality and thus the overall alternating optimization algorithm is not guaranteed to converge to a meaningful point (e.g., a KKT point) in general.

Here, we provide an algorithm that updates $\boldsymbol{\varphi}_M$ using a simple update strategy while guaranteeing convergence of the overall alternating optimization procedure. First, let us re-write Problem (33) as

$$\begin{aligned}&\underset{\mathbf{u}\in\mathbb{C}^M}{\text{minimize}} && \mathbf{u}^H \mathbf{F}_M \mathbf{u} - 2\mathrm{Re}\left\{\mathbf{u}^H \mathbf{G}_M \mathbf{h}^{(t)}\right\}\\ &\text{subject to} && |u_m| = 1, \forall m = 1, \ldots, M.\end{aligned}\tag{34}$$

Notice that Problem (34) and Problem (33) are equivalent: the solutions of the latter have a one-to-one correspondence with those of the former, i.e., $\boldsymbol{\varphi}_M = \arg(\mathbf{u})$. Therefore, the update in (31a) can be written as

$$\mathbf{u}^{(t+1)} \leftarrow \arg\min_{|u_m|=1,\forall m} g\left(\mathbf{h}^{(t)}, \mathbf{u}\right), \tag{35}$$

where we have substituted $\mathbf{e}^{j\boldsymbol{\varphi}_M}$ with $\mathbf{u}$. Re-writing the problem does not reduce the difficulty—but it allows us to handle it using a simpler update strategy. Specifically, we propose to update $\mathbf{u}$ via solving a majorizing surrogate of Problem (34) [42], [43]:

$$\mathbf{u}^{(t+1)} \leftarrow \arg\min_{|u_m|=1,\forall m} \bar{g}\left(\mathbf{h}^{(t)}, \mathbf{u}\right). \tag{36}$$

The objective function in (36) is

$$\begin{aligned}\bar{g}\left(\mathbf{h}^{(t)}, \mathbf{u}\right) &:= g\left(\mathbf{h}^{(t)}, \mathbf{u}^{(t)}\right)\\ &\quad + \mathrm{Re}\left\{\left(\nabla_{\mathbf{u}} g\left(\mathbf{h}^{(t)}, \mathbf{u}^{(t)}\right)\right)^H\left(\mathbf{u} - \mathbf{u}^{(t)}\right)\right\}\\ &\quad + \frac{1}{\rho}\left\|\mathbf{u} - \mathbf{u}^{(t)}\right\|_2^2,\end{aligned}\tag{37}$$

where $0 < \rho < 1/\lambda_{\max}(\mathbf{F}_M)$ is a pre-specified parameter and

$$\nabla_{\mathbf{u}} g\left(\mathbf{h}^{(t)}, \mathbf{u}^{(t)}\right) = 2\mathbf{F}_M \mathbf{u}^{(t)} - 2\mathbf{G}_M \mathbf{h}^{(t)}. \tag{38}$$

Note that Problem (36) has a closed-form solution [44], [45]:

$$\mathbf{u}^{(t+1)} = \mathbf{e}^{j\arg\left(\mathbf{u}^{(t)} - \frac{\rho}{2}\nabla_{\mathbf{u}} g\left(\mathbf{h}^{(t)}, \mathbf{u}^{(t)}\right)\right)}. \tag{39}$$

The algorithm is summarized in Algorithm 2. Since we essentially employ a one-step gradient projection to update $\mathbf{u}$, we name the algorithm GMLE-G. There are several good properties of the algorithm. First, the update of $\mathbf{u}$ is very simple. Second, the overall alternating optimization procedure is guaranteed to converge to a KKT point of Problem (30). To be specific, we have the following statement:

*Proposition 2:* At each time slot, every limit point of the solution sequence produced by Algorithm 2 is a Karush-Kuhn-Tucker (KKT) point of Problem (30) given that $0 < \rho < 1/\lambda_{\max}(\mathbf{F}_M)$.

Note that the above convergence property is not obvious: the constraint of the $\mathbf{u}$-subproblem (34) is nonconvex, and thus most popular inexact alternating optimization frameworks [32], [33] cannot cover it. Nevertheless, when $0 < \rho < 1/\lambda_{\max}(\mathbf{F}_M)$,

**Algorithm 2:** GMLE-G under the AR model (at time slot $M$).

```
Input: $\mathbf{W}_M, \alpha, \sigma_u^2, \sigma_v^2, \mathbf{y}_M, \beta \in (0,1)$, and $\hat{\mathbf{h}}(M-1)$;
$\mathbf{F}_M \leftarrow \mathbf{D}_M^y \mathbf{C}_M^{-1} \mathbf{D}_M^y$;
$\mathbf{G}_M \leftarrow \mathbf{D}_M^y \mathbf{C}_M^{-1} \mathbf{D}_M^\alpha \mathbf{W}_M$;
$\tilde{\mathbf{H}}_M \leftarrow \left(\mathbf{W}_M^H \mathbf{D}_M^\alpha \mathbf{C}_M^{-1} \mathbf{D}_M^\alpha \mathbf{W}_M\right)^{-1} \mathbf{G}_M^H$;
$\rho \leftarrow \beta / \lambda_{\max}(\mathbf{F}_M)$;
$\mathbf{h}^{(0)} \leftarrow \hat{\mathbf{h}}(M-1)$;
$\boldsymbol{\varphi}_M^{(0)} \leftarrow \arg\left(\mathbf{W}_M \mathbf{h}^{(0)}\right)$;
$t \leftarrow 0$;
repeat
    $\boldsymbol{\varphi}_M^{(t+1)} \leftarrow \arg\left(\mathbf{e}^{j\boldsymbol{\varphi}_M^{(t)}} - \rho\left(\mathbf{F}_M \mathbf{e}^{j\boldsymbol{\varphi}_M^{(t)}} - \mathbf{G}_M \mathbf{h}^{(t)}\right)\right)$;
    $\mathbf{h}^{(t+1)} \leftarrow \tilde{\mathbf{H}}_M \mathbf{e}^{j\boldsymbol{\varphi}_M^{(t+1)}}$;
    $t \leftarrow t+1$;
until stopping criterion;
Output: $\hat{\mathbf{h}}(M) \leftarrow \mathbf{h}^{(t)}$.
```

we have

$$\bar{g}\left(\mathbf{h}^{(t)}, \mathbf{u}^{(t)}\right) = g\left(\mathbf{h}^{(t)}, \mathbf{u}^{(t)}\right), \tag{40a}$$

$$\bar{g}\left(\mathbf{h}^{(t)}, \mathbf{u}\right) \geq g\left(\mathbf{h}^{(t)}, \mathbf{u}\right), \ \forall \mathbf{u}, \tag{40b}$$

$$\nabla_{\mathbf{u}} \bar{g}\left(\mathbf{h}^{(t)}, \mathbf{u}^{(t)}\right) = \nabla_{\mathbf{u}} g\left(\mathbf{h}^{(t)}, \mathbf{u}^{(t)}\right). \tag{40c}$$

Therefore, the algorithm falls into the category of two-block alternating optimization with one nonconvex block constraint as those in [31], [45], [46]. Applying the same analysis in [46, Proposition 1], one can show that the algorithm converges to a KKT point of Problem (30).

### C. Diagonal Approximation

The GMLE-G algorithm is simple and it is guaranteed to converge. On the other hand, it needs to compute the largest eigenvalue of an $M \times M$ Hermitian matrix $\mathbf{F}_M$ at each time slot, which is cumbersome when $M$ is relatively large. In addition, since the step size $\rho$ is limited to be smaller than $1/\lambda_{\max}(\mathbf{F}_M)$ to ensure convergence, the progress of $\mathbf{u}$ can be slow when $\mathbf{F}_M$ is ill-conditioned. In this subsection, we propose another solution to update $\boldsymbol{\varphi}_M$ that well approximates the solution of Problem (33) under the setting of massive MISO. The approximation is based on the observation that $\mathbf{F}_M$ is approximately diagonal when the number of transmit antennas is large and the channels are spatially uncorrelated. Specifically, we have the following statement:

*Proposition 3:* Under the model $\mathbf{u}(m) \sim \mathcal{CN}(\mathbf{0}, \sigma_u^2 \mathbf{I}_N)$, when $N \rightarrow +\infty$, the following holds almost surely (a.s.):

1) $\mathbf{C}_M$ is a diagonal matrix;
2) The optimal solution of (33) is

$$\boldsymbol{\varphi}_M^\star = \arg\left(\mathbf{W}_M \mathbf{h}^{(t)}\right). \tag{41}$$

TABLE I
THE DIAGONAL-TO-OFF-DIAGONAL-RATIO (DODR) OF $\mathbf{C}_M$ FOR DIFFERENT NUMBER OF TRANSMIT ANTENNAS $N$; $M = 500$, $\alpha = 0.998$, $\sigma_u = 0.01$, AND $\sigma_v = 0.1$; AVERAGED OVER 100 MONTE-CARLO TRIALS

| $N$ | 2 | 4 | 8 | 16 | 32 | 64 | 128 | 256 |
|---|---|---|---|---|---|---|---|---|
| DODR | 6.46 | 10.5 | 18.5 | 34.5 | 66.7 | 131 | 259 | 515 |

The proof of Proposition 3 is relegated to the Appendix. According to Proposition 3, when the AR process driving noise $\mathbf{u}(m)$ is spatially uncorrelated and the number of transmit antennas is large (which is exactly the case in massive antenna systems), it is legitimate to employ the simple solution in (41) when updating $\boldsymbol{\varphi}_M$, i.e.,

$$\boldsymbol{\varphi}_M^{(t+1)} = \arg\left(\mathbf{W}_M \mathbf{h}^{(t)}\right). \tag{42}$$

Although the above solution is merely an approximation when $N$ is finite, we observe quite satisfactory results in the simulations—as will be demonstrated shortly in the next section.

*Remark 2:* Note that $N \rightarrow \infty$ would not be possible in practice. However, Proposotion 3 explains the good performance of (41) that treats $\mathbf{C}_M$ as a diagonal matrix (as will be seen shortly in the next section). The interesting implication of Proposition 3 is that when the number of transmit antennas is large, the update of $\boldsymbol{\varphi}_M$ is not affected by the conditioning of $\mathbf{F}_M$—which is desired. Some numerical support of this approximation is presented in Table I, where the diagonal-to-off-diagonal-ratio (DODR) of $\mathbf{C}_M$ is presented for different numbers of antennas when $M = 500$. The DODR is defined as

$$\text{DODR} = \frac{\left\|\mathbf{d}_{\mathbf{C}_M}\right\|_2^2 / M}{\left\|\mathbf{C}_M - \mathbf{D}_M\right\|_F^2 / (M^2 - M)}, \tag{43}$$

where $\mathbf{d}_{\mathbf{C}_M}$ denotes the main diagonal of $\mathbf{C}_M$ and $\mathbf{D}_M := \text{Diag}(\mathbf{d}_{\mathbf{C}_M})$. The DODR measures the ratio between the average squared magnitude of the diagonal elements and that of the off-diagonal elements in $\mathbf{C}_M$. One can see that the matrix $\mathbf{C}_M$ is already fairly diagonal dominant even when the number of antennas is small.

Finally, putting (32) and (42) together, a compact formula for updating $\mathbf{h}$ can be expressed as

$$\mathbf{h}^{(t+1)} = \hat{\mathbf{H}}_M \mathbf{e}^{j \arg\left(\mathbf{W}_M \mathbf{h}^{(t)}\right)}, \tag{44}$$

where $\hat{\mathbf{H}}_M := (\mathbf{W}_M^H \mathbf{D}_M^\alpha \mathbf{D}_M^{-1} \mathbf{D}_M^\alpha \mathbf{W}_M)^{-1} \mathbf{W}_M^H \mathbf{D}_M^\alpha \mathbf{D}_M^{-1} \mathbf{D}_M^y$; i.e., we have substituted $\mathbf{C}_M$ with $\mathbf{D}_M$.

Algorithm 3 summarizes the simplified procedures at time slot $M$. The vector $\mathbf{h}^{(0)}$ is also initialized as the channel vector $\hat{\mathbf{h}}(M-1)$ estimated at the previous time slot. The algorithm is named GMLE-Diagonal Approximation (GMLE-D).

*Remark 3:* Algorithm 3 appears fairly similar to Algorithm 1, which also has a very simple update rule of $\mathbf{h}$ (namely, matrix-vector multiplication and phase extraction only). The difference is that, in Algorithm 3, the matrix $\hat{\mathbf{H}}_M$ cannot be updated recursively as the matrix $\mathbf{H}_M$ in Algorithm 1, which means that the computational complexity of Algorithm 3 is higher. On the other hand, what we have gained is performance: by taking the channel progression model into consideration, higher channel estimation accuracy can be achieved, as will be shown shortly.

**Algorithm 3:** GMLE-D under the AR model (at time slot $M$).

```
Input: $\mathbf{W}_M$, $\alpha$, $\sigma_u^2$, $\sigma_v^2$, $\mathbf{y}_M$, and $\hat{\mathbf{h}}(M-1)$;
$\hat{\mathbf{H}}_M \leftarrow \left(\mathbf{W}_M^H \mathbf{D}_M^\alpha \mathbf{D}_M^{-1} \mathbf{D}_M^\alpha \mathbf{W}_M\right)^{-1} \mathbf{W}_M^H \mathbf{D}_M^\alpha \mathbf{D}_M^{-1} \mathbf{D}_M^y$;
$\mathbf{h}^{(0)} \leftarrow \hat{\mathbf{h}}(M-1)$;
$t \leftarrow 0$;
repeat
    $\mathbf{h}^{(t+1)} \leftarrow \hat{\mathbf{H}}_M \mathbf{e}^{j \arg\left(\mathbf{W}_M \mathbf{h}^{(t)}\right)}$;
    $t \leftarrow t+1$;
until stopping criterion;
Output: $\hat{\mathbf{h}}(M) \leftarrow \mathbf{h}^{(t)}$.
```

## VI. Simulations

In this section, we use extensive simulations to showcase the effectiveness of the proposed channel tracking algorithms. All simulations are conducted on a computer with a 3.20 GHz Intel Core i5-4570 CPU and 8.00 GB RAM. The algorithms are implemented in Matlab R2014b.

*Settings:* We generate time-varying channel vectors $\{\mathbf{h}(m) \in \mathbb{C}^N\}_{m=1}^{500}$ according to (22), where $\mathbf{h}(0)$ is generated following the unit-variance zero-mean circularly-symmetric Gaussian distribution. The beamforming vectors $\{\mathbf{w}(m)\}_{m=1}^{500}$ are also drawn from the i.i.d. unit-variance zero-mean Gaussian distribution. The feedback signal magnitude information $\{y(m)\}_{m=1}^{500}$ is measured following (8). All of the algorithms are initialized randomly at the beginning of the tracking process. The updates of the proposed algorithms start at time slot $M = N$. At every time slot, the algorithms stop when the following criterion is met:

$$\frac{\left\|\mathbf{h}^{(t)} - \mathbf{h}^{(t-1)}\right\|_2}{\left\|\mathbf{h}^{(t-1)}\right\|_2} < 10^{-4}. \tag{45}$$

*Baselines.* To benchmark the proposed algorithms, we present the performance of three variations of the Kalman filter [47]—which is a classical tool for estimating and tracking the state of linear dynamical systems. The classical Kalman filter works with linear measurements of $\mathbf{h}(m)$ rather than the magnitude measurements as under our setting. To circumvent this issue, we present the following three versions of modified Kalman filter to serve as baselines:

- *KF-P:* First, we present the performance of the Kalman filter with perfect phase information (KF-P). At each time slot $m$, we assume that the Kalman filter knows the exact phase information and thus the classical Kalman filter can be applied. Notice that assuming perfect phase knowledge is unrealistic since the phase information is lost when measuring the magnitude. Nevertheless, the performance of KF-P can serve as a lower bound of the channel estimation error.
- *KF-A:* In addition, we present a more practical modification of the Kalman filter, which is an alternating optimization approach (namely, KF-A) that alternately estimates $\{\phi(m)\}_{m=1}^M$ using a phase estimator and $\{\mathbf{h}(m)\}_{m=1}^M$ by applying the classical Kalman filter.
- *KF-S:* We also incorporate the Rauch-Tung-Striebel (RTS) smoother [48] on top of KF-A, resulting in the KF-S method. The RTS smoother performs forward-backward smoothing together with channel estimation and is known to be able to enhance the performance of the Kalman filter in the estimation mode.

Due to the page limitation, we put the detailed pseudo codes and derivations of these three baselines in the supplementary material.

*Evaluation Metric:* We employ the normalized squared error (NSE) between the estimated $\hat{\mathbf{h}}(M)$ and the ground-truth $\mathbf{h}(M)$ after accounting for the global phase ambiguity:

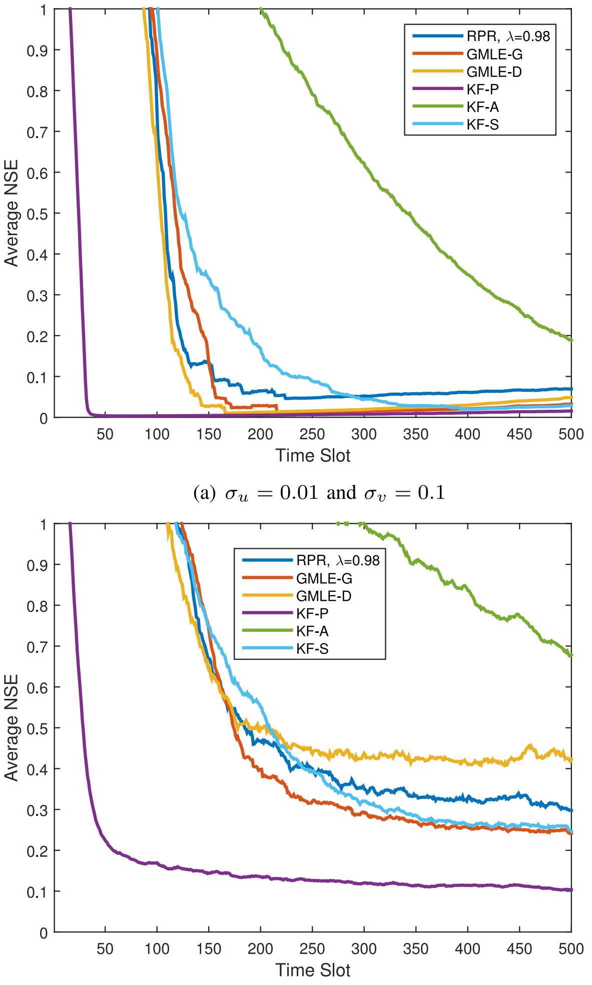
$$\begin{aligned}\text{NSE}(M) &= \min_\theta \frac{\left\|\hat{\mathbf{h}}(M)e^{j\theta} - \mathbf{h}(M)\right\|_2^2}{\|\mathbf{h}(M)\|_2^2} \\ &= \frac{\left\|\hat{\mathbf{h}}(M)e^{j\arg\left(\hat{\mathbf{h}}^H(M)\mathbf{h}(M)\right)} - \mathbf{h}(M)\right\|_2^2}{\|\mathbf{h}(M)\|_2^2}.\end{aligned} \tag{46}$$

In all the simulations, the presented NSEs are averaged over 100 Monte Carlo trials.

Fig. 2(a) shows the NSEs of different algorithms when the standard deviations of the AR process driving noise and the measurement noise are $\sigma_u = 0.01$ and $\sigma_v = 0.1$, respectively. The parameter in the AR model is set to be $\alpha = 0.998$. As expected, KF-P has the best performance as it uses the exact phase information—it reaches a very good NSE level after 50 time slots. The three proposed algorithms, namely RPR, GMLE-G, and GMLE-D, all reach satisfactory NSE levels after 150 time slots. Both GMLE-G and GMLE-D reach the same NSE eventually—which suggests that the proposed diagonal approximation in Section V-C is fairly reasonable. RPR also performs very well, but the NSEs are slightly higher relative to GMLE-G and GMLE-D. This is reasonable since RPR does not exploit the AR channel model. KF-A does not work well, since it alternates between phase estimation and channel estimation—and the channel estimation stage is not optimal. KF-S performs significantly better than KF-A due to the RTS smoother—the smoother makes the channel estimation part optimal given a fixed phase. Nevertheless, even with the smoother, KF-S still uses 200 more time slots to reach the same accuracy level of the proposed methods.

Fig. 2(b) shows the performance of the algorithms when the noise levels are lifted to $\sigma_u = 0.1$ and $\sigma_v = 1$, respectively. One can see that all algorithms perform similarly as in the previous case, except that GMLE-D gives worse NSEs compared to those of GMLE-G. This suggests that the accuracy of the diagonal approximation deteriorates when the noise level increases.

Table II presents the average CPU Time of different algorithms achieving an NSE level of 0.1, under the same settings as those in Fig. 2(a). One can see that both RPR and GMLE-D are very efficient. GMLE-G is slower because the progress of $\boldsymbol{\varphi}_M$ is limited by the step size $\rho$ (i.e., the conditioning of $\mathbf{F}_M$), as mentioned. The KF-S algorithm, although outputs reasonable NSEs, is significantly slower than the other methods—which makes it unpractical.



(a) $\sigma_u = 0.01$ and $\sigma_v = 0.1$

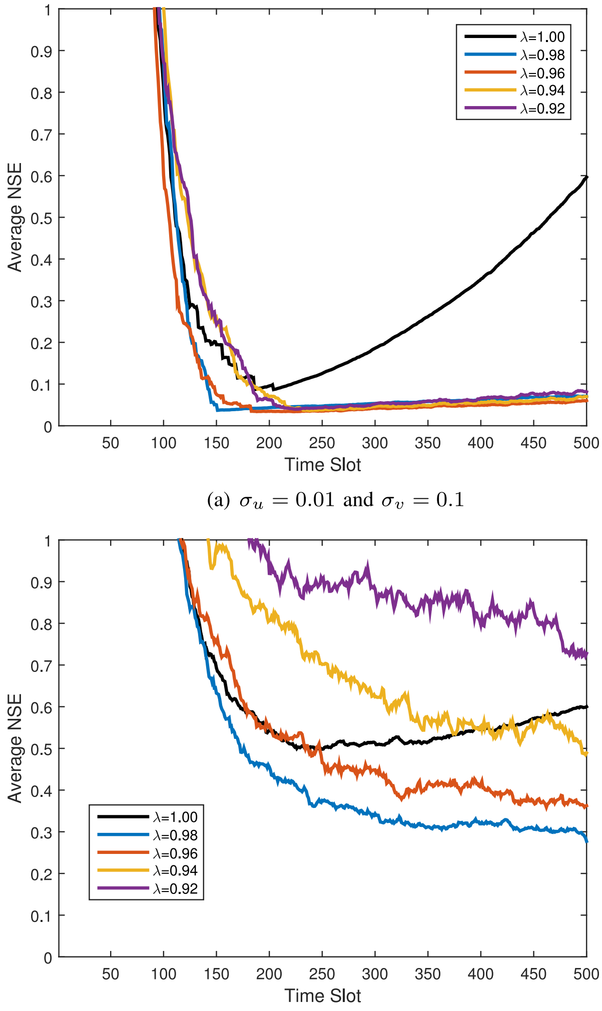
(b) $\sigma_u = 0.1$ and $\sigma_v = 1$

Fig. 2. NSE at different time slots; $N = 32$ and $\alpha = 0.998$.

TABLE II
AVERAGE CPU TIME OVER 100 MONTE CARLO TRIALS TO ACHIEVE AN NSE LEVEL OF 0.1; $N = 32$, $\alpha = 0.998$, $\sigma_u = 0.01$, AND $\sigma_v = 0.1$

| Algorithms | Average CPU Time (s) |
|---|---|
| RPR, $\lambda = 0.98$ | 0.313 |
| GMLE-G | 1.69 |
| GMLE-D | 0.324 |
| KF-P | 0.00244 |
| KF-A | 11.0 |
| KF-S | 312 |

Fig. 3 shows the performance of the RPR algorithm with different choices of $\lambda$ under the same noise settings as those in Fig. 2. We test five different values, namely $\lambda = 1.00, 0.98, 0.96,$ 0.94, and 0.92. One can see that, when the noise levels are low, using different choices of $\lambda$ (except $\lambda = 1$ that does not consider channel progression) gives similar results. When the noise levels are higher, $\lambda = 0.98$ and 0.96 give more satisfactory results since they match the real channel changing speed ($\alpha = 0.998$ in this case) better. It is also important to notice that using $\lambda = 1$ leads to fast divergence of the algorithm—the channel estimation error increases very quickly after a certain point. This suggests that when the channel is varying (even slowly), using a static channel model is rather harmful to the estimation accuracy.



(a) $\sigma_u = 0.01$ and $\sigma_v = 0.1$

(b) $\sigma_u = 0.1$ and $\sigma_v = 1$

Fig. 3. NSE vs. $\lambda$ of the Recursive Phase Retrieval (RPR) algorithm at different time slots; $N = 32$ and $\alpha = 0.998$.

Fig. 4 shows the NSEs of different algorithms at time slot $M = 200$, when $\alpha$ ranges from 0.9950 ($0.9950^{200} \approx 0.36705$) to 0.9995 ($0.9995^{200} \approx 0.9048$). RPR performs well in tracking a slowly time-varying channel (corresponding to larger $\alpha$'s), but its performance degrades when the channel changes faster. The other two proposed algorithms GMLE-G and GMLE-D give much lower NSEs relative to RPR when $\alpha \leq 0.998$, which means that the AR model based methods are more robust to rapid changes. This is not surprising—because GMLE-G and GMLE-D exploit the channel progression model but RPR works with more general slow-changing assumptions. KF-S gives similar NSEs as GMLE-G and GMLE-D across the entire range of $\alpha$ in this simulation. However, as mentioned, KF-S is significantly slower than GMLE-G, GMLE-D and RPR.

Fig. 5 shows the NSEs of different algorithms at time slot $M = 200$ as well, when $\sigma_u$ varies from 1 to 0.001. Recall that

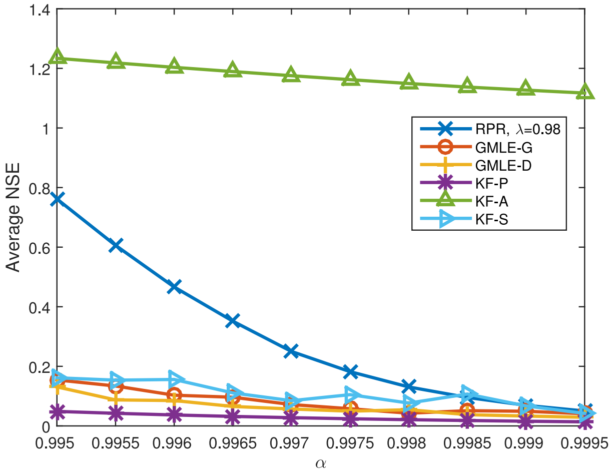


Fig. 4. NSE vs. $\alpha$ at time slot $M = 200$; $N = 32$, $\sigma_u = 0.01$, and $\sigma_v = 1$.

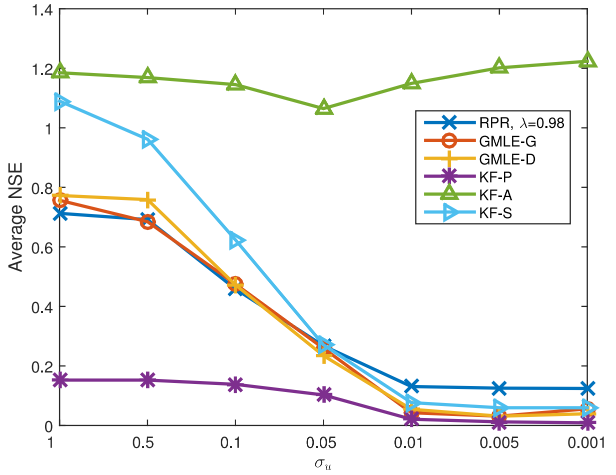


Fig. 5. NSE vs. $\sigma_u$ at time slot $M = 200$; $N = 32$, $\alpha = 0.998$, and $\sigma_v = 1$.

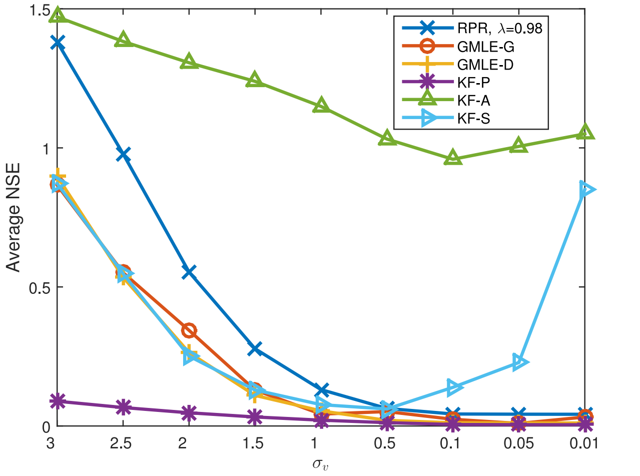


Fig. 6. NSE vs. $\sigma_v$ at time slot $M = 200$; $N = 32$, $\alpha = 0.998$, and $\sigma_u = 0.01$.

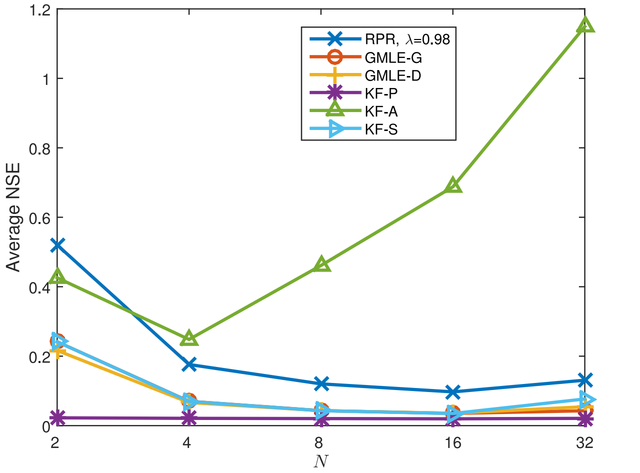


Fig. 7. NSE vs. $N$ at time slot $M = 200$; $\alpha = 0.998$, $\sigma_u = 0.01$, and $\sigma_v = 1$.

$\sigma_u$ is the standard deviation of the AR model process $\mathbf{u}(m)$; and the smaller $\sigma_u$ is, the slower the channel varies. As expected, the NSEs of all the algorithms decrease as $\sigma_u$ decreases since using larger $\sigma_u$'s poses more difficult estimation problems. Nevertheless, all of the proposed approaches give much lower NSEs relative to the benchmarking algorithms when $\sigma_u > 0.05$.

Fig. 6 shows the NSEs of different algorithms at $M = 200$ when the standard deviation of the measurement noise, i.e., $\sigma_v$, decreases from 3 to 0.01. Similar to the previous simulation, the proposed algorithms work better when the measurement noise is smaller. The GMLE-based approaches and KF-S give the most favorable NSEs when $\sigma_v$ is larger than 0.5. Interestingly, KF-S does not work well when $\sigma_v < 0.05$. There might be several reasons for this happening: first, the KF-S is not a disciplined alternating optimization approach—there is no guarantee of convergence; second, when the AR process noise is very small compared to the measurement noise, it could result in an ill-conditioned state estimation problem and thus affect the method significantly.

Fig. 7 shows the NSEs of different algorithms at time slot $M = 200$, when the base station uses different numbers of transmit antennas $N$. Surprisingly, although Proposition 3 suggests that GMLE-D approximates the GMLE well only when $N$ is large, the NSEs obtained via GMLE-D and GMLE-G are essentially the same even when $N = 2$ in this simulation. This is also consistent with the observation in Fig. 2(a), when the model process noise level is not high, GMLE-D works similarly as GMLE-G.

In Fig. 8, we test the performance of GMLE-G and GMLE-D when using an inaccurate $\alpha$—which is most likely the case in practice. In the simulation, $\alpha = 0.998$ is used to generate the original unknown channel, and three inaccurate coefficients $\tilde{\alpha} = 0.999$, 0.994, and 0.990 are used in the algorithms. One can see that the mismatch between $\alpha$ and $\tilde{\alpha}$ does affect the performance of GMLE-D and GMLE-G, but if the mismatch is not large (e.g., when $\tilde{\alpha} = 0.994$ and $\alpha = 0.998$), GMLE-D and GMLE-G still give reasonable results. Interestingly, one can see that the GMLE-D exhibits better model-robustness compared to GMLE-G.

Fig. 9 shows the NSEs of the three proposed algorithms when the receiver uses 1, 2 and 4 bits to quantize the RSS, respectively. This setting is of particular practical interest, since in practice

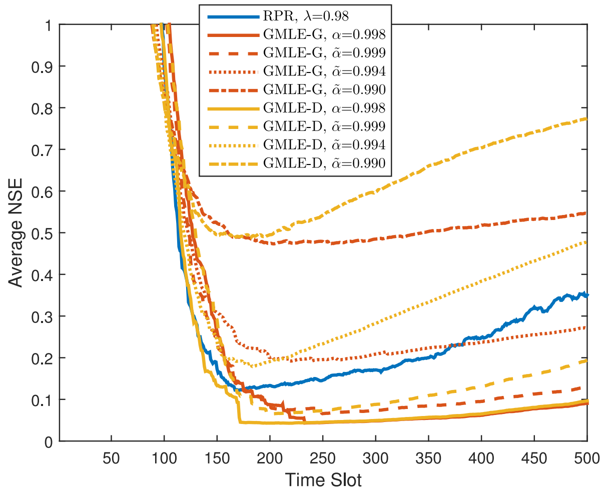


Fig. 8. NSE vs. inaccurate $\alpha$ at different time slots; $N = 32$, $\alpha = 0.998$, $\sigma_u = 0.01$, and $\sigma_v = 1$.

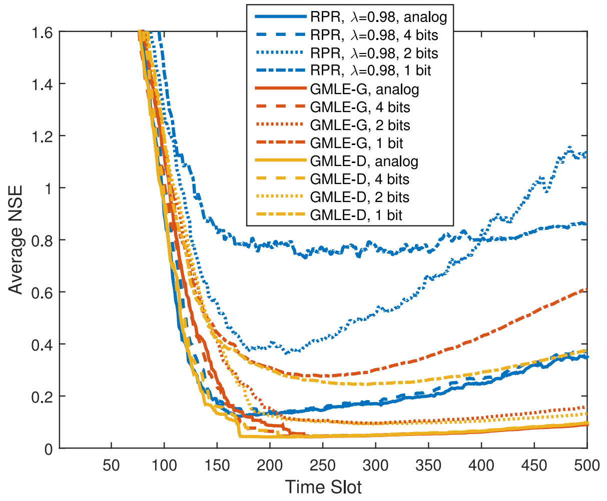


Fig. 9. NSE vs. quantization level at different time slots; $N = 32$, $\alpha = 0.998$, $\sigma_u = 0.01$, and $\sigma_v = 1$.

all the feedback will be quantized. While 1-bit quantization seems to be too aggressive, GMLE-G and GMLE-D perform surprisingly well using as few as 2-bits feedback. Compared with the result of analog feedback, the NSEs of GMLE-G and GMLE-D after 2-bits quantization only increase slightly due to the quantization error. Under the 4-bit quantization, all three proposed algorithms achieve almost identical performances to those using analog feedback. In addition, GMLE-G and GMLE-D achieve smaller NSE levels relative to RPR after such coarse quantization.

To further investigate the impact of different quantization levels, we plot in Fig. 10 the average NSEs of the three proposed algorithms at time slot $M = 200$, when different number of bits are used to quantize the signal magnitude. The other settings are the same as those in Fig. 9. One can see that, if 3 bits are employed to quantize the RSS feedback, the performance is essentially the same as using analog feedback. This is because the RSS is just a real-valued scalar and is reasonably easy to quantize accurately. This also suggests that the proposed algorithms are fairly robust to quantization errors.

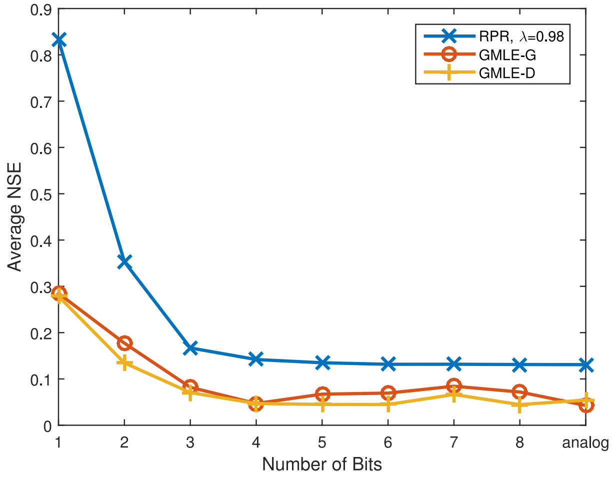


Fig. 10. NSE vs. different quantization levels at time slot $M = 200$; $N = 32$, $\alpha = 0.998$, $\sigma_u = 0.01$, and $\sigma_v = 1$.

## VII. Conclusion

In this work, the limited feedback-based downlink MISO channel estimation and tracking problem has been revisited from a new perspective that is inspired by recent developments in phase retrieval. A novel limited feedback scheme has been proposed. The proposed scheme only feeds back Received Signal Strength (RSS) / Channel Quality Indicator (CQI) information to the base station. RSS/CQI can be easily measured by any *existing* mobile device—which makes real-system implementation of the proposed scheme well within reach. The RSS/CQI measurement is merely a real-valued scalar and thus the feedback scheme is also very economical. Although RSS/CQI-type feedback does not contain any phase information for the channel, a phase retrieval-based formulation has been proposed to compensate the information loss. Under this setting, three efficient channel tracking algorithms have been proposed, which are based on a general slow-changing channel assumption and a widely adopted AR channel progression model, respectively. Extensive simulations have shown that the proposed algorithms are very effective in tracking the MISO channel.

## Appendix

### Proof of Proposition 3

We first notice two properties of the i.i.d. circularly-symmetric complex Gaussian beamforming vectors $\{\mathbf{w}(m) \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_N)\}_m$; i.e., according to the strong law of large numbers, when $N \to +\infty$,

$$\frac{\mathbf{w}^H(m)\mathbf{w}(m)}{N} \xrightarrow{\text{a.s.}} 1, \ \forall m, \tag{47a}$$

$$\frac{\mathbf{w}^H(p)\mathbf{w}(q)}{N} \xrightarrow{\text{a.s.}} 0, \ \forall p \neq q. \tag{47b}$$

Therefore, for $1 \leq p, q, m \leq M$ and $p \neq q$,

$$\frac{[\mathbf{C}_M]_{m,m}}{[\mathbf{C}_M]_{p,q}} = \frac{\frac{\alpha^{2(m-M)}-1}{1-\alpha^2}\sigma_u^2 \mathbf{w}^H(m)\mathbf{w}(m) + \sigma_v^2}{\frac{\alpha^{p+q-2M}-\alpha^{-|q-p|}}{1-\alpha^2}\sigma_u^2 \mathbf{w}^H(p)\mathbf{w}(q)}$$

$$= \frac{\beta_1 \mathbf{w}^H(m)\mathbf{w}(m)/N + \sigma_v^2/N}{\beta_2 \mathbf{w}^H(p)\mathbf{w}(q)/N}, \tag{48}$$

where $\beta_1 := \frac{\alpha^{2(m-M)}-1}{1-\alpha^2}\sigma_u^2$ and $\beta_2 := \frac{\alpha^{p+q-2M}-\alpha^{-|q-p|}}{1-\alpha^2}\sigma_u^2$. Taking $N \to +\infty$ and using equations (47a) and (47b), we have

$$\left| \frac{\beta_1 \mathbf{w}^H(m)\mathbf{w}(m)/N + \sigma_v^2/N}{\beta_2 \mathbf{w}^H(p)\mathbf{w}(q)/N} \right| \xrightarrow{\text{a.s.}} +\infty. \tag{49}$$

This implies that $\mathbf{C}_M$ is a diagonal matrix almost surely when $N$ goes to infinity. As a result, we have

$$\left(\mathbf{e}^{j\boldsymbol{\varphi}_M}\right)^H \mathbf{F}_M \mathbf{e}^{j\boldsymbol{\varphi}_M} \xrightarrow{\text{a.s.}} \mathrm{Tr}\left(\mathbf{F}_M\right) \tag{50}$$

is a constant with regard to $\boldsymbol{\varphi}_M$. Under such circumstance, Problem (33) is equivalent to

$$\underset{\boldsymbol{\varphi}_M}{\text{minimize}} \quad -2\mathrm{Re}\left\{\left(\mathbf{e}^{j\boldsymbol{\varphi}_M}\right)^H \mathbf{G}_M \mathbf{h}^{(t)}\right\}, \tag{51}$$

which has a simple closed-form solution

$$\boldsymbol{\varphi}_M^\star = \arg\left(\mathbf{G}_M \mathbf{h}^{(t)}\right) = \arg\left(\mathbf{W}_M \mathbf{h}^{(t)}\right) \tag{52}$$

as $\mathbf{D}_M^y$, $\mathbf{C}_M$, and $\mathbf{D}_M^\alpha$ are real-valued diagonal matrices.

## References

[1] J. G. Andrews *et al.*, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.

[2] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.

[3] A. Alkhateeb, O. E. Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, Oct. 2014.

[4] A. Liu, F. Zhu, and V. K. N. Lau, "Closed-loop autonomous pilot and compressive CSIT feedback resource adaptation in multi-user FDD massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 65, no. 1, pp. 173–183, Jan. 2017.

[5] S. Noh, M. D. Zoltowski, and D. J. Love, "Multi-resolution codebook and adaptive beamforming sequence design for millimeter wave beam alignment," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 5689–5701, Sep. 2017.

[6] J. Choi, D. J. Love, and P. Bidigare, "Downlink training techniques for FDD massive MIMO systems: Open-loop and closed-loop training with memory," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 802–814, Oct. 2014.

[7] D. J. Love, R. W. Heath, W. Santipach, and M. L. Honig, "What is the value of limited feedback for MIMO channels?" *IEEE Commun. Mag.*, vol. 42, no. 10, pp. 54–59, Oct. 2004.

[8] D. J. Love, R. W. Heath, V. K. N. Lau, D. Gesbert, B. D. Rao, and M. Andrews, "An overview of limited feedback in wireless communication systems," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 8, pp. 1341–1365, Oct. 2008.

[9] P. Xia and G. B. Giannakis, "Design and analysis of transmit-beamforming based on limited-rate feedback," *IEEE Trans. Signal Process.*, vol. 54, no. 5, pp. 1853–1863, May 2006.

[10] O. Mehanna and N. D. Sidiropoulos, "Channel tracking and transmit beamforming with frugal feedback," *IEEE Trans. Signal Process.*, vol. 62, no. 24, pp. 6402–6413, Dec. 2014.

[11] B. Gopalakrishnan and N. D. Sidiropoulos, "Cognitive transmit beamforming from binary CSIT," *IEEE Trans. Wireless Commun.*, vol. 14, no. 2, pp. 895–906, Feb. 2015.

[12] P. N. Alevizos, X. Fu, N. Sidiropoulos, Y. Yang, and A. Bletsas, "Nonuniform directional dictionary-based limited feedback for massive mimo systems," in *Proc. 15th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wireless Netw.*, 2017, pp. 1–8.

[13] D. C. Araújo, T. Maksymyuk, A. L. F. de Almeida, T. Maciel, J. C. M. Mota, and M. Jo, "Massive MIMO: Survey and future research topics," *IET Commun.*, vol. 10, no. 15, pp. 1938–1946, 2016.

[14] J. R. Fienup, "Phase retrieval algorithms: A comparison," *Appl. Opt.*, vol. 21, no. 15, pp. 2758–2769, 1982.

[15] Y. Shechtman, Y. Eldar, O. Cohen, H. Chapman, J. Miao, and M. Segev, "Phase retrieval with application to optical imaging: A contemporary overview," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 87–109, May 2015.

[16] A. H. Sayed, *Fundamentals of Adaptive Filtering*. Hoboken, NJ, USA: Wiley, 2003.

[17] A. H. Sayed, *Adaptive Filters*. Hoboken, NJ, USA: Wiley, 2008.

[18] R. W. Gerchberg and W. O. Saxton, "A practical algorithm for the determination of phase from image and diffraction plane pictures," *Optik*, vol. 35, pp. 237–250, 1972.

[19] P. Netrapalli, P. Jain, and S. Sanghavi, "Phase retrieval using alternating minimization," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4814–4826, Sep. 2015.

[20] A. Walther, "The question of phase retrieval in optics," *Optica Acta, Int. J. Opt.*, vol. 10, no. 1, pp. 41–49, 1963.

[21] J. R. Fienup, "Reconstruction of an object from the modulus of its Fourier transform," *Opt. Lett.*, vol. 3, no. 1, pp. 27–29, Jul. 1978.

[22] J. R. Fienup, "Iterative method applied to image reconstruction and to computer-generated holograms," *Opt. Eng.*, vol. 19, no. 3, pp. 297–305, 1980.

[23] E. J. Candès, T. Strohmer, and V. Voroninski, "Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming," *Commun. Pure Appl. Math.*, vol. 66, no. 8, pp. 1241–1274, 2013.

[24] I. Waldspurger, A. d'Asspremont, and S. Mallat, "Phase recovery, Maxcut and complex semidefinite programming," *Math. Program.*, vol. 149, no. 1/2, pp. 47–81, 2015.

[25] E. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval via Wirtinger flow: Theory and algorithms," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1985–2007, Apr. 2015.

[26] G. Wang, G. B. Giannakis, and Y. C. Eldar, "Solving systems of random quadratic equations via truncated amplitude flow," *IEEE Trans. Inf. Theory*, vol. 64, no. 2, pp. 773–794, Feb. 2018, doi: 10.1109/TIT.2017.2756858.

[27] T. Qiu, P. Babu, and D. P. Palomar, "PRIME: Phase retrieval via majorization-minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 19, pp. 5174–5186, Oct. 2016.

[28] B. G. Bodmann and N. Hammen, "Stable phase retrieval with low-redundancy frames," *Adv. Comput. Math.*, vol. 41, no. 2, pp. 317–331, 2015.

[29] A. Conca, D. Edidin, M. Hering, and C. Vinzant, "An algebraic characterization of injectivity in phase retrieval," *Appl. Comput. Harmonic Anal.*, vol. 38, no. 2, pp. 346–356, 2015.

[30] N. Vaswani, S. Nayer, and Y. C. Eldar, "Low-rank phase retrieval," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 4059–4074, Aug. 2017.

[31] C. Qian, X. Fu, N. D. Sidiropoulos, L. Huang, and J. Xie, "Inexact alternating optimization for phase retrieval in the presence of outliers," *IEEE Trans. Signal Process.*, vol. 65, no. 22, pp. 6069–6082, Nov. 2017.

[32] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 1999.

[33] M. Hong, M. Razaviyayn, Z. Q. Luo, and J. S. Pang, "A unified algorithmic framework for block-structured optimization involving big data: With applications in machine learning and signal processing," *IEEE Signal Process. Mag.*, vol. 33, no. 1, pp. 57–77, Jan. 2016.

[34] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 1126–1153, 2013.

[35] B. Chen, S. He, Z. Li, and S. Zhang, "Maximum block improvement and polynomial optimization," *SIAM J. Optim.*, vol. 22, no. 1, pp. 87–107, 2012.

[36] K. Huang, N. D. Sidiropoulos, and A. Swami, "Non-negative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition," *IEEE Trans. Signal Process.*, vol. 62, no. 1, pp. 211–224, Jan. 2014.

[37] X. Fu, W. K. Ma, K. Huang, and N. D. Sidiropoulos, "Blind separation of quasi-stationary sources: Exploiting convex geometry in covariance domain," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2306–2320, May 2015.

[38] Z. Liu, X. Ma, and G. B. Giannakis, "Space-time coding and Kalman filtering for time-selective fading channels," *IEEE Trans. Commun.*, vol. 50, no. 2, pp. 183–186, Feb. 2002.

[39] M. Sadek, A. Tarighat, and A. H. Sayed, "Exploiting spatio-temporal correlation for rate-efficient transmit beamforming," in *Proc. Conf. Rec. 38th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2004, vol. 2, pp. 2027–2031.

[40] A. Ribeiro, G. B. Giannakis, and S. I. Roumeliotis, "SOI-KF: Distributed Kalman filtering with low-cost communications using the sign of innovations," *IEEE Trans. Signal Process.*, vol. 54, no. 12, pp. 4782–4795, Dec. 2006.

[41] C. Komninakis, C. Fragouli, A. H. Sayed, and R. D. Wesel, "Multi-input multi-output fading channel tracking and equalization using Kalman estimation," *IEEE Trans. Signal Process.*, vol. 50, no. 5, pp. 1065–1076, May 2002.

[42] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *Amer. Statist.*, vol. 58, no. 1, pp. 30–37, 2004.

[43] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 3, pp. 794–816, Feb. 2017.

[44] J. Song, P. Babu, and D. P. Palomar, "Optimization methods for designing sequences with low autocorrelation sidelobes," *IEEE Trans. Signal Process.*, vol. 63, no. 15, pp. 3998–4009, Aug. 2015.

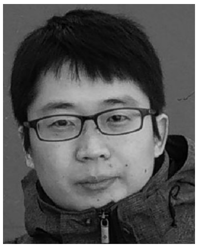
[45] J. Tranter, N. D. Sidiropoulos, X. Fu, and A. Swami, "Fast unit-modulus least squares with applications in beamforming," *IEEE Trans. Signal Process.*, vol. 65, no. 11, pp. 2875–2887, Jun. 2017.

[46] X. Fu, K. Huang, M. Hong, N. D. Sidiropoulos, and A. M. C. So, "Scalable and flexible multiview MAX-VAR canonical correlation analysis," *IEEE Trans. Signal Process.*, vol. 65, no. 16, pp. 4150–4165, Aug. 2017.

[47] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, 1960.

[48] H. E. Rauch, F. Tung, and C. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA J.*, vol. 3, no. 8, pp. 1445–1450, 1965.

**Tianyu Qiu** received the B.Eng. degree in electronic and information from the Huazhong University of Science and Technology, Wuhan, China, in 2013, and is currently working toward the Ph.D. degree in electronic and computer engineering at the Hong Kong University of Science and Technology, Hong Kong. His research interests include statistical signal processing, mathematical optimization, and machine learning.



**Xiao Fu** (S'12–M'15) received the Ph.D. degree in electronic engineering from the Chinese University of Hong Kong, Hong Kong, in 2014. From 2014 to 2017, he was a Postdoctoral Associate with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA. In 2017, he joined the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, USA, as an Assistant Professor. His research interests include signal processing and machine learning, with a recent emphasis on factor analysis and its applications. He was the recipient of the Best Student Paper Award at ICASSP 2014, and (coauthor) the Best Student Paper Award at IEEE CAMSAP 2015.

**Nicholas D. Sidiropoulos** (F'09) received the Diploma in electrical engineering from the Aristotelian University of Thessaloniki, Thessaloniki, Greece, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland at College Park, College Park, MD, USA, in 1988, 1990, and 1992, respectively. He was an Assistant Professor with the University of Virginia, Charlottesville, VA, USA, an Associate Professor with the University of Minnesota, Minneapolis, MN, USA, and a Professor with TU Crete, Chania, Greece. From 2011 to 2017, he was with the University of Minnesota, where he held the ADC Chair in digital technology. He is now with the Department of Electrical and Computer Engineering, University of Virgina, where he is currently the Department Chair. His research interests include signal processing theory and algorithms, optimization, communications, and factor analysis—with a long-term interest in tensor decomposition and its applications. His current focus is primarily on signal and tensor analytics for learning from big data. He was the recipient of the NSF/CAREER Award in 1998, and the IEEE Signal Processing (SP) Society Best Paper Award in 2001, 2007, and 2011. He was the IEEE SP Society Distinguished Lecturer (2008–2009), and was the Chair for the IEEE Signal Processing for Communications and Networking Technical Committee (2007–2008). He was the recipient of the 2010 IEEE SP Society Meritorious Service Award, and the 2013 Distinguished Alumni Award from the Department of Electrical and Computer Engineering, University of Maryland. He is a Fellow of EURASIP (2014). He is currently a VP Member of the IEEE SP Society.

**Daniel P. Palomar** (S'99–M'03–SM'08–F'12) received the Electrical Engineering and Ph.D. degrees from the Technical University of Catalonia (UPC), Barcelona, Spain, in 1998 and 2003, respectively. He is a Professor in the Department of Electronic and Computer Engineering with the Hong Kong University of Science and Technology (HKUST), Hong Kong, which he joined in 2006. Since 2013, he has been a Fellow of the Institute for Advance Study, HKUST. He had previously held several research appointments, namely with King's College London, London, U.K.; Stanford University, Stanford, CA, USA; Telecommunications Technological Center of Catalonia, Barcelona, Spain; Royal Institute of Technology, Stockholm, Sweden; University of Rome "La Sapienza," Rome, Italy; and Princeton University, Princeton, NJ, USA. His current research interests include applications of convex optimization theory and signal processing to financial systems, big data systems, and communication systems. He was the recipient of a 2004/2006 Fulbright Research Fellowship, the 2004 and 2015 (coauthor) Young Author Best Paper Awards by the IEEE Signal Processing Society, the 2015–2016 HKUST Excellence Research Award, the 2002/2003 best Ph.D. prize in Information Technologies and Communications by the UPC, the 2002/2003 Rosina Ribalta first prize for the Best Doctoral Thesis in information technologies and communications by the Epson Foundation, and the 2004 prize for the Best Doctoral Thesis in Advanced Mobile Communications by the Vodafone Foundation and COIT.