Fronthaul Data Reduction in Massive MIMO Aided C-RAN via Two-timescale Hybrid Compression

An Liu[1], *Senior Member, IEEE*, Xihan Chen[1], Wei Yu[2], *Fellow, IEEE*, Vincent Lau[3], *Fellow, IEEE* and Min-Jian Zhao[1]
[1]College of Information Science and Electronic Engineering, Zhejiang University
[2]The Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto
[3]Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology

***Abstract*—In massive MIMO aided cloud radio access network (C-RAN), plenty of remote radio heads (RRHs), each equipped with a massive MIMO array, are distributed within a specific geographical area and are connected to a centralized baseband unit (BBU) pool through fronthaul links. One major performance bottleneck in the uplink of massive MIMO aided C-RAN is that, the RRHs need to transport a huge amount of data to the BBU for baseband processings. Existing fronthaul compression methods that rely on fully-digital processing are not suitable for the massive MIMO regime due to their high implementation cost. To overcome this challenge, we propose a two-timescale hybrid analog-and-digital spatial compression scheme at RRHs to reduce the fronthaul data, where the analog filter is updated at a slow timescale according to the channel statistics to achieve massive MIMO array gain, and the digital filter is updated at a fast timescale according to the instantaneous effective channel state information (CSI) to achieve spatial multiplexing gain. Such a design can alleviate the performance bottleneck of limited fronthaul with reduced hardware cost and power consumption, and is more robust to the CSI delay. We propose an online algorithm for the two-timescale non-convex optimization of analog and digital filters. Simulations verify the advantages of the proposed scheme over state-of-the-art baseline schemes.**

***Index Terms*—Cloud radio access network, Massive MIMO, Hybrid compression and forward**

I. Introduction

Recently, massive MIMO aided C-RAN has been proposed to improve the spectral efficiency of wireless systems [1]. However, such an architecture requires a huge amount of digital sampled data to be transported over the fronthaul link. Therefore, it is necessary to compress the uplink data at each RRH to satisfy the limited fronthaul capacity constraint. Various fully-digital fronthaul compression techniques have been proposed for C-RAN with small-scale multi-antenna RRHs [2], [3]. In particular, the spatial compression and forward scheme proposed in [3] combines fully-digital spatial filtering and uniform scalar quantization to alleviate the performance bottleneck caused by the limited fronthaul capacity. Unfortunately, fully-digital spatial filtering requires a larger number of analog-to-digital converter (ADCs) and radio frequency (RF) chains at each massive MIMO RRH. In [4], a fully-analog linear spatial filtering is used at each RRH to achieve the fronthaul compression with reduced hardware cost and power consumption. However, fully-analog processing is known to be less efficient than hybrid analog and digital processing.

In this paper, we propose a two-timescale hybrid (analog and digital) compression and forward (THCF) scheme for the uplink transmission of massive MIMO aided C-RAN, to alleviate the performance bottleneck of the limited fronthaul, with reduced hardware cost and power consumption. In this scheme, each RRH first performs a two-timescale hybrid analog and digital spatial filtering to reduce the dimension of its received signal. Specifically, the analog filtering matrix is adapted to the long-term channel statistics to achieve massive MIMO array gain, and the digital filtering matrix is adapted to the instantaneous effective CSI to achieve spatial multiplexing gain. Then, each RRH applies the uniform scalar quantization over each of these dimensions. Finally, the quantized signals at the RRHs are sent to the BBU for joint decoding.

The power allocation at users, analog/digital filtering matrices and quantization bits allocation at RRHs, and the receive beamforming matrix at the BBU are jointly optimized to maximize a general utility function. We propose an online *block-coordinate stochastic successive convex approximation* (BC-SSCA) algorithm to solve this joint optimization problem. Simulations show that the proposed two-timescale hybrid scheme achieves better tradeoff performance than the baselines.

II. System Model

A. Network Architecture and Channel Model

Consider the uplink of a massive MIMO aided C-RAN, where N RRHs, each equipped with a massive MIMO array of $M \gg 1$ antennas and $S < M$ Rx RF chains, are distributed within a specific geographical area to serve K single-antenna users. Each RRH n serves as a relay between the BBU and users, and is connected to the BBU via a fronthaul link of capacity C_n bits per second (bps). We assume that the number of users K is fixed and $NS \gg K$ so that there are enough spatial degrees of freedom to serve all the K users. In this case, the received signal at RRH n is given by

$$\boldsymbol{y}_n = \sum_{k=1}^{K} \boldsymbol{h}_{n,k}\sqrt{p_k}s_k + \boldsymbol{z}_n = \boldsymbol{H}_n\boldsymbol{P}^{1/2}\boldsymbol{s} + \boldsymbol{z}_n,$$

where $\boldsymbol{h}_{n,k} \in \mathbb{C}^M$ is the channel vector from user k to RRH n, $\boldsymbol{H}_n = [\boldsymbol{h}_{n,1}, ..., \boldsymbol{h}_{n,K}] \in \mathbb{C}^{M\times K}$, $s_k \sim \mathcal{CN}(0,1)$ is the data symbol of user k, $\boldsymbol{s} = [s_1, ..., s_K]^T$, p_k is the transmit power of user k, $\boldsymbol{P} = \mathrm{diag}(p_1, ..., p_K)$, and $\boldsymbol{z}_n \sim \mathcal{CN}(0, \boldsymbol{I})$ is the noise vector.

B. Two-timescale Hybrid Compression and Forward at RRHs

Each RRH n applies the THCF scheme to make sure that the compressed received signal $\widetilde{\boldsymbol{y}}_n$ satisfies the fronthaul capacity

978-1-5386-3599-5/18/$31.00©2018 IEEE

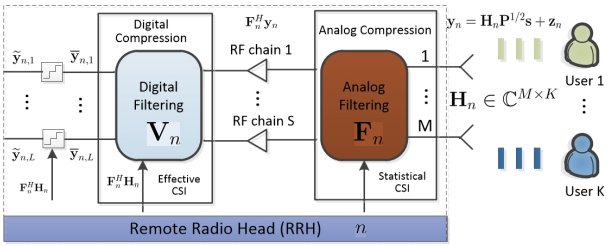

Figure 1: An illustration of the THCF scheme

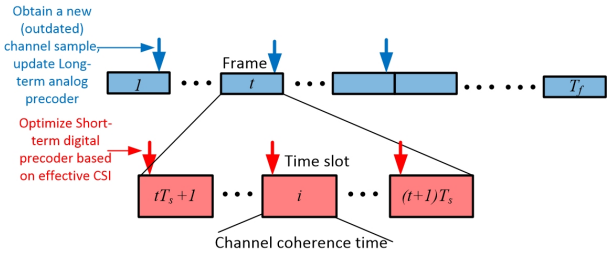

Figure 2: An illustration of two-timescale frame structure.

constraint of C_n bps, as illustrated in Fig. 1. Specifically, a hybrid filtering matrix $\boldsymbol{F}_n\boldsymbol{V}_n \in \mathbb{C}^{M\times L}$ is applied at RRH n to compress the received signal $\boldsymbol{y}_n$ into a low-dimensional signal $\overline{\boldsymbol{y}}_n = \boldsymbol{V}_n^H\boldsymbol{F}_n^H\boldsymbol{y}_n = \left[\overline{y}_{n,1},...,\overline{y}_{n,L}\right]^T \in \mathbb{C}^L$, where $\boldsymbol{F}_n \in \mathbb{C}^{M\times S}$ and $\boldsymbol{V}_n = [\boldsymbol{v}_{n,1},...,\boldsymbol{v}_{n,L}] \in \mathbb{C}^{S\times L}$ are the analog and digital filtering matrices, respectively, and we set $L = \min(K,S)$ such that there is no information loss due to digital filtering [3]. The analog filtering matrix $\boldsymbol{F}_n$ is usually implemented using an RF phase shifting network [5]. Hence, $\boldsymbol{F}_n$ can be represented by a phase vector $\boldsymbol{\theta}_n \in [0,2\pi]^{MS}$, whose $((j-1)M+i)$-th element $\theta_{n,i,j}$ is the phase of the (i,j)-th element of $\boldsymbol{F}_n$. Then, a simple uniform scalar quantization [3] is applied to each element of $\overline{\boldsymbol{y}}_n$ at RRH n.

After the uniform scalar quantization, the compressed received signal $\widetilde{\boldsymbol{y}}_n = [\widetilde{y}_{n,1},...,\widetilde{y}_{n,L}]^T$ is modeled by

$$\widetilde{\boldsymbol{y}}_n = \overline{\boldsymbol{y}}_n + \boldsymbol{e}_n = \boldsymbol{V}_n^H\boldsymbol{F}_n^H\left(\boldsymbol{H}_n\boldsymbol{P}^{1/2}\boldsymbol{s}+\boldsymbol{z}_n\right)+\boldsymbol{e}_n,$$

where $\boldsymbol{e}_n = [e_{n,1},...,e_{n,L}] \in \mathbb{C}^L$ with $e_{n,l}$ denoting the quantization error for $\overline{y}_{n,l}$. Let $d_{n,l}$ denote the number of bits that RRH n uses to quantize the real or imaginary part of $\overline{y}_{n,l}$. With uniform scalar quantization, the covariance matrix of $\boldsymbol{e}_n$ is given by a function of $\boldsymbol{p} = [p_1,...,p_K]^T$, $\boldsymbol{F}_n\boldsymbol{V}_n$ and $\boldsymbol{d}_n = [d_{n,1},...,d_{n,L}]^T$ as [3]

$$\boldsymbol{Q}_n\left(\boldsymbol{p},\boldsymbol{F}_n\boldsymbol{V}_n,\boldsymbol{d}_n\right) = \mathrm{diag}\left(q_{n,1},...,q_{n,L}\right),$$

$$q_{n,l} = \begin{cases} \frac{3}{4^{d_{n,l}}}\left(\sum_{k=1}^K p_k|\boldsymbol{h}_{n,k}^H\widetilde{\boldsymbol{v}}_{n,l}|^2 + \|\widetilde{\boldsymbol{v}}_{n,l}\|^2\right) & \text{if } d_{n,l}>0, \\ \infty & \text{if } d_{n,l}=0, \end{cases} \tag{1}$$

where $\widetilde{\boldsymbol{v}}_{n,l} = \boldsymbol{F}_n\boldsymbol{v}_{n,l}$. Finally, each RRH forwards the quantized bits to the BBU via the fronthaul link.

C. Joint Rx Beamforming at the BBU

The received signal $\widetilde{\boldsymbol{y}} = \left[\widetilde{\boldsymbol{y}}_1^T,...,\widetilde{\boldsymbol{y}}_N^T\right]^T$ at the BBU from all RRHs can be expressed as

$$\widetilde{\boldsymbol{y}} = \widetilde{\boldsymbol{V}}^H\boldsymbol{H}\boldsymbol{P}^{1/2}\boldsymbol{s}+\widetilde{\boldsymbol{V}}^H\boldsymbol{z}+\boldsymbol{e},$$

where $\widetilde{\boldsymbol{V}} = \mathrm{diag}\left(\boldsymbol{F}_1\boldsymbol{V}_1,...,\boldsymbol{F}_N\boldsymbol{V}_N\right) \in \mathbb{C}^{MN\times LN}$, $\boldsymbol{H} = [\boldsymbol{h}_1,...,\boldsymbol{h}_K] \in \mathbb{C}^{MN\times K}$ with $\boldsymbol{h}_k = \left[\boldsymbol{h}_{1,k}^T,...,\boldsymbol{h}_{N,k}^T\right]^T$ denoting the composite channel vector of user k, $\boldsymbol{z} = \left[\boldsymbol{z}_1^T,...,\boldsymbol{z}_N^T\right]^T$, and $\boldsymbol{e} = \left[\boldsymbol{e}_1^T,...,\boldsymbol{e}_N^T\right]^T$. A joint Rx beamforming vector $\boldsymbol{u}_k \in \mathbb{C}^{NL\times 1}$ is applied at the BBU to obtain the estimated data symbol for each user k as

$$\hat{s}_k = \boldsymbol{u}_k^H\widetilde{\boldsymbol{V}}^H\boldsymbol{H}\boldsymbol{P}^{1/2}\boldsymbol{s}+\boldsymbol{u}_k^H\widetilde{\boldsymbol{V}}^H\boldsymbol{z}+\boldsymbol{u}_k^H\boldsymbol{e},\forall k.$$

D. Frame Structure and Achievable Data Rate

We focus on a coherence time interval of channel statistics, which is divided into T_f frames with each frame consisting of T_s time slots (channel coherence intervals), as illustrated in Fig. 2. We assume that the BBU can obtain the real-time effective CSI $\boldsymbol{F}_n^H\boldsymbol{H}_n \in \mathbb{C}^{S\times K},\forall n$ at each time slot, and one (possibly outdated) channel sample $\boldsymbol{H}$ at each frame. The long-term analog filtering matrices $\mathbf{F}_n,\forall n$ are only updated once per frame based on a channel sample to achieve massive MIMO array gain. The short-term control variables $\{\boldsymbol{p},\boldsymbol{V}_n,\boldsymbol{d}_n,\boldsymbol{u}_k\}$ are adaptive to the real-time effective CSI $\boldsymbol{F}_n^H\boldsymbol{H}_n,\forall n$ to achieve the spatial multiplexing gain. For convenience, we let $\boldsymbol{v} = \mathrm{Vec}\left([\boldsymbol{V}_1,...,\boldsymbol{V}_N]\right)$, $\boldsymbol{d} = \left[\boldsymbol{d}_1^T,...,\boldsymbol{d}_N^T\right]^T$ and $\boldsymbol{u} = \left[\boldsymbol{u}_1^T,...,\boldsymbol{u}_K^T\right]^T$.

For given long-term control variables $\boldsymbol{\theta} = \left[\boldsymbol{\theta}_1^T,...,\boldsymbol{\theta}_n^T\right]^T$ (phase vectors of analog filtering matrices), short-term control variables $\boldsymbol{x} \triangleq \left[\boldsymbol{p}^T,\boldsymbol{v}^T,\boldsymbol{d}^T,\boldsymbol{u}^T\right]^T$ and channel realization $\boldsymbol{H}$, the achievable data rate of user k is given by

$$r_k^\circ\left(\boldsymbol{\theta},\boldsymbol{x},\boldsymbol{H}\right) = \log\left(1+\mathrm{SINR}_k\left(\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{H}\right)\right),$$

where $\mathrm{SINR}_k\left(\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{H}\right)$ is the SINR of user k given by

$$\mathrm{SINR}_k\left(\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{H}\right) = \frac{p_k|\boldsymbol{u}_k^H\widetilde{\boldsymbol{V}}^H\boldsymbol{h}_k|^2}{\sum\limits_{l\neq k}p_l|\boldsymbol{u}_k^H\widetilde{\boldsymbol{V}}^H\boldsymbol{h}_l|^2+||\boldsymbol{u}_k^H\widetilde{\boldsymbol{V}}^H||^2+\boldsymbol{u}_k^H\boldsymbol{Q}\left(\boldsymbol{\theta},\boldsymbol{p},\boldsymbol{v},\boldsymbol{d}\right)\boldsymbol{u}_k},$$

$$\boldsymbol{Q}\left(\boldsymbol{\theta},\boldsymbol{p},\boldsymbol{v},\boldsymbol{d}\right) = \mathrm{diag}\left(\boldsymbol{Q}_1\left(\boldsymbol{p},\boldsymbol{F}_1\boldsymbol{V}_1,\boldsymbol{d}_1\right),...,\boldsymbol{Q}_N\left(\boldsymbol{p},\boldsymbol{F}_N\boldsymbol{V}_N,\boldsymbol{d}_N\right)\right).$$

Note that $\boldsymbol{F}_n$ is a function of $\boldsymbol{\theta}_n$.

Let $\boldsymbol{x}(\boldsymbol{H})$ denote the short-term control variable under channel state $\boldsymbol{H}$ and $\overline{\Omega} \triangleq \left\{\boldsymbol{x}(\boldsymbol{H}) \in \overline{\mathcal{X}},\forall\boldsymbol{H}\right\}$ denote the collection of the short-term control variables for all possible channel states, with $\overline{\mathcal{X}}$ denoting the feasible set of the short-term control variables. To be more specific, $\overline{\mathcal{X}}$ is the set of all short-term control variables $\boldsymbol{x} = \left[\boldsymbol{p}^T,\boldsymbol{v}^T,\boldsymbol{d}^T,\boldsymbol{u}^T\right]^T$ that satisfy the following constraints:

$$p_k \in [0,P_k],\forall k, \tag{2}$$

$$2B_W\sum_{l=1}^L d_{n,l} \leq C_n,\forall n, \tag{3}$$

$$d_{n,l} \geq 0 \text{ is an integer},\forall n,l, \tag{4}$$

where P_k is the individual power constraint at user k, B_W is the system bandwidth, and (3) is the fronthaul capacity constraint. Then the average data rate of user k is

$$\overline{r}_k^{\circ}\left(\boldsymbol{\theta},\overline{\Omega}\right)=\mathbb{E}\left[r_k^{\circ}\left(\boldsymbol{\theta},\boldsymbol{x}(\boldsymbol{H});\boldsymbol{H}\right)\right],$$

For convenience, define $\overline{\boldsymbol{r}}^{\circ}\left(\boldsymbol{\theta},\overline{\Omega}\right) \triangleq [\overline{r}_1^{\circ}\left(\boldsymbol{\theta},\overline{\Omega}\right),...,\overline{r}_K^{\circ}\left(\boldsymbol{\theta},\overline{\Omega}\right)]^T$ as the average data rate vector.

III. Two-timescale Joint Optimization at BBU

Note that $r_k^{\circ}\left(\boldsymbol{\theta},\boldsymbol{x},\boldsymbol{H}\right)$ is not a smooth function of $d_{n,l},\forall n,l$ because $d_{n,l}$ is integer. To make the problem tractable, we relax the integer constraint on $d_{n,l}$ and approximate the quantization noise power $q_{n,l},\forall n,l$ with the following smooth function of a real variable $d_{n,l}\geq 0$ as [3]

$$\hat{q}_{n,l}=\frac{3}{4^{d_{n,l}}}\left(\sum_{k=1}^{K}p_k|\boldsymbol{h}_{n,k}^H\widetilde{\boldsymbol{v}}_{n,l}|^2+\|\widetilde{\boldsymbol{v}}_{n,l}\|^2\right). \quad (5)$$

We use $r_k\left(\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{H}\right)$ to denote the approximate data rate of user k obtained by replacing $q_{n,l}$ in (1) with $\hat{q}_{n,l}$ in (5) and the integer constraint in (4) with constraint $d_{n,l}\geq 0$. Moreover, define $\overline{\boldsymbol{r}}\left(\boldsymbol{\theta},\Omega\right)=\left[\overline{r}_1\left(\boldsymbol{\theta},\Omega\right),...,\overline{r}_K\left(\boldsymbol{\theta},\Omega\right)\right]^T$ as the approximate average data rate vector, where $\overline{r}_k\left(\boldsymbol{\theta},\Omega\right)=\mathbb{E}\left[r_k\left(\boldsymbol{\theta},\boldsymbol{x}(\boldsymbol{H});\boldsymbol{H}\right)\right]$. Then, the two-timescale joint optimization of long-term and short-term control variables can be formulated as the following utility maximization problem

$$\mathcal{P}:\max_{\boldsymbol{\theta}\in\Theta,\Omega}\ g\left(\overline{\boldsymbol{r}}\left(\boldsymbol{\theta},\Omega\right)\right), \quad (6)$$

where $\Omega\triangleq\{\boldsymbol{x}(\boldsymbol{H})\in\mathcal{X},\forall\boldsymbol{H}\}$ with $\mathcal{X}$ denoting the set of all short-term control variables that satisfy constraint (2), (3) and $d_{n,l}\geq 0$, the utility function $g\left(\overline{\boldsymbol{r}}\right)$ is continuously differentiable function of $\overline{\boldsymbol{r}}$, $\Theta\triangleq[0,2\pi]^{NMS}$ is the feasible set of $\boldsymbol{\theta}$. Moreover, $g\left(\overline{\boldsymbol{r}}\right)$ is non-decreasing w.r.t. $\overline{r}_k,\forall k$ and its derivative $\nabla_{\overline{\boldsymbol{r}}}g\left(\overline{\boldsymbol{r}}\right)$ w.r.t. $\overline{\boldsymbol{r}}$ is Lipschitz continuous. This general utility function $g\left(\overline{\boldsymbol{r}}\right)$ includes many important network utilities as special cases, such as average sum rate ($g\left(\overline{\boldsymbol{r}}\right)=\sum_{k=1}^{K}\overline{r}_k$) and proportional fairness (PFS) utility ($g\left(\overline{\boldsymbol{r}}\right)=\sum_{k=1}^{K}\log\left(\overline{r}_k+\varepsilon\right)$, where $\varepsilon>0$ is a small number to avoid the singularity at $\overline{r}_k=0$).

Since Problem $\mathcal{P}$ is a two-timescale stochastic non-convex problem, we focus on designing an efficient algorithm to find stationary solutions of Problem $\mathcal{P}$, as defined below.

Definition 1 (Stationary solution of $\mathcal{P}$)**.** A solution $(\boldsymbol{\theta}^*,\Omega^*=\{\boldsymbol{x}^*\left(\boldsymbol{H}\right)\in\mathcal{X},\forall\boldsymbol{H}\})$ is called a stationary solution of Problem $\mathcal{P}$ if it satisfies the following conditions:

1) For every $\boldsymbol{H}$ outside a set of probability zero,

$$\left(\boldsymbol{x}-\boldsymbol{x}^*(\boldsymbol{H})\right)^T\mathbf{J}_{\boldsymbol{x}}\left(\boldsymbol{\theta}^*,\boldsymbol{x}^*(\boldsymbol{H});\boldsymbol{H}\right)\nabla_{\overline{\boldsymbol{r}}}g\left(\overline{\boldsymbol{r}}^*\right)\leq 0, \quad (7)$$

$\forall\boldsymbol{x}\in\mathcal{X}$, where $\mathbf{J}_{\boldsymbol{x}}\left(\boldsymbol{\theta}^*,\boldsymbol{x}^*(\boldsymbol{H});\boldsymbol{H}\right)$ is the Jacobian matrix of the (approximate) rate vector $\boldsymbol{r}\left(\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{H}\right)\triangleq[r_1\left(\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{H}\right),...,r_K\left(\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{H}\right)]^T$ w.r.t. $\boldsymbol{x}$ at $\boldsymbol{\theta}=\boldsymbol{\theta}^*$ and $\boldsymbol{x}=\boldsymbol{x}^*(\boldsymbol{H})$, and $\nabla_{\overline{\boldsymbol{r}}}g\left(\overline{\boldsymbol{r}}^*\right)$ is the derivative of $g\left(\overline{\boldsymbol{r}}\right)$ at $\overline{\boldsymbol{r}}=\overline{\boldsymbol{r}}^*\triangleq\overline{\boldsymbol{r}}\left(\boldsymbol{\theta}^*,\Omega^*\right)$.

2)

$$\left(\boldsymbol{\theta}-\boldsymbol{\theta}^*\right)^T\nabla_{\boldsymbol{\theta}}g\left(\overline{\boldsymbol{r}}\left(\boldsymbol{\theta}^*,\Omega^*\right)\right)\leq 0,\forall\boldsymbol{\theta}\in\Theta, \quad (8)$$

where $\nabla_{\boldsymbol{\theta}}g\left(\overline{\boldsymbol{r}}\left(\boldsymbol{\theta}^*,\Omega^*\right)\right)\triangleq\mathbb{E}[\mathbf{J}_{\boldsymbol{\theta}}\left(\boldsymbol{\theta}^*,\boldsymbol{x}^*(\boldsymbol{H});\boldsymbol{H}\right)]\nabla_{\overline{\boldsymbol{r}}}g\left(\overline{\boldsymbol{r}}^*\right)$ is the partial derivative of $g\left(\overline{\boldsymbol{r}}\left(\boldsymbol{\theta}^*,\Omega^*\right)\right)$ w.r.t. $\boldsymbol{\theta}$ at $\boldsymbol{\theta}=\boldsymbol{\theta}^*$ and $\Omega=\Omega^*$, $\mathbf{J}_{\boldsymbol{\theta}}\left(\boldsymbol{\theta}^*,\boldsymbol{x}^*(\boldsymbol{H});\boldsymbol{H}\right)$ is the Jacobian matrix of the (approximate) rate vector $\boldsymbol{r}\left(\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{H}\right)$ w.r.t. $\boldsymbol{\theta}$.

Note that a stationary solution $(\boldsymbol{\theta}^*,\Omega^*)$ of $\mathcal{P}$ may not satisfy all the integer constraints in (4). To obtain an integer solution for the quantization bits allocation, we use the same method as in [3] to round each $d_{n,l}^*$ to its nearby integer.

IV. Online Optimization Algorithm

A. Summary of the BC-SSCA Algorithm

The proposed online BC-SSCA algorithm is summarized in Algorithm 1 and its time line is illustrated in Fig. 2. In BC-SSCA, an auxiliary weight vector $\boldsymbol{\mu}=[\mu_1,...,\mu_K]^T$ is introduced to approximate the derivative $\nabla_{\overline{\boldsymbol{r}}}g\left(\overline{\boldsymbol{r}}\left(\boldsymbol{\theta},\Omega\right)\right)$. At the beginning of each coherence time of channel statistics, the BBU resets the BC-SSCA algorithm with an initial analog filter phase vector $\boldsymbol{\theta}^0$ and a weight vector $\boldsymbol{\mu}^0$. Then $\boldsymbol{\theta}$ and $\boldsymbol{\mu}$ are updated once at the end of each frame, where $\boldsymbol{\theta}$ is updated by maximizing a concave surrogate function $\bar{f}^t\left(\boldsymbol{\theta}\right)$ of $g\left(\overline{\boldsymbol{r}}^{\circ}\left(\boldsymbol{\theta},\overline{\Omega}\right)\right)$ w.r.t. $\boldsymbol{\theta}$. Specifically, let $\boldsymbol{\theta}^t$ and $\boldsymbol{\mu}^t$ denote the analog filter phase vector and weight vector used during the t-th frame. The t-th iteration (t-th frame) of the BC-SSCA algorithm is described as follows.

Step 1 (Short-term optimization at each time slot): At time slot $i\in[tT_s+1,(t+1)T_s]$ in the t-th frame, the BBU first acquires the effective channel $\left(\boldsymbol{F}_n^t\right)^H\boldsymbol{H}_n\left(i\right),\forall n$, where $\boldsymbol{H}_n\left(i\right)$ is the channel state of RRH n at time slot i, and $\boldsymbol{F}_n^t$ is the analog filtering matrix at RRH n corresponding to $\boldsymbol{\theta}^t$. Then it calculates the short-term variables $\boldsymbol{x}^{J_t}\left(\boldsymbol{\mu}^t,\boldsymbol{\theta}^t,\boldsymbol{H}(i)\right)$ from $\left(\boldsymbol{F}_n^t\right)^H\boldsymbol{H}_n(i),\forall n$ by running a *short-term block-coordinate (BC) algorithm* with input J_t, $\boldsymbol{\mu}^t,\boldsymbol{\theta}^t$ and $\boldsymbol{H}_n(i)$, where J_t determines the total number of iterations for the short-term BC algorithm at frame t. For any finite iteration $t<\infty$, J_t is finite, and we let $J_t\to\infty$ as $t\to\infty$. Note that $\boldsymbol{x}^{J_t}\left(\boldsymbol{\mu}^t,\boldsymbol{\theta}^t,\boldsymbol{H}(i)\right)$ depends on $\boldsymbol{\theta}^t,\boldsymbol{H}(i)$ only through the effective channels $\left(\boldsymbol{F}_n^t\right)^H\boldsymbol{H}_n(i)$'s. Specifically, for given input J, $\boldsymbol{\mu},\boldsymbol{\theta}$ and $\boldsymbol{H}$, the short-term BC runs J iterations to find a stationary point (up to certain accuracy) $\boldsymbol{x}^J\left(\boldsymbol{\mu},\boldsymbol{\theta},\boldsymbol{H}\right)$ of the following weighted sum-rate maximization problem (WSRMP):

$$\mathcal{P}_S\left(\boldsymbol{\mu},\boldsymbol{\theta},\boldsymbol{H}\right):\max_{\boldsymbol{x}=\left[\boldsymbol{p}^T,\boldsymbol{v}^T,\boldsymbol{d}^T,\boldsymbol{u}^T\right]^T}\sum_{k=1}^{K}\mu_k r_k\left(\boldsymbol{\theta},\boldsymbol{x};\boldsymbol{H}\right).$$

The details will be postponed to Section IV-B.

Step 2 (Long-term optimization at the end of frame t): In Step 2a, the BBU obtains a full channel sample $\boldsymbol{H}^t\triangleq\boldsymbol{H}(tT_s+1)$ before the end of t-th frame. Then, in Step 2b (at the end of the t-th frame), the BBU updates the surrogate function $\bar{f}^t\left(\boldsymbol{\theta}\right)$ based on $\boldsymbol{H}^t$, the current iterate $\boldsymbol{\theta}^t$, and the short-term control variables $\boldsymbol{x}\left(i\right)\triangleq\boldsymbol{x}^{J_t}\left(\boldsymbol{\mu}^t,\boldsymbol{\theta}^t,\boldsymbol{H}(i)\right),\forall i\in\mathcal{T}_t\triangleq[tT_s+1,(t+1)T_s]$ as

$$\bar{f}^t\left(\boldsymbol{\theta}\right)=g\left(\hat{\mathbf{r}}^t\right)+\left(\mathbf{f}^t\right)^T\left(\boldsymbol{\theta}-\boldsymbol{\theta}^t\right)-\tau\left\|\boldsymbol{\theta}-\boldsymbol{\theta}^t\right\|^2, \quad (9)$$

Algorithm 1 BC-SSCA Algorithm

Initialize: $\boldsymbol{\theta}^0 \in \Theta$; $\boldsymbol{\mu}^0 = [1, ..., 1]^T$, $t = 0$.
Step 1 (Short-term optimization at time slot $i \in \mathcal{T}_t$):
Apply the short-term BC algorithm with input J_t, $\boldsymbol{\mu}^t$, $\boldsymbol{\theta}^t$ and $\boldsymbol{H}_n(i)$, to obtain the short-term variable $\boldsymbol{x}^{J_t}\left(\boldsymbol{\mu}^t, \boldsymbol{\theta}^t, \boldsymbol{H}(i)\right)$.
Step 2 (Long-term optimization at the end of frame t):
2a: Obtain a full channel sample $\boldsymbol{H}^t \triangleq \boldsymbol{H}(tT_s + 1)$.
2b: Update the surrogate function $\bar{f}^t(\boldsymbol{\theta})$ according to (9). Calculate $\bar{\boldsymbol{\mu}}^t = \nabla_{\overline{\boldsymbol{r}}} g\left(\hat{\boldsymbol{r}}^t\right)$ and update $\boldsymbol{\mu}^{t+1}$ according to (11).
2c: Solve (12) to obtain $\bar{\boldsymbol{\theta}}^t$. Update $\boldsymbol{\theta}^{t+1}$ according to (13).
Let $t = t + 1$ and return to Step 1.

where $\tau > 0$ is a constant; $\hat{\boldsymbol{r}}^t = [\hat{r}_1^t, ..., \hat{r}_K^t]^T$ is approximate average data rate updated recursively as

$$\hat{r}_k^t = (1 - \rho_t)\,\hat{r}_k^{t-1} + \rho_t \sum_{i \in \mathcal{T}_t} \frac{r_k\left(\boldsymbol{\theta}^t, \boldsymbol{x}(i); \boldsymbol{H}(i)\right)}{|\mathcal{T}_t|}, \forall k, \quad (10)$$

with $\hat{r}_k^{-1} = 0, \forall k$; $\mathbf{f}^t$ is an approximation of the partial derivative $\nabla_{\boldsymbol{\theta}} g\left(\overline{\boldsymbol{r}}\left(\boldsymbol{\theta}, \Omega\right)\right)$, which is updated recursively as

$$\mathbf{f}^t = (1 - \rho_t)\,\mathbf{f}^{t-1} + \rho_t \mathbf{J}_{\boldsymbol{\theta}}\left(\boldsymbol{\theta}^t, \boldsymbol{x}\left(tT_s + 1\right); \boldsymbol{H}^t\right) \nabla_{\overline{\boldsymbol{r}}} g\left(\hat{\boldsymbol{r}}^t\right),$$

with $\mathbf{f}^{-1} = \mathbf{0}$, where $\rho_t \in (0, 1]$ is a sequence satisfying $\frac{1}{\rho_t} \leq O\left(t^\kappa\right)$ for some $\kappa \in (0.5, 1)$, $\mathbf{J}_{\boldsymbol{\theta}}(\boldsymbol{\theta}, \boldsymbol{x}; \boldsymbol{H})$ is the Jacobian matrix of the rate vector $\boldsymbol{r}(\boldsymbol{\theta}, \boldsymbol{x}; \boldsymbol{H})$ w.r.t. $\boldsymbol{\theta}$ and its expression is derived in [6]. The weight vector $\boldsymbol{\mu}$ is updated as

$$\boldsymbol{\mu}^{t+1} = (1 - \gamma_t)\,\boldsymbol{\mu}^t + \gamma_t \bar{\boldsymbol{\mu}}^t. \quad (11)$$

with $\bar{\boldsymbol{\mu}}^t \triangleq \nabla_{\overline{\boldsymbol{r}}} g\left(\hat{\boldsymbol{r}}^t\right)$, where $\gamma_t \in (0, 1]$ is a sequence satisfying $\sum_t \gamma_t = \infty$, $\sum_t \left(\gamma_t\right)^2 < \infty$ and $\lim_{t \to \infty} \gamma_t / \rho_t = 0$.

In Step 2c, the optimal solution $\bar{\boldsymbol{\theta}}^t$ of the following quadratic optimization problem is solved:

$$\bar{\boldsymbol{\theta}}^t = \underset{\boldsymbol{\theta} \in \Theta}{\operatorname{argmax}}\ \bar{f}^t(\boldsymbol{\theta}), \quad (12)$$

which has closed-form solution $\bar{\boldsymbol{\theta}}^t = \mathbb{P}_\Theta\left[\boldsymbol{\theta}^t + \frac{\mathbf{f}^t}{2\tau}\right]$, where $\mathbb{P}_\Theta[\cdot]$ denotes the projection on to the box feasible region Θ. Finally, $\boldsymbol{\theta}$ is updated according to

$$\boldsymbol{\theta}^{t+1} = (1 - \gamma_t)\,\boldsymbol{\theta}^t + \gamma_t \bar{\boldsymbol{\theta}}^t. \quad (13)$$

Then the above iteration is carried out until convergence. In the full version in [6], we established the convergence of the BC-SSCA algorithm to stationary solutions.

B. Short-term Block-Coordinate Algorithm

We first transform the WSRMP $\mathcal{P}_S(\boldsymbol{\mu}, \boldsymbol{\theta}, \boldsymbol{H})$ to the following weighted minimum mean square error (WMMSE) problem

$$\min_{\boldsymbol{\beta}, \boldsymbol{v}, \boldsymbol{d}, \boldsymbol{u}, \boldsymbol{w}} \sum_{k=1}^K \mu_k\left(w_k \eta_k - \log w_k\right) \quad (14)$$
$$\text{s.t. } \boldsymbol{d} \geq \mathbf{0},\ (2) \text{ and } (3),$$

where $\boldsymbol{w} = [w_1, ..., w_K]$ with $w_k > 0 : \forall k$ is a weight vector for MSE, $\boldsymbol{\beta} = [\beta_1, ..., \beta_k]^T$ with $|\beta_k|^2 = p_k$ and

$$\begin{aligned}\eta_k &\triangleq \mathbb{E}\left[\left|s_k - \hat{s}_k\right|^2 |\boldsymbol{H}\right] \\ &= \left|1 - \boldsymbol{u}_k^H \widetilde{\boldsymbol{V}}^H \boldsymbol{h}_k \beta_k\right|^2 + \sum_{l \neq k} \left|\boldsymbol{u}_k^H \widetilde{\boldsymbol{V}}^H \boldsymbol{h}_l \beta_l\right|^2 \\ &+ \boldsymbol{u}_k^H \widetilde{\boldsymbol{V}}^H \widetilde{\boldsymbol{V}} \boldsymbol{u}_k + \boldsymbol{u}_k^H \boldsymbol{Q}(\boldsymbol{\theta}, \boldsymbol{p}, \boldsymbol{v}, \boldsymbol{d})\, \boldsymbol{u}_k,\end{aligned}$$

is the MSE of user k. Following similar proof to that of Theorem 1 in [7], it can be shown that Problem $\mathcal{P}_S(\boldsymbol{\mu}, \boldsymbol{\theta}, \boldsymbol{H})$ is equivalent to (14). Therefore, we shall focus on designing a BC algorithm to find a stationary point of (14). In the proposed BC algorithm, starting from an initial point, the short-term control variables $\boldsymbol{\beta}, \boldsymbol{v}, \boldsymbol{d}, \boldsymbol{u}, \boldsymbol{w}$ are optimized in an alternating way by solving a convex subproblem w.r.t. each variable. The update equation for each variable is elaborated below.

When fixing the other short-term variables, the optimal $\boldsymbol{u}$ is given by the MMSE receiver

$$\boldsymbol{u}_k = \left(\sum_{l=1}^K \widetilde{\boldsymbol{V}}^H \boldsymbol{h}_l |\beta_l|^2 \boldsymbol{h}_l^H \widetilde{\boldsymbol{V}} + \widetilde{\boldsymbol{V}}^H \widetilde{\boldsymbol{V}} + \boldsymbol{Q}\right)^{-1} \widetilde{\boldsymbol{V}}^H \boldsymbol{h}_k \beta_k,$$

$\forall k$, where $\boldsymbol{Q}$ is an abbreviation for $\boldsymbol{Q}(\boldsymbol{\theta}, \boldsymbol{p}, \boldsymbol{v}, \boldsymbol{d})$; the optimal $w_k, \forall k$ is given by $w_k = \left(1 - \boldsymbol{u}_k^H \widetilde{\boldsymbol{V}}^H \boldsymbol{h}_k \beta_k\right)^{-1}$; and the optimal $\boldsymbol{\beta}$ is given by $\beta_k = \beta_k^*(\lambda_k), \forall k$ with

$$\begin{aligned}\beta_k^*(\lambda_k) = {} & \mu_k w_k \mathrm{Re}\left[\boldsymbol{u}_k^H \widetilde{\boldsymbol{V}}^H \boldsymbol{h}_k\right] \\ & \times \left(\sum_{l=1}^K 2\mu_l w_l \boldsymbol{h}_k^H \widetilde{\boldsymbol{V}} \boldsymbol{u}_l \boldsymbol{u}_l^H \widetilde{\boldsymbol{V}}^H \boldsymbol{h}_k + \nu_k + 2\lambda_k\right)^{-1},\end{aligned}$$

where $\nu_k = \sum_{n,l} \frac{6}{4^{d_{n,l}}} |u_{k,n,l}|^2 |\boldsymbol{h}_{n,k}^H \widetilde{\boldsymbol{v}}_{n,l}|^2$, $u_{k,n,l}$ is the $((n-1)N + s)$-th element of $\boldsymbol{u}_k$, λ_k is zero if $|\beta_k^*(0)|^2 \leq P_k$ and chosen to satisfy $|\beta_k^*(\lambda_k)|^2 = P_k$ otherwise.

When fixing the other short-term variables, we solve the following modified subproblem w.r.t. $\boldsymbol{v}$ by adding a proximal regularization term $\epsilon \left\|\boldsymbol{v} - \boldsymbol{v}'\right\|^2$ with $\epsilon > 0$ a small number:

$$\min_{\boldsymbol{v}} \sum_{k=1}^K \mu_k\left(w_k \eta_k - \log w_k\right) + \epsilon \left\|\boldsymbol{v} - \boldsymbol{v}'\right\|^2, \quad (15)$$

where $\boldsymbol{v}'$ is the current digital filter. By solving (15), we obtain the updated digital filter as follows

$$\boldsymbol{v} = (\boldsymbol{B} + \epsilon \boldsymbol{I})^{-1}(\boldsymbol{J} + \epsilon \boldsymbol{v}'), \quad (16)$$

where $\boldsymbol{B} = [\boldsymbol{B}_{1,1}, ..., \boldsymbol{B}_{N,L}]^T$ and $\boldsymbol{B}_{n,l} = [\boldsymbol{B}_{1,1,n,l}, ..., \boldsymbol{B}_{N,L,n,l}]^T$ with

$$\begin{aligned}&\boldsymbol{B}_{n',l',n,l} \\ &= \begin{cases} \sum_{k=1}^K \mu_k w_k |u_{k,n,l}|^2 (\frac{3}{4^{d_{n,l}}} + 1) \boldsymbol{D}_n, & n' = n, l' = l, \\ \sum_{k=1}^K \mu_k w_k u_{k,n,l}^* u_{k,n,l'} \boldsymbol{D}_n, & n' = n, l' \neq l, \\ \sum_{k=1}^K \mu_k w_k u_{k,n,l}^* u_{k,n',l'} \boldsymbol{D}_{n,n'}, & n' \neq n, \end{cases}\end{aligned}$$

$$\boldsymbol{D}_n = \boldsymbol{F}_n^H \boldsymbol{F}_n + \sum_{k=1}^K \beta_k^2 \boldsymbol{F}_n^H \boldsymbol{h}_{n,k} \boldsymbol{h}_{n,k}^H \boldsymbol{F}_n,$$

$$\boldsymbol{D}_{n,n'} = \sum_{k=1}^K \beta_k^2 \boldsymbol{F}_n^H \boldsymbol{h}_{n,k} \boldsymbol{h}_{n',k} \boldsymbol{F}_n,$$

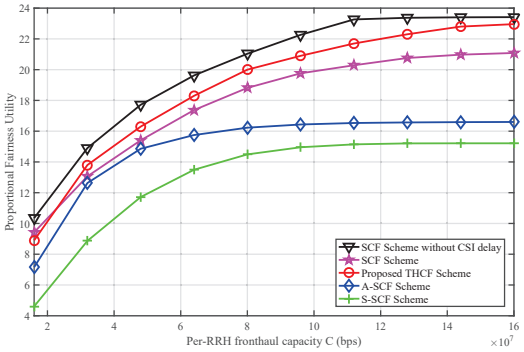

Figure 3: PFS utility versus per-RRH fronthaul capacity C

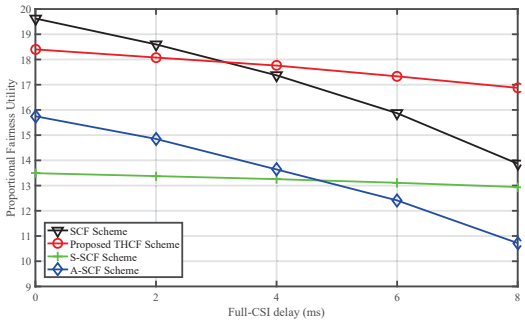

Figure 4: PFS utility versus the CSI delay.

and $\boldsymbol{J} = [\boldsymbol{J}_{1,1}, ..., \boldsymbol{J}_{N,L}]^T$ with $\boldsymbol{J}_{n,l} = \sum_{k=1}^{K} \mu_k w_k \beta_k u_{k,n,l}^* \boldsymbol{F}_n^H \boldsymbol{h}_{n,k}, \forall n, l$.

Finally, the optimal quantization bits allocation is obtained by solving the KKT conditions as

$$d_{n,l}^*(\lambda_n) = \left[\frac{\log 2B\lambda_n - \log(\log 4 \sum_{k=1}^{K} \mu_k w_k \varsigma_{k,n,l})}{\log 4} \right]^+,$$

$\forall n, l$, where $\varsigma_{k,n,l} = 3|u_{k,n,l}|^2 (\sum_{k=1}^{K} p_k |\boldsymbol{h}_{n,k}^H \widetilde{\boldsymbol{v}}_{n,l}|^2 + \|\widetilde{\boldsymbol{v}}_{n,l}\|^2)$ and the optimal Lagrange multiplier $\lambda_n \geq 0$ is chosen such that $2B \sum_{l=1}^{L} d_{n,l}^*(\lambda_n) = C_n$.

V. Simulation Results and Discussions

Consider a C-RAN with 4 RRHs placed in a circle cell of radius 500 m. There are 8 users randomly distributed in the cell. The channel bandwidth is 1 MHz. We adopt the geometry-based channel model in [8] for simulations. Unless otherwise specified, we consider $M = 64$ antennas, $S = 16$ RF chains for each RRH. There are $T_s = 10$ time slots in each frame and the slot size is 1 ms. The coherence time for the channel statistics is 10 s. As in [9], we assume that the CSI delay is proportional to the dimension of the channel vector that is required at the BS. The carrier frequency is 2.14 GHz and the velocity of users is 3 Km/h. The CSI delay for the full channel matrix is set to be $\tau = 4$ ms except for Fig. 4. We consider PFS utility. Three baseline schemes are considered for comparison: the spatial-compression-and-forward (SCF) scheme in [3], the analog SCF (A-SCF) scheme in [4] and the slow-timescale SCF (S-SCF) obtained by removing the short-term optimization in the proposed scheme.

Fig. 3 shows the performance comparison of different schemes versus per-RRH fronthaul capacity C varies from $C = 16$ Mbps to $C = 160$ Mbps. It can be observed that the best PFS utility is achieved by the SCF scheme without CSI delay, followed by the proposed THCF scheme. Furthermore, the proposed THCF scheme achieves significant gain over A-SCF and S-SCF, which demonstrates the importance of hybrid analog-and-digital processing and two-timescale joint optimization. Finally, it is observed that the performance of SCF is inferior to the proposed THCF since the full-CSI delay is larger than the effective-CSI delay. In Fig. 4, we plot the PFS utility versus the CSI delay, where the the per-RRH fronthaul capacity is fixed as $C = 64$Mbps. We can see that as the CSI delay increases, the PFS of all schemes decreases gradually. It is observed that the PFS achieved with the proposed THCF scheme is higher than that achieved by the other schemes for moderate and large full-CSI delay. This is because the performance of the proposed THCF scheme is insensitive to the full-CSI delay. Although the performance of the S-SCF scheme is also insensitive to the full-CSI delay, its performance is still much worse than the proposed THCF scheme due to the lack of optimal power control and quantization bits allocation.

VI. Conclusion

We propose a two-timescale hybrid compression and forward (THCF) scheme to reduce the fronthaul consumption in Massive MIMO aided C-RAN. We formulate the optimization of THCF as a general utility maximization problem, and propose a BC-SSCA algorithm to find stationary solutions of this two-stage non-convex stochastic optimization problem. Simulations verify that the proposed BC-SSCA algorithm achieves significant gain over existing solutions.

References

[1] N. Chen, B. Rong, X. Zhang, and M. Kadoch, "Scalable and flexible massive MIMO precoding for 5G H-CRAN," *IEEE Wireless Communications*, vol. 24, no. 1, pp. 46–52, February 2017.

[2] Y. Zhou and W. Yu, "Optimized backhaul compression for uplink cloud radio access network," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1295–1307, June 2014.

[3] L. Liu and R. Zhang, "Optimized uplink transmission in multi-antenna C-RAN with spatial compression and forward," *IEEE Transactions on Signal Processing*, vol. 63, no. 19, pp. 5083–5095, Oct 2015.

[4] L. Combi and U. Spagnolini, "Hybrid beamforming in RoF fronthauling for millimeter-wave radio," in *2017 European Conference on Networks and Communications (EuCNC)*, June 2017, pp. 1–5.

[5] X. Zhang, A. Molisch, and S.-Y. Kung, "Variable-phase-shift-based RF-baseband codesign for MIMO antenna selection," *IEEE Trans. Signal Processing*, vol. 53, no. 11, pp. 4091–4103, Nov. 2005.

[6] A. Liu, X. Chen, W. Yu, V. Lau, and M.-J. Zhao, "Two-timescale hybrid compression and forward for massive MIMO aided C-RAN," *in preparation*, 2018.

[7] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, "An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel," *IEEE Trans. Signal Processing*, vol. 59, no. 9, pp. 4331 –4340, Sept. 2011.

[8] S. Park, J. Park, A. Yazdan, and R. W. Heath, "Exploiting spatial channel covariance for hybrid precoding in massive MIMO systems," *IEEE Trans. Signal Processing*, vol. 65, no. 14, pp. 3818–3832, July 2017.

[9] A. Liu and V. K. N. Lau, "Phase only RF precoding for massive MIMO systems with limited RF chains," *IEEE Trans. Signal Processing*, vol. 62, no. 17, pp. 4505–4515, Sept. 2014.