Synchronization for Multi-hop Distributed MIMO-OFDM

Yong Jun Chang, Qiongjie Lin and Mary Ann Weitnauer*
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, Georgia 30332-0250
Email:changyongjun@gmail.com; lin3@gatech.edu; maweit@gatech.edu

***Abstract*—Multi-hop or cascaded distributed multi-input and multi-output (DMIMO) occurs in a series of clusters of single-antenna radios, when the clusters relay to each other, using concurrent transmission one cluster to the next, within a cooperative route. We address the case of no external synchronization, where one cluster derives its transmit synchronization from a packet received from the previous cluster. For DMIMO system based on orthogonal frequency division multiplexing (OFDM) transmission, frequency offsets between cooperating transmitters or cluster members induce inter-carrier-interference (ICI) in the receivers of the next cluster. Also, the timing offsets between transmitters create additional delay spread potentially necessitating a longer guard interval. Rather than estimating each of the multiple offsets, each receiver estimates just one offset, which implicitly is a channel gain-weighted average, known from our previous work to yield statistically stable synchronization from hop to hop. To effectively suppress the timing and frequency offsets between distributed transmitters, we designed a novel preamble which facilitates this one-step estimation without explicit knowledge of channel gains. The performance improvement of the estimation algorithm is evaluated through simulation as well as experiment on GNU Radio and USRPs.**

I. Introduction

Distributed multiple-input-multiple-output (DMIMO) has been gaining attention recently in the context of multi-hop networks as a means for highly reliable and low-latency cooperative routing, when the MIMO channel is exploited for transmit and receive diversity [1][2][3][4], or as a means for high-throughput ad hoc networking, when the MIMO channel is used with spatial multiplexing [5][6]. In both cases, a cooperative route is realized as a series of relaying clusters, where one cluster concurrently relays a packet to the next cluster. This paper focuses on the type of multi-hop DMIMO where relays do not share the message with each other prior to transmission and when there is no external synchronization for the cluster, such as GPS. Instead, in our proposed method, each relay derives its transmit time and carrier frequency from the super-imposed versions of a preamble, received through several diversity channels from the previous cluster. We note that synchronization for relay transmission is more demanding than synchronization for decoding, because worst-case synchronization for decoding at the relays in one cluster plus multipath spreads in the next hop can exceed the delay and frequency spread tolerances in the receivers of the next hop.

The authors gratefully acknowledge partial support for this research from the National Science Foundation, grant number CNS-1017984.

Synchronization for DMIMO is challenging because each wireless node generates its carrier signal and clock from a local oscillator, and although local oscillators on different nodes are designed to have the same nominal frequency, they generally have slightly different frequencies because of tolerances in manufacturing and variations in temperature [7]. In orthogonal frequency division multiplexing (OFDM), which is popular in standards today because of its high bandwidth efficiency and convenient implementation, these differences cause carrier frequency offsets (CFOs) between cooperative transmitters, which can induce inter-carrier-interference (ICI). Furthermore, the processing time that is required to process the data packet in each transmitter is highly unpredictable [8][9], contributing to a large delay spread at the receiver, which could exceed the cyclic prefix.

Most of the previous work in synchronization for cooperative or distributed transmission explicitly estimates each of the multiple offsets and then exploits those estimates to equalize or otherwise compensate the offsets in the destination receiver [10][11][12]. The estimation can be done using either a time-division multiplexing (TDM)-based training sequences [10] or an approach designed especially for simultaneous or concurrent transmission by the cooperative transmitters [11][12]. In the TDM-based approach, each relay transmits in a different time slot, causing excessive overhead when there are more than two cooperating transmitters, e.g., in opportunistic large arrays [13] or in a wide cooperative route [3][4].

Several authors have addressed explicit offsets estimation for simultaneous transmission [11][12]. In [11], orthogonal training preambles are proposed to estimate multiple timing and frequency offsets for OFDM distributed space-time block codes (STBCs). The method estimates the multiple offsets at the destination and feeds back the estimates to the relays so that the data can arrive in a synchronous manner. However, the feedback operation requires extensive resources in some applications. Furthermore, the feedback might not be possible if the cooperation creates range extension where the destination cannot reach back to the relay by itself. Also, although our proposed preambles are similar to [11] in that they are both orthogonal and OFDM-based, the preambles in [11] would not enable the low-complexity, single-estimate approach that we propose. Linearly independent training signals simultaneously transmitted from the relays are proposed in [12]. Even though they utilize the multiple estimated offsets for decoding in a two-hop OFDM-based cooperative system, they do not discuss about the pre-synchronization for a multi-hop DMIMO network. We note that explicit offset estimation usually also requires knowledge of the number of offsets, which is problematic in opportunistic large array-based approaches, such as [4][14][13], because the number of relays in any particular cluster is not known a priori.

All of the approaches that explicitly estimate each offset tend to be computationally intense. In contrast, in our approach, the receiver directly computes just one offset estimate for each of time and frequency, with no explicit knowledge of channel gains, and with relatively low computational complexity. The proposed algorithm uses just a single OFDM preamble to estimate the time and frequency with a reasonable constraint that the CFO is bounded by half the subcarrier spacing. Furthermore, our fractional CFO estimate is of the form that produces convergent estimation error statistics as a function of number of hops [15][16]. The main contributions in this paper are (1) a novel OFDM preamble design that enables our low-complexity method while effectively suppressing the timing and frequency offsets between distributed transmitters, (2) the low-complexity signal processing method in the receiver that computes the estimate, and (3) experimental demonstration.

The performance of proposed algorithm is compared with Schmid & Cox's method [17], and Park's method [18], which are popular OFDM synchronization methods for the SISO link, and Ding's method [19] which has transmitter diversity but is designed only for non-distributed MIMO.

II. Proposed Method

A. System Model and Preamble Structure

We consider a multi-hop DMIMO network as shown in Fig. 1. The source node, labelled 'S' in the figure, broadcasts a message to the nodes $R_1^{(1)}$ and $R_2^{(1)}$ within its transmission range. $R_1^{(1)}$ and $R_2^{(1)}$ form a virtual antenna array that retransmits the source message to the next array, simultaneously. The nodes in the next array use the superimposed signals from the previous array as a pre-synchronization reference. Nodes in each hop do timing and frequency synchronization using the proposed algorithm. The estimations in each receiver are then used for pre-synchronization of re-transmission to the next hop to establish the multi-hop DMIMO system [15][16].

The proposed preamble structure in the time and frequency domains is illustrated in Fig. 2. In the frequency domain, all odd subcarriers are set to zero. This causes the inverse Fourier transform of the sequence to be repeated in the time domain as in [17]. One OFDM symbol consists of N subcarriers, which are divided into Z sectors $\{P_0, P_1, \cdots, P_{Z-1}\}$ excluding Q length of a guard band on both sides. The guard band gives a safety margin to avoid aliasing in the presence of CFO. Each sector is designed to have K non-zero, even-indexed subcarriers, which have a one-to-one correspondence with each of the K relays in each cluster [1]. Therefore, the number of sectors becomes $Z = (N-2Q)/2K$. As shown in the figure, the subcarriers for each relay are assigned in left-to-right in the left band, while they are assigned right-to-left in the right band. The symmetric preamble of the k^{th} relay can be written as $q_k(l) = q_k(Z-n-1)$, $n = 0, 1, \ldots, Z/2-1$. Z preamble symbols $[q_k(0), q_k(1), \ldots, q_k(Z-1)]$ for the k^{th} relay are inserted into corresponding locations in each sector as

$$P_k(n) = \begin{cases} q_k(n-Q), & Q \le n < N-Q \\ 0, & \text{otherwise,} \end{cases}$$

where $P_k(n)$ is the frequency domain preamble symbol of the k^{th} relay. The constructed preamble in the frequency domain is converted into the time-domain through the inverse discrete Fourier transform (IDFT) and a cyclic prefix is attached to the time-domain preamble. For the sake of convenience, we assume that the time-domain preamble starts at discrete index zero and the attached cyclic prefix has negative time index as shown in the figure. It is pointed out that the preambles transmitted from K relays are orthogonal in frequency since they choose different subcarriers.

Let $p_k[n]$ denote the time-domain preamble of the k^{th} relay. The transmission signal at the k^{th} relay with cyclic prefix G can be written as

$$s_k(n) = \begin{cases} p_k(n), & 0 \le n < N \\ p_k(n+N), & -G \le n < 0. \end{cases} .$$

As discussed above, the time domain preamble has repeated sequences in the time domain as $p_k(n) = p_k(N/2+n)$ for $n = 0, 1, \ldots, \frac{N}{2}-1$. In addition, since all transmissions are orthogonal for the preambles, we have

$$\sum_{n=0}^{N-1} p_i^*(n) \cdot p_j(n) = 0, \quad \text{for all } i, j \text{and } i \ne j.$$

Assuming that the k^{th} relay transmits its preamble with normalized time offset ϵ_k to sampling period and normalized frequency offset ω_k to subcarrier spacing, the received signal from the k^{th} relay at the receiver can be expressed as

$$r(n) = \sum_{k=1}^{K} e^{j2\pi\omega_k \frac{n}{N}} \left\{ \sum_{l=0}^{L-1} h_k(l) s_k(n-l-\epsilon_k) \right\} + v(n), \tag{1}$$

where $\mathbf{h}_k = [h_k(0), \ldots, h_k(L-1)]^T$ and $v(n)$ are the channel impulse response and an AWGN with variance σ_v^2, respectively. L is the number of resolvable mutipath components.

Letting $\mathbf{r} = [r(0), \ldots, r(N-1)]^T$, $\mathbf{\Gamma}_k = diag\{1, e^{j2\pi\omega_k/N}, \ldots, e^{j2\pi\omega_k(N-1)/N}\}$, $[\mathbf{S}_k]_{mn} = s_k(m-n-\epsilon_k)$, where $m = 0, \ldots, N-1$ and $n = 0, \ldots, L-1$, the received signal from the k^{th} relay can be expressed in a matrix form as $\mathbf{r}_k = \mathbf{\Gamma}_k \cdot \mathbf{S}_k \cdot \mathbf{h}_k$. The superimposed signal from K relays can be expressed as a vector form as $\mathbf{r} = \mathbf{\Lambda} \cdot \mathbf{h} + \mathbf{v}$ where $\mathbf{\Lambda} = [\mathbf{\Gamma}_1\mathbf{S}_1, \ldots, \mathbf{\Gamma}_K\mathbf{S}_K]$ and $\mathbf{h} = [\mathbf{h}_1^T, \ldots, \mathbf{h}_K^T]^T$ and $\mathbf{v} = [v(0), \ldots, v(N-1)]^T$.

Since the vector $\mathbf{v}$ in the above equation is a vector of AWGN variables with the same mean and variance, the joint ML estimator of the timing and CFO parameters can be expressed as

$$(\hat{\epsilon}, \hat{\omega}) = \arg\max_{\epsilon, \omega} \left\{ \mathbf{r}^H \mathbf{\Lambda} (\mathbf{\Lambda}^H \mathbf{\Lambda})^{-1} \mathbf{\Lambda}^H \mathbf{r} \right\}. \tag{2}$$

Since the maximization in Eq.(2) requires a search over the $2K$ dimensional space spanned by (ϵ, ω), the ML estimation is not practical.

[1] This assignment can be achieved by piggybacking on "HELLO" messages by a simple distributed algorithm[20].

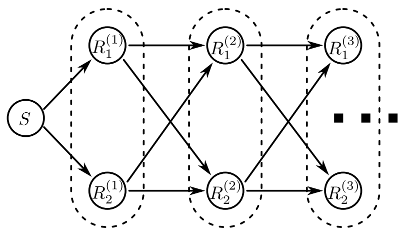


Fig. 1. Multi-hop DMIMO with K=2.

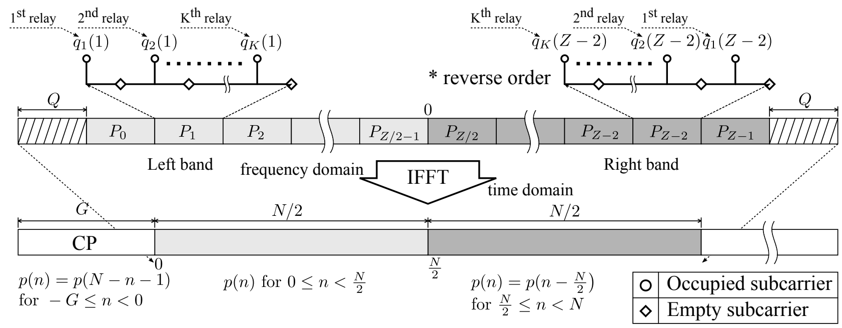


Fig. 2. Illustration of the proposed preamble structure in time and frequency domain.

B. Coarse Timing Estimation

The coarse timing of the preamble location can be obtained by delayed auto-correlation of the two halves of the preamble in the time domain [17]. From the construction of the preambles, the first $N/2$ half of the preamble is identical to the second $N/2$ half. The correlation function and the normalized function for the coarse timing estimation are calculated as

$$A_c(m) = \sum_{n=0}^{N/2-1} r^*(m+n)r(m+n+\frac{N}{2}) \tag{3}$$

$$B_c(m) = \frac{1}{2}\sum_{n=0}^{N-1} \left|r(m+n)\right|^2 \tag{4}$$

and the timing metric becomes $M_c(m) = \frac{|A_c(m)|}{B_c(m)}$. Since $s_k(n) = s_k(n+N/2)$, the coarse timing metric within the GI becomes $M_c(m) = 1$ for ideal channel. By the Cauchy Schwarz inequality, it is readily shown that $A_c(m) \leq B_c(m)$ for all m. Therefore, the coarse timing metric around the preamble location for ideal (noiseless) channel becomes

$$M_c(m) \begin{cases} = 1, & -G \leq m < 0 \\ < 1, & \text{otherwise.} \end{cases}$$

However, when considering the timing spread caused by the timing error among distributed transmitters, the peak of timing metric will be shifted. Therefore, the coarse timing will be used to define a window for following estimation processes. Once the timing metric $M_c(m)$ exceeds a predefined threshold, the coarse timing can be found by searching

$$\tilde{m}_c = \arg\max_{m \in \mathcal{M}} \{M_c(m)\}$$

where $\mathcal{M}$ is a set of adjacent sample points that exceed the threshold. The threshold is a user-defined system parameter, which determines the packet detection rate directly.

We next show, for the Rayleigh flat fading channel (L=1), the role that the different channel gains play in this metric. The correlation function can be written as

$$\begin{aligned} A_c(m) = \sum_{n=0}^{N/2-1} \Bigg\{ & \sum_{k=1}^{K} \left|h_k\right|^2 e^{j\pi\omega_k} \left|s_k(n+m-\epsilon_k)\right|^2 \\ & + \sum_{\substack{x=1 \\ x\neq y}}^{K}\sum_{y=1}^{K} h_x^* h_y e^{j\theta_{xy}} s_x^*(n+m-\epsilon_x) s_y(n+m-\epsilon_y) \\ & + v^*(m+n)\sum_{k=1}^{K} h_k e^{j\pi\omega_k} s_k(n+m-\epsilon_k) \\ & + v(m+n+\frac{N}{2})\sum_{k=1}^{K} h_k^* e^{-j\pi\omega_k} s_k^*(n+m-\epsilon_k) \\ & + v^*(m+n)v(m+n+\frac{N}{2})\Bigg\}. \end{aligned} \tag{5}$$

The second term of (5) becomes zero because of the orthogonal property of preamble in frequency domain. Then we reverse the order of summation and take expectation to yield

$$E\{A_c(m)|\mathbf{h_k}\} = \sum_{1\leq k\leq K} \left|h_k\right|^2 e^{j\pi\omega_k} \Big[\sum_{n=0}^{N/2-1} \left|s_k(n+m-\epsilon_k)\right|^2 \Big]. \tag{6}$$

$B_c(m)$ can be similarly simplified. In the SISO case, the square-bracketed term is maximized over m at the SOP. But in DMIMO, the square-bracketed terms are weighted by $|h_k|^2 e^{j\pi\omega_k}$. For small fractional CFOs, ω_k, the channel gains $|h_k|^2$ provide relative weighting, which in turn gives the diversity gain in estimation.

C. Fractional Combined-CFO Estimation

In [15][16], we show that the weighted average of multiple start of packet (SOP) times and CFOs, weighted by the channel gain, minimizes the error variance of the retransmission time and frequency. In this section, we show how this type of estimate can be obtained for the fractional CFO, under the assumption that the CFOs from the K relays differ by less than half of the subcarrier spacing as $|w_x - w_y| < 0.5$, for all x and y.

When $K = 1$, (6) can be written as

$$A_c(m) = \sum_{n=0}^{N/2-1} |h|^2 \cdot e^{j\pi\omega} \cdot \left|s(m+n)\right|^2.$$

Referring to [17], at the coarse timing point $\tilde{m}_c$ the CFO can be estimated as $\hat{\omega} = \frac{\angle A_c(\tilde{m}_c)}{\pi}$. The phase angle operator $\angle(\cdot)$ wraps the angle to the interval $[-\pi, \pi]$ corresponding to the normalized frequency interval $[-1, 1]$. Therefore, $\hat{\omega}$ is considered as a fractional part of the CFO. The residual CFO η, referred to as an integer part of CFO, will be treated in the next subsection.

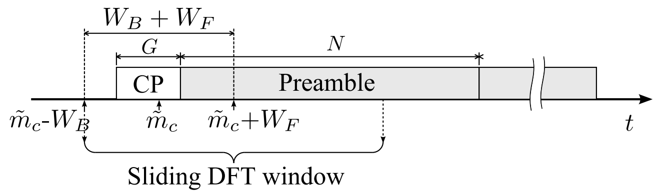


Fig. 3. Illustration of defining the searching window

For $K > 1$, the combined CFO can be written as[2]

$$\omega_c = \frac{\sum_{k=1}^{K} |h_k|^2 \omega_k}{\sum_{k=1}^{K} |h_k|^2}. \tag{7}$$

In Appendix A, assuming that the relative CFO errors are small enough, we show (7) can be approximated by simply taking the phase angle of $A_c(\tilde{m}_c)$ as

$$\hat{\omega}_c \approx \frac{\angle A_c(\tilde{m}_c)}{\pi}. \tag{8}$$

It is pointed out that this operation is the same as $K = 1$ so that the fractional CFO estimation method can be used regardless of the number of relays.

D. Integer CFO Estimation

Once the coarse timing $\tilde{m}_c$ of the SOP is estimated, it can be used to define the searching window for the following estimation processes. From the property of the coarse timing estimation, $\tilde{m}_c$ is within the guard interval. To reduce the searching dimension and computational complexity, we set the searching window around the coarse timing location as shown in Fig. 3. The backward and forward window sizes are user-defined parameters, which are denoted as W_B and W_F respectively in the figure. N samples $r(m)$ for $m = m_s, \ldots, m_s + N - 1$ where $\tilde{m}_c - W_B \leq m_s < \tilde{m}_c + W_F$ are converted into frequency domain samples through the sliding DFT window. Let $R_m(n)$ denote the DFT output of the samples $\{r(m), \ldots, r(m+N-1)$ for $n = 0, \ldots, N-1\}$, and let the actual preamble start at $m = 0$. In this section, we assume that the combined fractional CFO is already compensated. Since the residual CFO error η is an integer, the output from the DFT window is shifted by η. The DFT output of the received preamble from the k^{th} relay that is windowed starting at m can be written as

$$R_{k,m}(n) = \psi_k(n) \cdot e^{j2\pi \frac{m+n}{N}} \cdot P_k(n + \eta), \tag{9}$$

where $\psi_k(n)$ and $z(n)$ are the channel frequency response of the n^{th} subcarrier from the k^{th} relay node and AWGN with variance σ_z^2, respectively. Also the vector form of the DFT output can be written as

$$\mathbf{R}_{k,m} = \mathbf{\Psi}_k \cdot \mathbf{\Phi}_m \cdot \mathbf{P}_{k,\eta},$$

where

$$\mathbf{\Psi}_k = \text{diag}\{\psi_k(0), \ldots, \psi_k(N-1)\}$$
$$\mathbf{\Phi}_m = \text{diag}\{e^{j2\pi \frac{m}{N}}, \ldots, e^{j2\pi \frac{m+N-1}{N}}\}$$
$$\mathbf{P}_{k,\eta} = [P_k(\eta), \ldots, P_k(N-1+\eta))]^T.$$

[2]This "center-of-mass" form produced convergent statistics in [15][16].

Then the superimposed preamble in frequency domain transmitted from K relays can be written in vector form as $\mathbf{R}_m = \mathbf{\Phi}_m \cdot \mathbf{P}_\eta \cdot \mathbf{\Psi} + \mathbf{z}$ where $\mathbf{P}_\eta = [\mathbf{P}_{1,\eta}, \ldots, \mathbf{P}_{K,\eta}]$, $\mathbf{\Psi} = [\mathbf{\Psi}_1, \ldots, \mathbf{\Psi}_K]^T$, and $\mathbf{z} = [z(0), \ldots, z(N-1)]$.

The vector $\mathbf{z}$ in the above equation is a vector of AWGN variables with the same mean and variance. Let us assume that the integer part of the CFO is within $[-Q, Q]$; in the other words, the guard band Q limits the maximum integer CFO the system can detect, but could be adjusted according to the minimal synchronization requirement. This makes the DFT output of the received preamble not aliased regardless of the CFO amount. Without having exact knowledge of the channel impulse response $\mathbf{\Psi}$, the ML estimator of the integer CFO η at given starting point m can be readily expressed as $\hat{\eta}_m = \underset{\eta \in [-Q,Q]}{\arg\max} \left\{ \left\| \mathbf{\Theta}(m, \eta) \right\|^2 \right\}$ where $\mathbf{\Theta}(m, \eta) = \mathbf{R}_m^H \cdot \mathbf{P}_\eta$.

Within the searching window, the integer part of CFO can be found by searching all possible m to maximize $\mathbf{\Theta}(m, \eta)$ as $\hat{\eta} = \max\{\mathbf{\Theta}(m, \hat{\eta}_m)\}$ for $\tilde{m}_c - W_B \leq m < \tilde{m}_c + W_F$.

E. Fine timing estimation for each relay

By the shifting property of DFT, the time offset from the actual preamble location causes phase rotation across subcarriers of the DFT output. Assuming that the combined fractional and integer part of CFO is already estimated and compensated, the fine preamble timing within the searching window for each relay can be found by using frequency-domain symmetric correlation (FSC). The proposed timing metric for the k^{th} relay using FSC can be defined as

$$A_k(m) = \Bigg| \sum_{l=0}^{L/2-1} R_m^*\big(Q + 2(Kl + k)\big) \cdot R_m\big(N - Q - 2(Kl + k) - 1\big) \Bigg|. \tag{10}$$

For the k^{th} relay, the fractional index of SOP timing can be estimated using quadratic interpolation as

$$\tilde{m}_k = \frac{1}{2} \frac{A_k(\bar{m}_k - 1) - A_k(\bar{m}_k + 1)}{A_k(\bar{m}_k - 1) - 2A_k(\bar{m}_k) + A_k(\bar{m}_k + 1)},$$

where $\bar{m}_k = \arg\max_m \{A_k(m)\}$. The interpolated peak value of $A_k(m)$ becomes

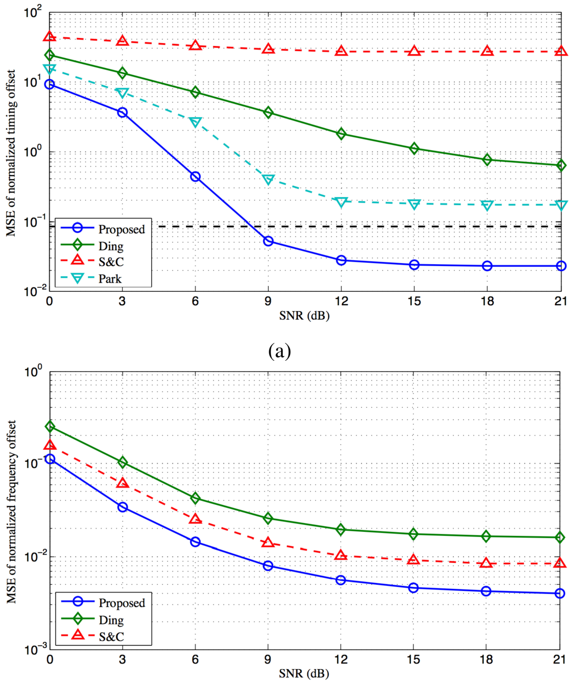
$$\tilde{A}_k(\tilde{m}_k) = A_k(\bar{m}_k) - \frac{1}{4}\big(A_k(\bar{m}_k - 1) - A_k(\bar{m}_k + 1)\big)\tilde{m}_k.$$

By constructing $|q(l)|^2 = 1$ for all l and assuming the frequency-flat fading channel, it can be readily shown that the weighted average of $\tilde{m}_k$ weighted by $A_k(\tilde{m}_k)$ becomes the combined start of packet (SOP) as

$$\hat{m} = \frac{\sum_{k=1}^{K} \tilde{A}_k(\tilde{m}_k)\tilde{m}_k}{\sum_{k=1}^{K} \tilde{A}_k(\tilde{m}_k)}.$$

III. Simulation Results

The mean square error (MSE) performance of the proposed method when $K = 2$ is evaluated by computer simulation. The OFDM system with 256 subcarriers and frequency-selective channel with the exponential power delay profile model [21] are used. The GI length, normalized to the sampling duration,



(a)

(b)

Fig. 4. Simulation result of the proposed time (a) and frequency (b) estimation method with $\omega_1 = 0.05$, $\omega_2 = -0.05$, and $\sigma_\epsilon^2 = 1$ in the frequency-selective fading channel

is eight. The transmission time, normalized by the sampling period of each relay, is a zero-mean Gaussian random variable with variance σ_ϵ^2=1. In addition, the normalized CFOs of the first transmitters are deterministically set to 0.05 and -0.05, respectively. The proposed method is compared with Schimidl & Cox's [17] and Park's [18] methods, which are labeled as 'S&C' and 'Park' in Fig.4. respectively. Representing the simple extension of the single-antenna-based estimation methods, these methods assign the same preamble to the two transmitters, keeping the same transmit power per node. On the other hand, Ding's method [19] is designed to exploit transmit diversity by using an orthogonal preamble structure for non-distributed MIMO.

The simulated MSE performances of the timing estimation methods are compared in Fig. 4(a). The timing error is normalized by the sampling period. We observe that the S&C method has high MSE compared to the others. While this error does not affect the OFDM decoding performance as long as it less than the length of the GI, it would create a large delay spread when multiple relays rely on this estimate for DMIMO. It is also observed that Park's method has better MSE performance than Ding's, despite that Ding's method is designed to exploit transmit diversity. It can be also observed that all the methods have an error floor. The error floor, denoted as a black dashed line in the figure, is due to the discrete sampling and CFOs between distributed transmitters. The error floor of the proposed method, however, is less than the error floor of the others because the fine timing between two samples is estimated by quadratic interpolation.

Fig. 4(b) shows the MSE performance of the frequency estimation methods. For this simulation, the Park's method is excluded because its frequency estimation performance is identical to the S&C's method. The simulation result clearly shows that the proposed method is superior to the others, as it conducts CFO estimation within a searching window.

IV. Experimental Results

The performance of the proposed method is evaluated on the experimental testbed on GNU Radio and USRPs. The same OFDM parameters as in the simulation have been used. For the performance metrics, we define the rms transmit time spread (RTTS) and rms transmit frequency spread (RTFS) as

$$\sigma_\epsilon = \sqrt{\frac{\sum_k^K (\epsilon_k - \hat{\epsilon})^2}{K-1}} \tag{11}$$

where ϵ_k is an estimated quantity of time or frequency at the k^{th} relay node and $\hat{\epsilon}$ is the sample mean of the K time or frequency estimates. The measurements were taken in the Smart Antenna Research Laboratory of Centergy Building, Georgia Institute of Technology. The source, the observer and the two relay nodes, denoted as 'S', 'O', 'R_1', and 'R_2' respectively, are deployed in the lab as shown in Fig. 5. Therefore, $K = 2$ in (11). A common clock is provided to the source and observer and another common clock is provided to the two relays, for the purpose of measuring relative error only. In other words, the common clocks were not used in the synchronization process, but only to measure synchronization performance. The source transmits a packet with the proposed preamble to the two relays. Each relay autonomously estimates the time and frequency from the received preamble, and re-transmits the packet after a deterministic period of time, T_{proc}, which is long enough that the SDRs have finished processing with 0.9 probability[15] with the proposed pre-synchronization algorithm. Specifically, R_1 and R_2 choose different subcarrier slots in the proposed preamble and retransmit the source packet with the preamble. Next the source and observer nodes estimate their time and frequency synchronization parameters using the proposed method, as though they were relays. This 3-hop sequence was repeated 1000 times and the estimated time and frequency data was collected in each cluster.

Fig. 6 and 7 show the empirical cumulative distributed function (CDF) of the RTTS and RTFS in each cluster, respectively. The measurement shows that the RTTS in the relay cluster is less than 0.13μsec in 90% of the cases, which is relatively small compared to the sampling duration that is 1μsec. It is noted that the bumpy shape of the RTTS curve is caused by a bimodal distribution of quadratic interpolation. The RTTS in the source cluster is less than 0.15μsec in 90% of the cases, which is slightly larger than the one in the relay cluster. This is because the reference for the synchronization in the source cluster's nodes is the superimposed received signals from the relay cluster, while the reference for the relay cluster is from a single transmitter. In Fig. 7, the RTFS of the source cluster is larger than the RTFS of the relay cluster by three times at the 90% level. As with RTTS, the errors at the source cluster are higher than the relay cluster because the synchronization is based on a superposition rather than just a single transmission. However, the RTFS of the source cluster

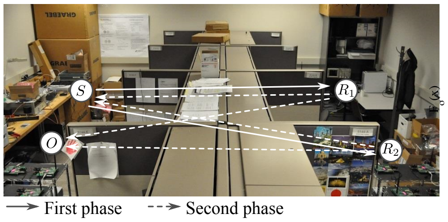


Fig. 5. Node placement for experiment

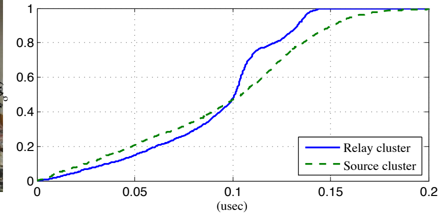


Fig. 6. Empirical CDF of the RTTS in Relay and Source cluster.

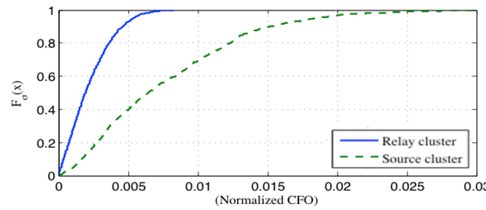


Fig. 7. Empirical CDF of the RTFS in Relay and Source cluster.

is still less than 1.5% of the subcarrier spacing, at the 90% level.

V. Conclusion

In this paper, a method for time and frequency pre-synchronization for OFDM-based DMIMO was proposed. The proposed low-complexity estimates implicitly exploit the relative relay-to-receiver channel gains to provide a diversity benefit. The performance of the proposed algorithm was evaluated through computer simulation, and demonstrated on an experimental testbed on GNU Radio and USRPs. The simulation and experimental results show that the transmit time and frequency offsets of OFDM-based DMIMO can be effectively suppressed by the proposed pre-synchronization method.

Appendix

To simplify the problem, we assume that $K = 2$. From the trigonometry identity,

$$A_1 e^{j\omega_1} + A_2 e^{j\omega_1} = A_3 e^{j\omega_3} \tag{12}$$

where $A_3^2 = A_1^2 + A_2^2 + 2A_1 A_2 \cos(\Delta\omega)$ and

$$\omega_3 = \arctan\left(\frac{A_1 sin(\omega 1) + A_2 sin(\omega 2)}{A_1 cos(\omega 1) + A_2 cos(\omega 2)}\right)$$

where $\Delta\omega = \omega_1 - \omega_2$. Since we assume the difference of the angles ω_1 and ω_2 is small, from the small-angle approximation of *cosine* A_3 can be approximated as $A_1 + A_2$. By plugging $A3$ into (12), we can have

$$e^{j\omega_3} \approx \frac{A_1}{A_1 + A_2} e^{j\omega_1} + \frac{A_2}{A_1 + A_2} e^{j\omega_2}. \tag{13}$$

Setting the new variable ω_c as a weighted average of ω_1 and ω_2 weighted by A_1 and A_2, then (13) can be rewritten as

$$e^{j\omega_3} = e^{j\omega_c}\left(\frac{A_1}{A_1 + A_2} e^{j\hat{\omega}_1} + \frac{A_2}{A_1 + A_2} e^{j\hat{\omega}_2}\right).$$

where $\hat{\omega}_1 = \omega_1 - \omega_c$ and $\hat{\omega}_2 = \omega_2 - \omega_c$. It is noted that $|\hat{\omega}_1| << 1$ and $|\hat{\omega}_2| << 1$ by construction. By the approximation of a exponential function, (13) can be approximated as

$$e^{j\omega_3} \approx e^{j\omega_c}\left(\frac{A_1}{A_1 + A_2}\hat{\omega}_1 + \frac{A_2}{A_1 + A_2}\hat{\omega}_2\right).$$

Similarly the second term of the right-hand side is also small, the above equation can be approximated to the product of two exponential terms as

$$\begin{aligned} e^{j\omega_3} &\approx e^{j\omega_c} \cdot e^{j(\frac{A_1}{A_1+A_2}\hat{\omega}_1 + \frac{A_2}{A_1+A_2}\hat{\omega}_2)} \\ &= e^{j(\frac{A_1}{A_1+A_2}\omega_1 + \frac{A_2}{A_1+A_2}\omega_2)} \end{aligned}$$

References

[1] S. et al., "Cooperative communications in mobile ad hoc networks," *IEEE Signal Processing Magazine*, vol. 23, no. 5, pp. 18–29, 2006.
[2] Y. J. Chang, H. Jung, and M. A. Ingram, "Demonstration of an OLA-based cooperative routing protocol in an indoor environment," Apr. 2011.
[3] J. Lin, H. Jung, Y. J. Chang, J. W. Jung, and M. A. Weitnauer, "On cooperative transmission range extension in multi-hop wireless ad-hoc and sensor networks: A review," to appear in Ad Hoc Networks.
[4] A. Blair, T. Brown, K. M. Chugg, T. R. Halford, and M. Johnson, "Barrage relay networks for cooperative transport in tactical MANETs," *IEEE MILCOM*, Nov. 2008.
[5] A. Ozgur, R. Johari, D. Tse, and O. Leveque, "Information-theoretic operating regimes of large wireless networks," *IEEE Trans. on Information Theory*, vol. 56, no. 1, pp. 427–437, 2010.
[6] Q. Lin, Y. J. Chang, F. Wang, and M. Weitnauer, "Implementation and analysis of multi-user mimo with amplify-and-forward relaying," in *Communications Workshops (ICC), 2014 IEEE International Conference on*, June 2014, pp. 32–37.
[7] J. R. Vig, "Quartz crystal resonators and oscillators; for frequency control and timing applications - A Tutorial," 2004.
[8] T. Schmid, O. Sekkat, and M. B. Srivastava, "An experimental study of network performance impact of increased latency in software-defined radios," ser. Proc. WinTECH. ACM, 2007, pp. 59–66.
[9] A. Mohammad, X. Hong, M. Islam, and K. Zunnurhain, "Delay analysis of wireless ad hoc networks: Single vs. multiple radio," in *IEEE LCN*, 2010, pp. 814–820.
[10] M.-K. Oh, X. Ma, G. Giannakis, and D.-J. Park, "Cooperative synchronization and channel estimation in wireless sensor networks," in *Asilomar Conference on Signals, Systems and Computers*, vol. 1, 2004, pp. 238–242 Vol.1.
[11] Q. Huang, M. Ghogho, J. Wei, and P. Ciblat, "Practical timing and frequency synchronization for OFDM-based cooperative systems," *IEEE Trans. on Signal Processing*, vol. 58, no. 7, pp. 3706–3716, 2010.
[12] A. Nasir, H. Mehrpouyan, S. Durrani, S. Blostein, R. Kennedy, and B. Ottersten, "Transceiver design for distributed STBC based AF cooperative networks in the presence of timing and frequency offsets," *IEEE Trans. on Signal Processing*, vol. 61, no. 12, pp. 3143–3158, 2013.
[13] A. Scaglione and Y.-W. Hong, "Opportunistic large arrays: cooperative transmission in wireless multihop ad hoc networks to reach far distances," *Signal Processing, IEEE Transactions on*, vol. 51, no. 8, pp. 2082 – 2092, Aug. 2003.
[14] L. Thanayankizil and M. A. Ingram, "Reactive robust routing with Opportunistic Large Arrays," *IEEE ICC Workshop*, June 2009.
[15] Y. J. Chang and M. A. Ingram, "Convergence property of transmit time pre-synchronization for concurrent cooperative communication," *IEEE GlobeCom*, Dec 2010.
[16] Z. Gao, Y. J. Chang, and M.-A. Ingram, "Synchronization for cascaded distributed MIMO communications," in *MILCOM*, 2010, pp. 387–392.
[17] T. Schmidl and D. Cox, "Robust frequency and timing synchronization for OFDM," *IEEE Trans. on Communications*, vol. 45, no. 12, pp. 1613–1621, 1997.
[18] B. Park, H. Cheon, C. Kang, and D. Hong, "A novel timing estimation method for OFDM systems," *IEEE Comm. Letters*, vol. 7, no. 5, pp. 239–241, 2003.
[19] W. Ding, F. Yang, J. Song, and L. He, "Signaling-embedded preamble design for OFDM system with transmit diversity," in *IEE IWCMC*, 2012, pp. 211–215.
[20] H. Jung, "Experimentation and physical layer modeling for opportunistic large array-based networks," 2014, phD Dissertation, School of Electrical and Computer Engineering, Georgia Institute of Technology.
[21] Y. S. Cho, J. Kim, W. Y. Yang, and C. G. Kang, *MIMO-OFDM wireless communications with MATLAB*. Wiley. com, 2010.