Acoustic Multicarrier ICI-mitigating Receiver

Amir Tadayon and Milica Stojanovic

CONTENTS

I. INTRODUCTION

High data rate transmission over acoustic channels is a challenging problem due to the combined effects of long multipath and Doppler fluctuations. Orthogonal frequency division multiplexing (OFDM) offers remarkable robustness against frequency-selective channels at reasonably low computational loads. This fact motivates the use of OFDM in mobile acoustic communications where the channel exhibits long multipath delays but each narrowband carrier only experiences flat fading, thus eliminating the need for time-domain equalizers.

In mobile acoustic systems, Doppler effect can be severe enough that the received OFDM signal experiences non-negligible frequency offsets even after initial resampling. To target these offsets, two methods, one based on a hypothesis-testing approach and another based on a stochastic gradient algorithm, are used. The methods rely on differentially coherent detection which keeps the receiver complexity at a minimum and requires only a small pilot overhead. Differential encoding is applied across carriers, promoting the use of a large number of closely spaced carriers within a given bandwidth. These approaches simultaneously support frequency-domain coherence and efficient use of bandwidth for achieving high bit rates. While frequency synchronization capitalizes on differentially coherent detection, it can also be used as a preprocessing stage in coherent receivers without creating undue complexity.

In situations with high mobility, however, OFDM systems are challenged by the time-variability of the channel, as the relative motion between transmitter and receiver creates carrier-dependent Doppler shifts, thus introducing inter-carrier interference (ICI) which has a detrimental impact on data detection performance. In such situations, not only does the receiver need to compensate for the residual frequency offsets but it also has to atone for the impact of time-variability of broadband acoustic channels as well as the effect of carrier-dependent Doppler shifts. We apply a method called coherent partial FFT (P-FFT) demodulation which targets pre-FFT signal processing to countervail the channel's time variation before ICI has been created in the process of demodulation. To do so, the time interval of one OFDM block is divided into several partial intervals, giving the channel less chance to change over each shorter interval, and demodulation is performed in each interval separately. The partial demodulator outputs are then combined in a weighted sum, thus compensating for the channel variation within a block. The combined signals are finally processed by a second stage, consisting of channel estimation and subsequent data detection.

Reliable coherent data detection requires the channel state information (CSI) at the receiver. Pilot-assisted channel estimation is used as a standard method to obtain the necessary CSI for reliable coherent communications. Unlike the conventional sample-spaced and sub-sample-spaced methods, such as least squares (LS) and orthogonal matching pursuit (OMP), which target the *taps* of an equivalent discrete-delay channel response, the path identification (PI) method targets the physical propagation *paths* in a continuous-delay domain, and focuses on explicit estimation of delays and complex amplitudes of the channel paths in an iterative fashion. This method also has the capability to track the time-variation of the channel assuming that the channel is slowly varying from one OFDM block to another.

When multiple receive elements are available, two situations are possible: one in which the array elements see uncorrelated channel responses, and another in which the channel responses are correlated. In the first case, channel estimation must be accomplished element-by-element. This is done simply by applying the PI algorithm to each element individually. In the second case, correlation between the elements can be exploited. In doing so, our goal is to reduce the signal processing complexity without compromising the performance. Towards this goal, we propose an adaptive precombining method. Without requiring any a priori knowledge about the spatial distribution of received signals, the method exploits spatial coherence between receive channels by linearly combining them into fewer output channels so as to reduce the number of subsequent channel estimators. The algorithm learns the spatial coherence pattern recursively over the carriers, thus effectively achieving broadband beamforming. The reduced-complexity precombining method relies on differential encoding which keeps the receiver complexity at a minimum and requires a very low pilot overhead.

After a short overview on the OFDM system model in the Sec. II, the algorithms proposed for the receiver blocks shown in Fig. 1 are briefly discussed in Sec. III-VI. In Sec. VII, instructions for running the Matlab implementation of the receiver are provided.

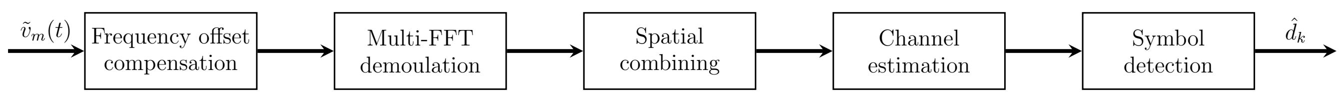

Fig. 1. Receiver block diagram.

II. System Model

We consider an OFDM system with K carriers within a total bandwidth B. Let f_0 and $\Delta f = B/K$ denote the first carrier frequency and carrier spacing, respectively. The transmitted OFDM block is then given by

$$s(t) = \sum_{k=0}^{K-1} s_k(t) = Re\left\{\sum_{k=0}^{K-1} d_k e^{2\pi i f_k t}\right\}, \quad t \in [0,\ T] \tag{1}$$

where $T = 1/\Delta f$ is the OFDM symbol duration. The data symbol d_k, which modulates the kth carrier of frequency $f_k = f_0 + k\Delta f$, belongs to a unit-amplitude phase shift keying alphabet (PSK).

The transmitted signal passes through a multipath acoustic channel whose impulse response can be modeled as

$$h(\tau, t) = \sum_{p=1}^{N_p} h_p(t)\delta(\tau - \tau_p + at) \tag{2}$$

where N_p is the number of physical propagation paths, $h_p(t)$ and $\tau_p(t)$ represent the gain and delay of the pth path, respectively, and $a = v/c$ is the Doppler factor corresponding to velocity v and propagation speed c.

After frame synchronization, initial resampling, down-shifting by the lowest carrier frequency f_0 and cyclic-prefix removal,[1] the lowpass equivalent signa received on the mth element is modeled as

$$\tilde{v}_m(t) = e^{i\beta t} \sum_{k=0}^{K-1} H_k^m(t) e^{2\pi i a' k \Delta f t} d_k e^{2\pi i k \Delta f t} + w_m(t),\ m = 1, \dots, M_r \tag{3}$$

where $H_k^m(t)$ is the channel frequency response at the kth carrier of the mth receiving element, a' is the effective residual Doppler scaling factor, and $w(t)$ is the equivalent lowpass complex noise.

OFDM demodulation on the signal $v(t)$, which is obtained after frequency offset compensation, carrier-dependent Doppler shift and time-variability mitigation, yields

$$y_k = \frac{1}{T}\int_T v(t) e^{-2\pi i k \Delta f t} dt = H_k d_k + z_k \tag{4}$$

where y_k is the demodulation output, H_k is the channel frequency at the kth carrier, and z_k is the corresponding noise.

III. Frequency Offset Compensation

A. Hypothesis Testing (HT) Approach

In this approach, whose block diagram is shown in Fig. 2, several hypothesized values of the frequency offset are used, e.g. with resolution of $\Delta f/10$, and differential maximal ratio combining (DMRC) is performed for each hypothesized value. Specifically, let us assume that the M_r signals are compensated by some hypothesized value $\hat{\beta}$, and that demodulation is

[1] In case of zero-pad OFDM signal, receiver performs overlap-and-add operation instead of cyclic-prefix removal.

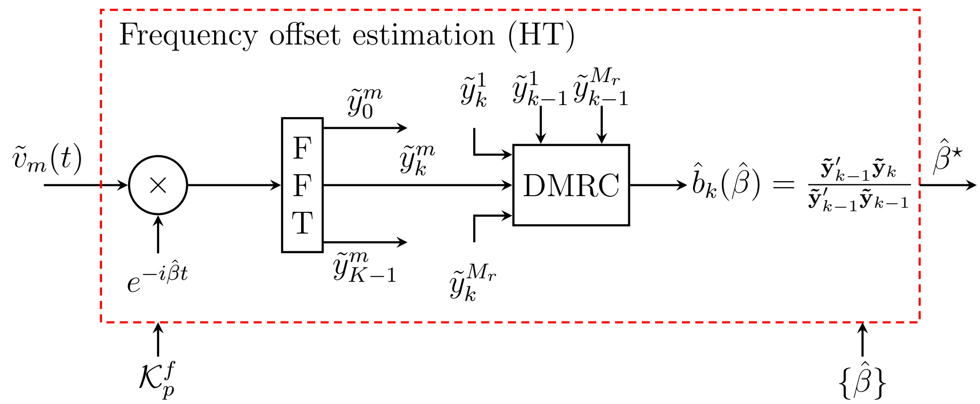

Fig. 2. Block diagram of hypothesis-testing approach.

performed on all the receiving elements to yield

$$\tilde{y}_k^m = \int_T \tilde{v}_m(t)e^{-i\hat{\beta}t}e^{-2\pi ik\Delta ft}dt,\ \ k=0,\dots,K-1,\ \ m=1,\dots,M_r \tag{5}$$

Arranging the signals corresponding to carrier k into a vector $\mathbf{y}_k$ and performing DMRC, the estimates of the differentially-decoded data symbols $b_k = d_{k-1}^* d_k$ are obtained as[2]

$$\hat{b}_k = \frac{\sum_{m=1}^{M_r}(\tilde{y}_{k-1}^m)^*\tilde{y}_k^m}{\sum_{m=1}^{M_r}(\tilde{y}_{k-1}^m)^*\tilde{y}_{k-1}^m} = \frac{1}{\tilde{\mathbf{y}}'_{k-1}\tilde{\mathbf{y}}_{k-1}}\tilde{\mathbf{y}}'_{k-1}\tilde{\mathbf{y}}_k \tag{6}$$

Using equally-spaced pilot data symbols $b_k, k\in\mathcal{K}_p^f$, where $\mathcal{K}_p^f$ is the set of pilot carriers, the composite squared error is formed as

$$E(\hat{\beta}) = \sum_{k\in\mathcal{K}_p} |b_k - \hat{b}_k|^2 \tag{7}$$

and the estimate $\hat{\beta}^\star$ is obtained as $\hat{\beta}^\star = \arg\min_{\hat{\beta}} E(\hat{\beta})$.

B. Stochastic Gradient (SG) Approach

The finite resolution of the frequency offset estimate obtained from the hypothesis testing approach gives rise to an error floor in the estimator variance. To reduce the variance of the estimate, we develop a stochastic gradient approach where the composite squared error (6) is used to close the loop and guide the estimation of β. The estimated offset $\hat{\beta}$ can be calculated iteratively as

$$\hat{\beta}(j+1) = \hat{\beta}(j) + K_\beta\gamma(j), \quad j\geq 0 \tag{8}$$

where $\gamma = -\frac{1}{2}\frac{\partial E(\hat{\beta})}{\partial\hat{\beta}}$, and K_β is the frequency offset update parameter. The initial point $\hat{\beta}(0)$ can be set to zero when the frequency offset $\beta/2\pi$ is a fraction of carrier spacing Δf. The algorithm can be set to run either for a prespecified maximum number of iterations N_I or a predefined error threshold η_f. Fig. 3 shows the block diagram of the stochastic gradient method. Further details about both hypothesis testing and stochastic gradient algorithms can be found in [1].

IV. Coherent Partial-FFT (P-FFT) Demodulation

We use the partial FFT demodulation (P-FFT), which targets pre-FFT signal processing to counteract the channel's time-variation *before* ICI has been created in the process of demodulation. To do so, the time interval of one OFDM block is divided into several partial intervals, giving the channel less chance to change over each shorter interval, and demodulation is performed in each interval separately. The partial demodulator outputs are then combined in a weighted sum, thus compensating for the channel variation within a block.

[2] $(\cdot)^*$ and $(\cdot)'$ denote complex conjugate and conjugate transpose, respectively.

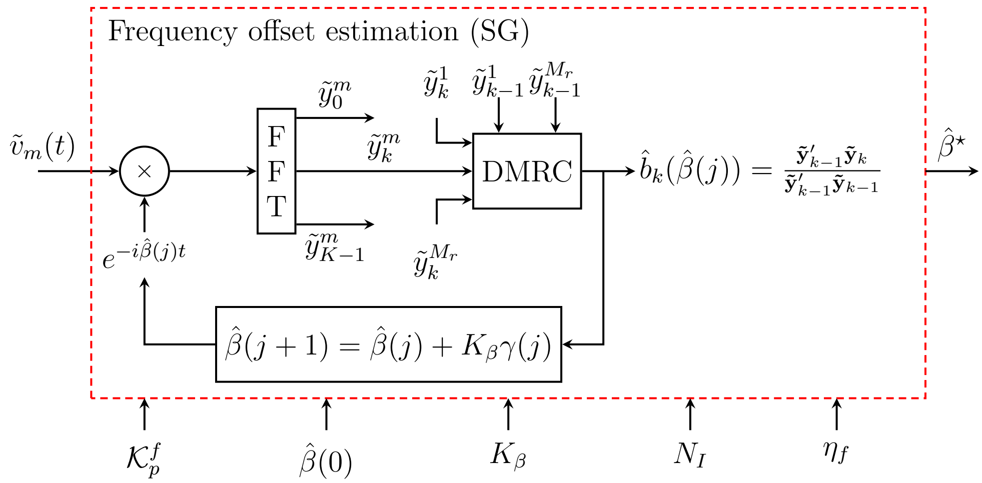

Fig. 3. Block diagram of stochastic gradient approach. The step size K_β has a notable impact on the convergence speed of the algorithm. To improve the convergence speed, the Barzilai–Borwein method is used to update the step size. In block-by-block receivers, the HT approach is used as an acquisition technique that can operate with an arbitrary range of hypothesized frequency offsets to compensate for the frequency offset in the first block. Capitalizing on the fact the frequency offset from one block to the next is not changing significantly, from the second block and on, the SG approach is employed with the initial value $\hat{\beta}(0)$ being set to the frequency offset estimated in the previous block.

A. Partial FFT Demodulation

Partial FFT demodulation with I intervals yields the observations

$$\tilde{y}_{k,i}^m = \frac{1}{T}\int_{\frac{iT}{I}}^{\frac{(i+1)T}{I}} \tilde{v}_m(t) e^{-2\pi i k \Delta f t} dt, \ \ i = 1, \dots, I \tag{9}$$

In practice, the same observations can be obtained by applying an FFT operation to the input $v_m(t)\phi_i(t)$, where $\phi_i(t), \ i = 1, \dots, I$ are the unit-amplitude rectangular pulses which divide the OFDM block into I non-overlapping sections. If the duration of each section is short enough, the corresponding channel variation can be considered negligible. Fig. 4 represents the block diagram of P-FFT demodulation with I FFT blocks.

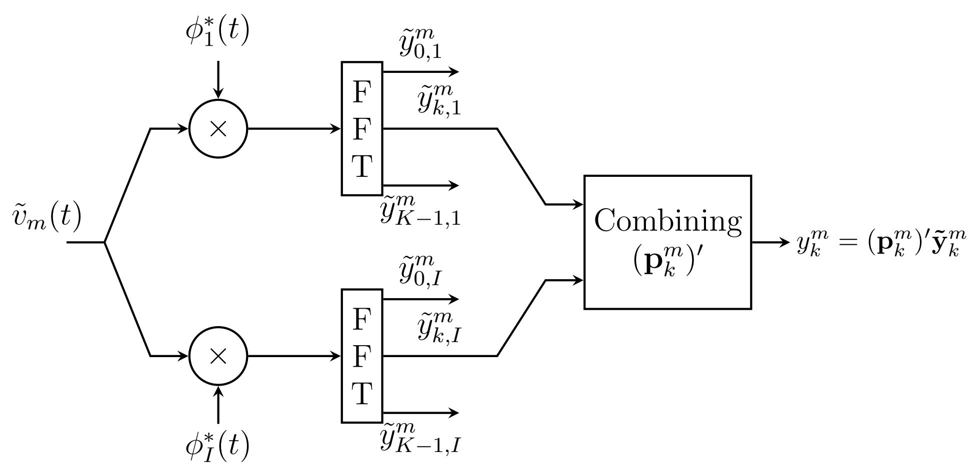

Fig. 4. Partial FFT demodulation with I intervals captured by functions $\phi_i(t), \ i = 1, \dots, I$. Weighted combining aims for suppression of ICI.

B. Coherent P-FFT combining

The observations obtained from P-FFT demodulators are linearly combined to produce the signals

$$y_k^m = \sum_{i=1}^{I} (p_{k,i}^m)^* \tilde{y}_{k,i}^m = (\mathbf{p}_k^m)'\tilde{\mathbf{y}}_k^m \tag{10}$$

where $\mathbf{p}_k^m$ is the column vector of P-FFT combiner coefficients, and $\mathbf{y}_k^m$ is the corresponding vector of demodulator outputs. Assuming that the signals y_k^m contain no ICI, these signals are modeled as

$$y_k^m = H_k^m d_k + z_k^m \tag{11}$$

where H_k^m is the equivalent post-combining channel coefficient, and z_k^m is the corresponding noise.

Assuming that the channel estimate $\hat{H}_k^m$ is available, the data symbol estimates are formed through maximum ratio combining (MRC) as

$$\hat{d}_k = \frac{\sum_{m=1}^{M_r} (\hat{H}_k^m)^* y_k^m}{\sum_{m=1}^{M_r} |\hat{H}_k^m|^2} \tag{12}$$

and the decisions $\tilde{d}_k$ are made on the so-obtained estimates. The MRC coefficients $\hat{H}_k^m$ and the P-FFT combiner vectors $\mathbf{p}_k^m$ can now be computed recursively over the carriers. The combining weight vectors at the next carrier $\mathbf{p}_{k+1}^m$ are computed by applying the RLS algorithm to the current values $\mathbf{p}_k^m$, the input $\tilde{\mathbf{y}}_k^m$ and the error $\epsilon_k^m = y_k^m - \hat{H}_k^m \tilde{d}_k$, where $\tilde{d}_k = d_k$ for k in the pilot set $\mathcal{K}_p^p$, or the decision made on the estimate. The equivalent channel estimate is computed as

$$\hat{H}_{k+1}^m = \alpha_H^p \hat{H}_k^m + (1 - \alpha_H^p) \frac{y_k^m}{\tilde{d}_k} \tag{13}$$

where $\alpha_H^p \in [0, 1)$, accounts for smoothing.

P-FFT combiner weights are initialized as a vector of all ones, and the first $2I$ carriers are designated as pilots for initial training. The algorithm is then switched into decision directed mode. To guard against error propagation in decision-directed mode, additional pilots are inserted periodically throughout the OFDM block.

The above expressions define the *pre-FFT* stage of processing, in which coherent combining of the partial FFT outputs is performed. Once the first stage is completed, the signals y_k^m and the tentative decisions $\tilde{d}_k$, acting as pilots, are fed to a channel estimator, where a refined channel estimate is formed using the path identification (PI) algorithm [2] that is briefly discussed below.

V. Channel Estimation

The physical, path-based channel model is introduced as an alternative to the conventional discrete-delay (sample-spaced) channel model. In the path-based channel model, the channel is parameterized by a pair $(\tau_p, h_p),\ p = 1, \ldots, N_p$ where the path delays τ_p have a continuum of values, the coefficients h_p represent the path amplitudes, and N_p is the number of channel paths. At the carrier frequency $f_k = f_0 + k\Delta f$, the channel frequency response is given by

$$H_k = H(f_k) = \sum_{p=1}^{N_p} h_p e^{-2\pi i f_k \tau_p} = \sum_{p=1}^{N_p} c_p e^{-2\pi i k \Delta f \tau_p} \tag{14}$$

where $c_p = h_p e^{-2\pi i f_0 \tau_p}$. The expression (14) can be written as

$$\mathbf{H} = \sum_{p=1}^{N_p} c_p \mathbf{s}_K (2\pi \Delta f \tau_p) \tag{15}$$

where $\mathbf{s}_K(2\pi\Delta f \tau) = \begin{bmatrix} 1 & e^{-2\pi i \Delta f \tau} & \ldots & e^{-2\pi i (K-1)\Delta f \tau} \end{bmatrix}^T$ is referred to as the steering vector at an arbitrary delay τ.

Assuming without the loss of generality that all K data symbols are available for channel estimation (e.g. correct symbol decisions, or all-pilots in an initial block), the input to the channel estimator is given by $x_k = y_k/d_k,\ k = 0, \ldots, K-1$, i.e.,

$$\mathbf{x} = \mathbf{H} + \mathbf{z} \tag{16}$$

where $\mathbf{z}$ is the noise vector. If the data symbols are available only on pilot carriers, the vector $\mathbf{x}$ is formed using those carriers only.

Consider now the following operation performed on the noisy channel observation $\mathbf{x}$:

$$x(\tau) = \frac{1}{K} \mathbf{s}'_K (2\pi\Delta f \tau) \mathbf{x} = \sum_{p=1}^{N_p} c_p g_K (2\pi \Delta f (\tau - \tau_p)) + z(\tau),\ \tau \in \tau_{obs} \tag{17}$$

where the interval τ_{obs} is a preset interval that captures the multipath spread, $z(\tau)$ is the corresponding noise, and

$$g_K(\varphi) = \frac{1}{K} \sum_{k=0}^{K-1} e^{ik\varphi} \tag{18}$$

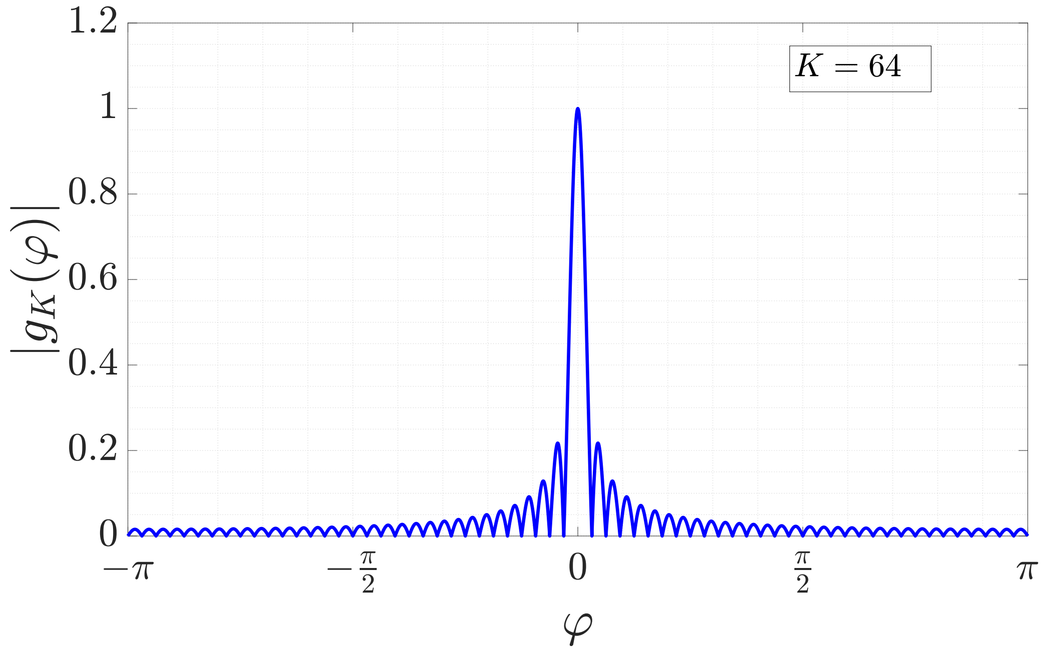

Fig. 5. Signature waveform.

is a known signature function. This operation corresponds to steering across the carriers. Fig. 5 shows the signature waveform (magnitude).

The fact that the signature waveform is *known* can be exploited to estimate the channel parameters explicitly. When we say explicitly, we mean that we are targeting directly both the path gains c_p and the path delays τ_p, unlike in the conventional estimation where the delay axis is discretized to avoid the non-linear problem of delay estimation.

Joint estimation of the parameters c_p and τ_p can be performed as follows. We start by setting

$$x_1(\tau) = x(\tau) \tag{19}$$

and evaluate this function for a preset range of delays τ with an arbitrary resolution $\Delta\tau$ in the delay domain. The range can be determined in accordance with the multipath spread T_{mp}. An iterative procedure now follows over the path indices $p = 1, \ldots, N_p$. In the pth iteration, we estimate the path delay and the path coefficient as

$$\hat{\tau}_p = \arg\max_{\tau} |x_p(\tau)| \tag{20a}$$

$$\hat{c}_p = x_p(\hat{\tau}_p) \tag{20b}$$

We then subtract this path's contribution from the current signal, so as to form the signal for the next iteration (next path)

$$x_{p+1}(\tau) = x_p(\tau) - \hat{c}_p g_K(2\pi\Delta f(\tau - \hat{\tau}_p)) \tag{21}$$

The procedure ends according to a predefined criterion such as an a priori set number of paths N_p, or when the power in the residual reaches a certain threshold η_c or stops to change significantly. Fig. 6 shows the block diagram of the PI algorithm.

An extension to the above algorithm can also be applied to improve the quality of the estimates $\hat{c}_p$. Once the algorithm has been executed, the path coefficients $\hat{c}_p$ generated in the process are discarded, but the delay estimates $\hat{\tau}_p$ are kept. The delay estimates are used to form the matrix

$$\hat{\mathbf{S}} = \begin{bmatrix} \mathbf{s}_K(2\pi\Delta f\hat{\tau}_1) & \cdots & \mathbf{s}_K(2\pi\Delta f\hat{\tau}_{N_p}) \end{bmatrix} \tag{22}$$

This matrix represents an estimate of the true matrix $\mathbf{S}$ which relates the vector $\mathbf{H}$ with the vector of path coefficients $\mathbf{c} = \begin{bmatrix} c_1 & \cdots & c_{N_p} \end{bmatrix}^T$ as $\mathbf{H} = \mathbf{Sc}$, i.e. it defines the observed signal as

$$\mathbf{x} = \mathbf{Sc} + \mathbf{z} \tag{23}$$

The corresponding LS estimate is $\hat{\mathbf{c}} = (\mathbf{S}'\mathbf{S})^{-1}\mathbf{S}'\mathbf{x}$. For lack of true $\mathbf{S}$, assuming that the delay estimates are accurate, we replace $\mathbf{S}$ by $\hat{\mathbf{S}}$. The channel coefficients are thus finally estimated as

$$\hat{\mathbf{c}} = (\hat{\mathbf{S}}'\hat{\mathbf{S}})^{-1}\hat{\mathbf{S}}'\mathbf{x} \tag{24}$$

Unlike the estimates (20b), which are obtained sequentially (one after another), these estimates are obtained jointly, and hence offer a potential improvement.

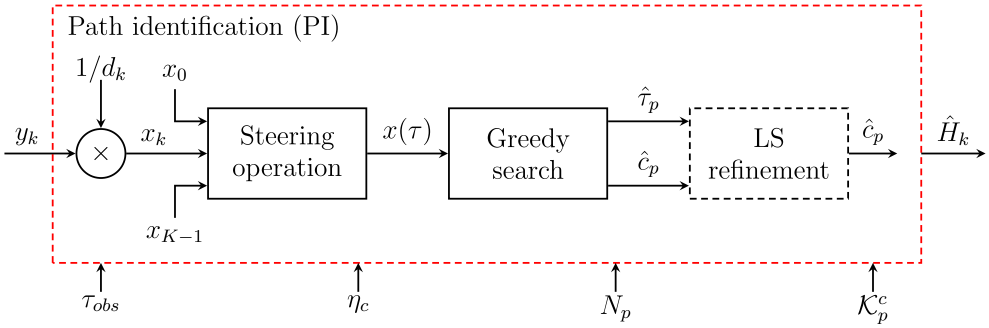

Fig. 6. Block diagram of the path identification (PI) algorith. The least squares (LS) refinement is optional.

VI. Spatial Coherence Learning

In a conventional coherent receiver with M_r spatially distributed elements, one FFT demodulator is associated with each input channel. OFDM demodulation then yields the M_r-element received signal vector

$$\mathbf{y}_k = d_k \mathbf{H}_k + \mathbf{z}_k, \quad k = 0, \ldots, K-1 \tag{25}$$

where $\mathbf{y}_k$ contains the demodulator outputs y_k^m, $m = 1, \ldots, M_r$, $\mathbf{H}_k$ and $\mathbf{z}_k$ contain the relevant channel and noise components, respectively. Using maximum ratio combining (MRC), the data symbols are then detected as

$$\hat{d}_k = \frac{\hat{\mathbf{H}}_k' \mathbf{y}_k}{\hat{\mathbf{H}}_k' \hat{\mathbf{H}}_k} \tag{26}$$

where $\hat{\mathbf{H}}_k$ is an estimate of the channel coefficients $\mathbf{H}_k$. If there is no spatial coherence, the M_r channel estimates $\hat{H}_k^m$, $m = 1, \ldots, M_r$ are formed independently. However, if there exists a correlation between the M_r channels, this correlation can be exploited to reduce the receiver complexity. This issue will be addressed in more details below.

To learn spatial distribution of the received signals, an adaptive precombining method is proposed. The technique reduces the complexity of coherent OFDM receivers through linearly combining the M_r input channels into $Q \leq M_r$ output channels, so as to reduce the number of channel estimators needed. The technique makes no assumption about the spatial distribution of signals, and relies on differential encoding which keeps the receiver complexity at a minimum. Fig. 7 shows the block diagram of the receiver.

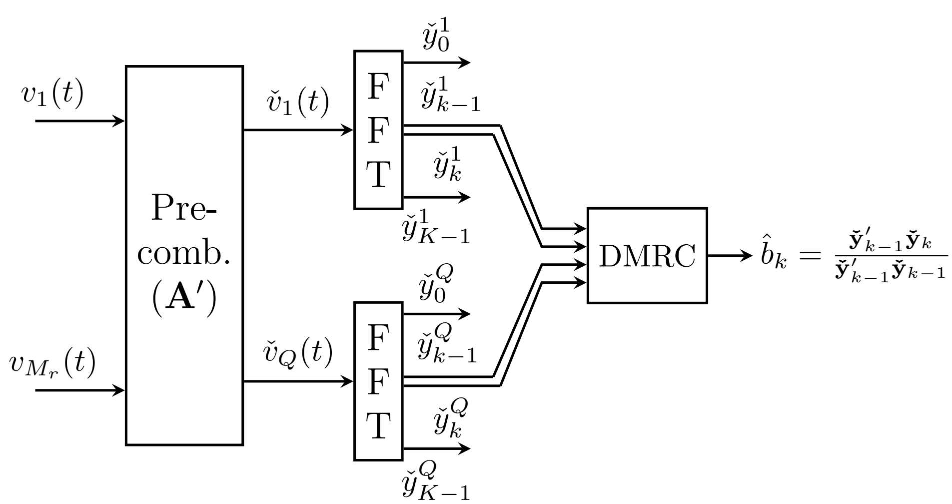

Fig. 7. Block diagram of the receiver with precombining. The precombiner $\mathbf{A}$ consisting of columns $\boldsymbol{\alpha}_q$, $q = 1, \ldots, Q$, linearly combines the M_r input channels into $Q \leq M_r$ output channels. FFT demodulation is then applied in each of the Q channels, followed by differential maximum ratio combining (DMRC). Alternatively, the order of precombining and FFT demodulation can be changed. FFT demodulation can be performed first on all M_r input channels, followed by precombining as indicated by the second equality in expression (23).

Let $\boldsymbol{\alpha}_q$ denote the precombiner weights for the qth output channel, and let the vector $\mathbf{v}(t)$ contain the M_r input signals $v_m(t)$. FFT demodulation then yields

$$\breve{y}_k^q = \frac{1}{T}\int_T \boldsymbol{\alpha}_q' \mathbf{v}(t) e^{-2\pi i k \Delta f t} dt = \boldsymbol{\alpha}_q' \mathbf{y}_k, \;\; q = 1, \ldots, Q \tag{27}$$

where the vector $\mathbf{y}_k, k = 0, \ldots, K-1$, contains the M_r FFT outputs corresponding to the M_r input channels. Expression (27) above shows that the receiver structure of Fig. 7, which employs pre-FFT combining, is equivalent to one that would employ post-FFT combining by applying the Q weights $\boldsymbol{\alpha}_q$ to the FFT outputs $\mathbf{y}_k$. We use this fact to develop a method for computing the weights recursively over the carriers.

Stacking the FFT outputs $\breve{y}_k^q$ into a column vector $\breve{\mathbf{y}}_k$, the differentially-decoded data symbols $b_k = d_{k-1}^* d_k$ are estimated by DMRC over the Q channels as

$$\hat{b}_k = \frac{\sum_{q=1}^{Q} (\breve{y}_{k-1}^q)^* \breve{y}_k^q}{\sum_{q=1}^{Q} (\breve{y}_{k-1}^q)^* \breve{y}_{k-1}^q} = \frac{\breve{\mathbf{y}}_{k-1}' \breve{\mathbf{y}}_k}{\breve{\mathbf{y}}_{k-1}' \breve{\mathbf{y}}_{k-1}} \tag{28}$$

Here, we implicitly assume that the channel frequency response changes slowly from one carrier to the next.

Treating $\breve{\mathbf{y}}_{k-1}$ as independent of the precombiner weights, and retaining only the dependence of $\breve{\mathbf{y}}_k$ on the precombiner weights, the estimated data symbol is expressed as

$$\hat{b}_k = \frac{\sum_{q=1}^{Q} (\breve{y}_{k-1}^q)^* \breve{y}_k^q}{\sum_{q=1}^{Q} |\breve{y}_{k-1}^q|^2} = \sum_{q=1}^{Q} (\bar{\breve{y}}_{k-1}^q)^* \boldsymbol{\alpha}_q' \mathbf{y}_k \tag{29}$$

where $\bar{\breve{y}}_{k-1}^q = \breve{y}_{k-1}^q / \sum_{q=1}^{Q} |\breve{y}_{k-1}^q|^2$. The expression (29) can be written in a vector form as

$$\hat{b}_k = \underbrace{\begin{bmatrix} \boldsymbol{\alpha}_1' & \cdots & \boldsymbol{\alpha}_Q' \end{bmatrix}}_{\mathbf{a}' = (\mathrm{vec}(\mathbf{A}))'} \underbrace{\begin{bmatrix} (\bar{\breve{y}}_{k-1}^1)^* \mathbf{y}_k \\ \vdots \\ (\bar{\breve{y}}_{k-1}^Q)^* \mathbf{y}_k \end{bmatrix}}_{\mathbf{u}_k = \bar{\breve{y}}_{k-1}^* \otimes \mathbf{y}_k} = \mathbf{a}' \mathbf{u}_k \tag{30}$$

where the vec$(\mathbf{A})$ operator creates a column vector $\mathbf{a}$ from the matrix $\mathbf{A}$ by stacking columns of $\mathbf{A}$ one below another, $\bar{\breve{y}}_k = \begin{bmatrix} \bar{\breve{y}}_k^1 & \cdots & \bar{\breve{y}}_k^Q \end{bmatrix}^T$, and $\otimes$ denotes the Kronecker product.

To arrive at the weights $\mathbf{a}$ without a priori knowledge of the channel, we use the error

$$e_k = b_k - \hat{b}_k = b_k - \mathbf{a}' \mathbf{u}_k \tag{31}$$

to formulate the MMSE solution for the precombiner coefficients $\mathbf{a}$ which are computed recursively over the carriers as

$$\mathbf{a}_k = \mathbf{a}_{k-1} + \frac{1}{\lambda + \mathbf{u}_k' \mathbf{P}_{k-1} \mathbf{u}_k} \mathbf{P}_{k-1} \mathbf{u}_k e_k^* \tag{32}$$

where λ is the RLS forgetting factor and $\mathbf{P}_k$ is an estimate of the inverse of the covariance matrix which is initially set to $\delta \mathbf{I}_{QM_r}$ with δ being the regularizing factor.

As shown in Fig. 8, the process can continue into the next block, where recursion will evolve in reverse order (from the highest carrier to the lowest) and require fewer (or no) pilots.

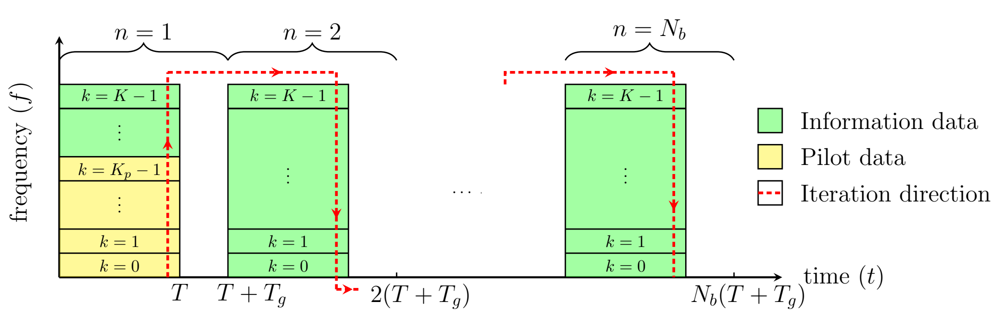

Fig. 8. Progression of the algorithm. Shown is a frame containing N_b OFDM blocks. The algorithm forms the error in (31) using the pilot data symbols $b_k, k \in \mathcal{K}_p^s = \{0, \ldots, K_p - 1\}$ in the training mode, shown in yellow (lighter). Thereafter, it switches into decision-directed mode where the decisions are made on the composite estimate (28). The red dashed line shows the direction of the algorithm's iteration. In the first block, the iteration goes from the lowest carrier to the highest carrier. At the end of the first block, the obtained precombining weights for the highest carrier are used to initialize the precombining weights for the highest carrier of the second block, in which the iteration goes from the highest to the lowest carrier.

In a coherent receiver, the precombined FFT outputs $\check{y}_k^q,\ k = 0, \ldots, K-1$ given by

$$\check{y}_k^q = \boldsymbol{\alpha}_q'(k)\mathbf{y}_k = d_k \underbrace{\boldsymbol{\alpha}_q'(k)\mathbf{H}}_{\check{H}_k^q} + \check{z}_k^q,\ q = 1, \ldots, Q \tag{33}$$

are fed into the PI channel estimator to form the $Q \leq M_r$ estimates $\hat{\check{H}}_k^q$, of the precombined channel coefficients $\check{H}_k^q$. Using a Q-channel configuration instead of the full-complexity configuration reduces the total number of channel estimators by M_r/Q. This feature brings a significant reduction in complexity when the value of Q at which saturation is reached is low. Using MRC, the data symbol estimates are then obtained as

$$\hat{d}_k = \frac{\hat{\check{\mathbf{H}}}_k' \check{\mathbf{y}}_k}{\hat{\check{\mathbf{H}}}_k' \hat{\check{\mathbf{H}}}_k} \tag{34}$$

where $\hat{\check{\mathbf{H}}} = \begin{bmatrix} \hat{\check{H}}_k^1 & \ldots & \hat{\check{H}}_k^Q \end{bmatrix}^T$.

VII. Software Package

The source code available for download here contains functions for frequency offset estimation, coherent partial FFT demodulation, adaptive spatial coherence learning, and path identification channel estimation with multiple receiving elements (spatial diversity). The code is not optimized to achieve low complexity, but is meant to provide a flexible platform for the readers to try the algorithms on their own and re-write the details of their choice. The package provides most of the features described in [1], [2]. Any of the functions in the package may be used for non-commercial purposes. For questions about the implementation, please contact Amir Tadayon.

To run the receiver, one should execute the Matlab script `Simulation.m` which executes the main receiver function `runSimRx.m` on a saved data stored in the file `K1024_CP1_MTPSK_M4.mat`. The parameters of the algorithms, whose implementations are located in the folder `MATLAB-Toolbox`, can be tuned in the function `helperSimMethodConfig.m`. These parameters are then used to configure the receiver algorithms in the function `helperSimParamConfig.m`. The simulation parameters such as bandwidth, number of carriers, frequency of the first carrier and etc as well as some parameters, that enables/disables plotting figures and logging some information as the simulation is running, are set in the function `helperSimParamInit.m` and stored in a Matlab struct named `simParam`. Some useful performance metrics such as mean-squared error (MSE) of data detection, number of symbols in error, and some of estimated parameters such as frequency offset, channel frequency response, as well as transmitted data symbol are stored in a Matlab struct named `Result`.

References

[1] A. Tadayon and M. Stojanovic, “Low-complexity superresolution frequency offset estimation for high data rate acoustic OFDM systems,” *IEEE Journal of Oceanic Engineering*, vol. 44, no. 4, pp. 932–942, Oct. 2019.

[2] ——, “Iterative sparse channel estimation and spatial correlation learning for multichannel acoustic OFDM systems,” *IEEE Journal of Oceanic Engineering*, vol. 44, no. 4, pp. 820–836, Oct. 2019.