Krylov Subspace Algorithms and Circulant Embedding Method for Efficient Wideband Single-Carrier Equalization

Raymond Guan and Thomas Strohmer
Department of Mathematics, University of California,
Davis, CA 95616-8633, USA.
Email: rguan, strohmer@math.ucdavis.edu

Abstract

Wider bandwidth allows higher data rate by transmitting narrower pulses. However this means that the effective channel response is longer and the number of significant taps increases. For single-carrier communication systems this results in higher computational burden at the receiver. We are concerned with single-carrier non-block transmission schemes with receiver oversampling, as they can provide higher spectral efficiency than block transmission schemes in the presence of large delay spreads. We first propose a simple FIR equalizer that is based on the circulant embedding method and analyze its performance by investigating the relationship between solutions of various finite-dimensional models and the original infinite-dimensional problem. We then focus on the conjugate gradient (CG) algorithm as an efficient means for equalization that is specifically well suited for dealing with large-delay-spread channels. We discuss the importance of stopping the iterations for the CG algorithm at the right time in the presence of noise and present several reliable low-cost stopping criteria. It turns out that the CG algorithm equipped with appropriate stopping criteria can outperform MMSE equalizers. Since both the CE and the CG methods can be efficiently implemented via Fast Fourier Transforms, equalization complexity is only in the order of $N \log(N)$ for N data symbols. Several numerical experiments demonstrate the performance of the proposed methods.

I. Introduction

In recent years, driven by military application as well as consumer products, the demand for high-data rate wireless communication systems has been increasing at a dazzling pace. This leads to wider bandwidth communication systems such as ultra wideband system with bandwidth in the order of hundreds megahertz or more [1]. While the use of multiple antennas is known to increase channel capacity, it comes at a significant increase in hardware costs. Thus in practice, the best way to achieve higher data rates in wireless communications is still to increase the bandwidth of the wireless channel since the channel capacity grows linearly with the channel bandwidth. However, simply increasing the channel bandwidth is only half the story – there are problems associated with this approach. For instance, channel state information is harder to obtain for wideband channels and multipath propagation plays an increasingly dominant role which makes equalization a challenging task.

There exist essentially two competing wireless transmission schemes: single-carrier and multi-carrier transmission [2], [3]. A prominent example for a multi-carrier communication system is Orthogonal Frequency Division Multiplexing (OFDM). Comparing the two schemes, OFDM [4] has the advantage of low receiver complexity while achieving ML decoding. The disadvantage of it is paying a higher price for the transmitter with a wide dynamic range power amplifier to handle the peak-to-average power ratio problem that is intrinsic to OFDM. Also, the ML performance is achieved by employing a cyclic prefix (or postfix) which effectively reduces data rate by transmitting redundant information. If reducing data rate is tolerable, single-carrier transmission with zero-padding or cyclic extension [5] may achieve ML performance as well with frequency domain equalization [6] while avoiding the use of expensive power amplifiers. Furthermore, single-carrier systems are less sensitive to channel estimation error and to carrier frequency offset.

This work was partially supported by NSF DMS grant No. 0511461 and NSF VIGRE grant No. DMS-0135345.

In this paper, our interest is in wideband real-time high data rate applications with single-carrier systems. Due to the long delay spread encountered in wideband communication systems, there is a severe penalty in terms of loss of data rate when employing single-carrier block transmission as is done for instance when using guard intervals or cyclic prefix. On the other hand the complexity of equalization can become significant for non-block transmission schemes which raises the question of how to construct numerically efficient equalizers for this case. It is also known that oversampling at the receiver is necessary for the existence of perfect reconstruction FIR equalizers, but again efficient computation becomes the main issue. Therefore we focus on designing efficient equalization algorithms for high-data rate single carrier non-block transmissions with oversampling at the receiver. Note that these methods are naturally applicable to baud rate sampling, block or non-block, transmission as well. The proposed algorithms are sufficiently fast to be used for (near) real-time applications such as video streaming, file or data storage sharing, etc..

We consider the computation of FIR equalizers based on approximating the (inverse of the) channel matrix by a circulant matrix. While this idea of circulant embedding is certainly not new, we discuss the necessary modifications in the presence of oversampling and provide a rigorous analysis between the relation of the standard FIR equalizer and its approximation via circulant embedding. Our theoretical analysis gives valuable insight in the choice of the length of the FIR equalizer with respect to the condition number of the channel. While this choice is not a big deal for channels with short delay spread, it does become a critical issue for large delay spread channels.

We then discuss the use of Krylov subspace techniques, specifically, conjugate gradient algorithms for equalization. The CG algorithm has been proposed for various purposes in wireless communications. E.g. in the downlink of Wideband Code Division Multiple-Access (WCDMA) it is used to compute a bank of FIR filters as pre-filters, in part, for approximating the minimum mean-square error (MMSE) equalizer [7]. An ASIP architecture implementation of a similar CG-FIR pre-filtering scheme for MIMO WCDMA downlink may be found in [8]. The main advantage of a CG approach is to reduce the computational cost of the MMSE equalizer that would otherwise involve a matrix inversion.

For multipath channels with many paths, the MMSE equalizer is superior to the conventional RAKE receiver since the latter can only focus on a few dominating paths due to its huge computational complexity. In [7], simulations have been performed and it is demonstrated that performance may improved by increasing the number of CG iterations. More importantly, it outperforms the traditional Rake receiver.

Another interesting application of CG methods is in the area of HSDPA as a means to implement low-cost MMSE equalizers [9].

However, a hitherto overlooked problem of using the CG algorithm, is that in the presence of noise, convergence of CG is not guaranteed. In fact, if the iterations are not stopped at the right time, the CG algorithm may diverge, leading to inferior results. On the other hand, the CG algorithm has an important—yet in the communication community not well known—regularization property, which manifests itself in the following way: in the initial steps the CG iterates approximate those components of the solution that are associated with the large singular values of the matrix, while the latter iterates pick up those components associated with the small singular values. Since the small singular values are the "culprits" that cause noise amplification, we could obtain a regularized solution that suppresses this detrimental effect by stopping the CG iterations at the right time. One of our main contributions is the introduction of several robust and low-cost stopping criteria tailored to use CG for equalization. We also discuss other practical aspects when using the CG algorithm, such as issues related to efficient numerical implementation as well as various regularization techniques related to CG.

Several numerical experiments for real-world wideband channels demonstrate the performance of the proposed equalization methods. For instance, simulations show that the CG algorithm equipped with appropriate stopping rules can even outperform MMSE equalizers.

II. Problem Formulation and Known Solutions

We review continuous-time and discrete-time single-carrier communication models in this section while establishing the notation that will be used throughout the rest of this paper. Then, we will discuss some well-known linear equalizers and point out some of their properties and problems when using them in practice.

A. Communication Model

To transmit the set of discrete information symbols $\{s_k\}_{k=-\infty}^{\infty}$, it is first converted to a continuous-time signal $c(t)$ by a sum of weighted copies of a finite duration pulse shaping function $p(t)$ via

$$c(t) = \sum_k s_k p(t - kT). \tag{1}$$

Let $h_c(t)$ represent the impulse response of the communication channel. Throughout the paper, exact channel state information at the receiver is assumed. At the receiver, the continuous-time received signal

$$r(t) = (h_c * c)(t) \tag{2}$$

is filtered with a finite duration matched filter $q(t)$ and then sampled to produce a digital signal so that it may further processes digitally with DSP chips to reduce cost.

Mathematically, define $p_k = p(t - kT)$, we are discretizing the continuous-time function

$$b(t) = (r * q)(t) = \sum_k s_k (h_c * p_k * q)(t) \tag{3}$$

to, depending on the sampling rate, produce discrete sequences $b(t_{i,j})$ and

$$h_d(i, j) = (h_c * p_k * q)(t_{ij}) \tag{4}$$

where $t_{i,j}$ is the sampling time that takes on values from the set

$$\{..., -2T/a - jP, -T/a - jP, 0 - jP, T/a - jP, ..., iT/a - jP, ...\}.$$

We have in general three cases: Nyquist sampling (if $a = 1, P = 0$), Q-fold integer oversampling (if $a = 1$, $P = T/Q$ and $j = 0, 1, ..., (Q - 1)$ for some natural number Q) and fractional sampling with a sampling period of P (if $a > 0$ is not an integer, $P = T/(aQ)$ and $j = 0, 1, ..., (Q - 1)$ for some natural number Q). For a more general multirate signal processing discussion, the reader is referred to [10].

B. Algebraic Formulation

Unless stated otherwise, the following notational conventions are used throughout the rest of the paper: (1) capital italic letters such as H denote finite dimensional matrices. (2) Lower case italic letters such as x or h are finite dimensional vectors. (3) For matrices, subscript like x in H_x refers to the x-th row of the matrix H if x is a number, otherwise it is a naming of the matrix, and H^j is the j-th column of H. Similarly, h_i is either the i-th entry in the vector h or an element from the set $\{h_i\}$, it should be clear from the context. (4) Superscript with parenthesis like $H^{(k)}$ (or $x^{(k)}$) refers to different matrices (or vectors) for different integer k. (5) Bold letter such as $\mathbf{H}$ is the infinite dimensional counterparts of H and other conventions such as subscript or superscript, for finite dimensional matrix or vector hold for the infinite dimensional one as well. (6) For random variables in matrix or vector form, bold slanted letters such as $\boldsymbol{n}$ are used.

To develop numerical algorithms, it is often convenient to formulate the problem in terms of algebraic equations. Discretizing $b(t)$ enables us to do just that. For the Nyquist sampling case, we have the bi-infinite dimensional linear system

$$\mathbf{Hx} = \mathbf{b}, \tag{5}$$

where

$$\mathbf{H} = \begin{bmatrix} \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ \ddots & h_L & h_{L-1} & ... & h_1 & 0 & ... & 0 & \ddots \\ \ddots & 0 & h_L & h_{L-1} & ... & h_1 & \ddots & \vdots & \ddots \\ \ddots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \ddots \\ \ddots & 0 & ... & 0 & h_L & h_{L-1} & ... & h_1 & \ddots \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}$$

is a bi-infinite band Toeplitz matrix formed by the length L discrete equivalent channel

$$h = [h_d(1,0), h_d(2,0), ..., h_d(L,0)]^T \tag{6}$$

and

$$\mathbf{x} = [..., s_{-1}, s_0, s_1, ...]^T. \tag{7}$$

Since the above linear system is infinite dimensional, we will truncate it so that it becomes numerically tractable. When the discrete channel is assumed to have only finite number of non-zero taps the truncated linear system still contains all the channel information, however it is rank deficient regardless of how large we make the linear system. In the absence of noise, a sampling rate higher than the Nyquist rate, i.e., with $a > 1$, is required for perfect (FIR) equalization [11]. For practical purposes, in this paper, we consider integer oversampling only. For the case of Q-fold oversampling, the finite channel matrix has block Toeplitz structure with a block size of Q rows:

$$H^{(Q)} = \begin{bmatrix} h_{(L)} & h_{(L-1)} & ... & h_{(1)} & 0 & ... & 0 \\ 0 & h_{(L)} & h_{(L-1)} & ... & h_{(1)} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & ... & 0 & h_{(L)} & h_{(L-1)} & ... & h_{(1)} \end{bmatrix} \tag{8}$$

where

$$h_{(l)} = [h_d(l,0), h(l,1), ..., h(l,Q-1)]^T. \tag{9}$$

For the rest of the paper we assume Q-fold [1] oversampling thus we drop the $H^{(Q)}$ notation and simply use H.

To be more accurate, a Q-fold oversampled single-carrier system that also models noise and interference with a random vector $\boldsymbol{n}$ is

$$Hx + \boldsymbol{n} = b. \tag{10}$$

For simplicity, throughout the rest of the paper, the following assumptions are made: (1) $x_k = 1$ or -1, i.e., we consider BPSK modulation scheme. (2) When the transmitting symbols are considered as random variables, we consider them as independent and identically distributed (i.i.d) with zero mean and equally likely. (3) We assume zero-mean noise with variance σ^2. It is clear that the samples of bandlimited noise are no longer uncorrelated if the receiver oversamples the incoming signal. Thus the noise samples in $\boldsymbol{n}$ are not independent. Therefore in principle to optimize equalizer performance we should utilize the corresponding noise correlation matrix R with entries $R_{k,l} = \mathcal{E}(\langle \boldsymbol{n}, \boldsymbol{m} \rangle)$ where $\boldsymbol{n}, \boldsymbol{m}$ are different noise realizations. However, partly to reduce computational complexity, but mainly to simplify the presentation of the key ideas in this paper, we will base our algorithms on the suboptimal approximation of R by the identity matrix. For the numerical simulations in Section VI, the noise is of course modeled as correlated noise (even though our algorithms do not exploit this fact).

[1] However we note that the results hold true with trivial modifications for any integer oversampling rate.

C. State of the Art Baud-Rate Equalizers

Various types of equalizers are known, notably linear equalizers such as *zero-forcing* (ZF) *equalizer* and *minimum-mean square error* (MMSE) *equalizer* (or *Wiener Filter*) – they are respectively

$$H_{ZF}^{\dagger} = (H^*H)^{-1}H^* \tag{11}$$

and

$$H_{MMSE}^{\dagger} = H^*(HH^* + \sigma^2 I)^{-1}, \tag{12}$$

cf. [12] The ZF equalizer is unbiased and it eliminates the *inter-symbol interference* (ISI) as much as possible while ignoring the additive noise enhancement. If the condition number of the channel matrix H is not small, the ZF solution is highly undesirable. The Wiener filter, on the other hand, balances ISI mitigation with moderate noise amplification so that the total distortion is minimum in the average square-error sense and it is biased.

Any left matrix inverse of H, denoted by $H^{\dagger}$, may be used as an equalizer for the single-carrier system (10). And any row from the left matrix inverse, e.g., the k-th row of $H_{MMSE}^{\dagger}$ written as

$$g_{MMSE}^{(k)} = \left(\left(H_{MMSE}^{\dagger}\right)_k\right)^T, \tag{13}$$

termed an *FIR equalizer* [13], may be used to decode symbols one at a time by taking inner product of the time synchronized received vector and the FIR equalizer. Particularly for wideband wireless single-carrier system, the matter of which FIR equalizer to choose among the rows of $H^{\dagger}$ is relevant; we will come back to this point in Section V-C. Also, for channels with long delay spread, the MMSE (or ZF) solution becomes impractical because of its cubic computational complexity in inverting the matrix $(HH^*) + \sigma^2 I$ (or (H^*H)).

For the Nyquist rate sampling case a computationally efficient sub-optimal FIR equalizer is given by [13]

$$g = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}h}{|\mathcal{F}h|^2 + \sigma^2},\right\} \tag{14}$$

where $\mathcal{F}$ denotes the discrete Fourier transform (efficiently implemented via the FFT algorithm) and h is the baud rate discrete channel impulse response. For good performance, in general, the FFT length is much longer than the length of h, however, to reduce cost, the equalizer g is then truncated.

Non-linear equalizers like decision feedback equalizers (DEFs) are effective and simple to implement when the channel delay spread is short or sparse with few significant taps. However, for channels with many significant taps, error propagation limits the use of them. In terms of performance, the best equalizer is the *maximum-likelihood* (ML) *equalizer* given by

$$\underset{x\in\mathbb{S}}{\operatorname{argmin}} ||b - Hx||, \tag{15}$$

where $\mathbb{S}$ is the set of all possible solutions and $\|.\|$ denotes the Euclidean norm. While recent results show that the expected complexity of ML is not exponential [14], for large delay spread channels ML is nevertheless way too costly or even impossible to implement in practice.

D. Oversampling and SIMO

Note that the basic idea in the solution of (14) is to approximate the baud rate Toeplitz channel matrix H_b by a circulant matrix C_b. In matrix notation, the solution is

$$H_b^{\dagger} = (C_b^*C_b + \sigma^2 I)^{-1}C_b^* \tag{16}$$

where the FIR equalizer g may be found in one of the rows of $H_b^\dagger$ if the FFT length in (14) is the same size as the circulant matrix C_b in (16). However, unlike the matrix ZF and the Wiener Filter of (11) and (12), for baud rate sampling, the simple FFT-based equalizer in (16) and (14) is not directly applicable to the oversampling case since the oversampled channel matrix H is block Toeplitz instead of Toeplitz.

Incidentally, the formulation for the integer oversampling case is the same as for a single-input-multiple-output (SIMO) communication systems [15] with the exception that for the SIMO case the noise is indeed uncorrelated. For the Q-fold oversampled case, the m-th *sub-channel*

$$h^{(m)} = [h_d(1,m), h_d(2,m), ..., h_d(L,m)]^T, \tag{17}$$

where $m = 0, 1, ..., (Q-1)$, is the equivalent m-th receiver discrete channel. Therefore, the results in this paper are applicable directly to SIMO single-carrier communication systems.

III. An Efficient FIR Equalizer Implementation

In this section we present an equalizer based on the idea of circulant embedding. In practice, for reasons such as efficient implementation (last topic of this section) and stability, an FIR equalizer is preferable. However, analysis through matrices is more general as well as more convenient, thus, both solutions are presented.

A. Circulant-Embedding Equalizers

To develop similar efficient FFT-based solutions as (14) or (16), we rely on the following two results:

Lemma III.1: For any matrix $A \in \mathbb{C}^{\rhd\times\ltimes}$ and real number $\alpha > 0$, the minimum mean-square error inverse of A,

$$A^\dagger_{MMSE-I} = A^*(AA^* + \alpha I_m)^{-1} \tag{18}$$

is equivalent to

$$A^\dagger_{MMSE-II} = (A^*A + \alpha I_n)^{-1}A. \tag{19}$$

Proof: Assume $m > n$, we follow the usual convention of singular value decomposition (SVD) of a matrix and write $A = UDV^*$, where $U \in \mathbb{C}^{\rhd\times\ltimes}$ and $V \in \mathbb{C}^{\ltimes}$ is an unitary matrices and $D = \text{diag}(d_1, d_2, \ldots, d_n)$ is such that $d_1 \geq d_2 \geq \ldots \geq d_n \geq 0$. By substituting the SVD of matrix A into (18) we have (19) with the following manipulation:

$$\begin{aligned}
A^\dagger_{MMSE-I} &= A^*(AA^* + \alpha I_m)^{-1} \\
&= VD^*U^*(UDV^*VD^*U^* + \alpha I_m)^{-1} \\
&= VD^*U^*U_m(D_m^2 + \alpha I_m)^{-1}U_m^* \\
&= VD^*(D^2 + \alpha I_n)^{-1}U^* \\
&= V(D^2 + \alpha I_n)^{-1}V^*VD^*U^* \\
&= (VDU^*UDV^* + \alpha I)^{-1}A^* \\
&= (A^*A + \alpha I)^{-1}A
\end{aligned} \tag{20}$$

where $U_m \in \mathbb{C}^{\rhd}$ is an unitary matrix with U in its first n columns and $D_m = \text{diag}(d_1, d_2, \ldots, d_n, 0, \ldots, 0)$. Similar derivation holds for the case when $m < n$. ❏

To distinguish the two different forms of the MMSE solution, we called (18) the matrix MMSE Equalizer Form-I and the alternative expression (19) MMSE Equalizer Form-II. And the following result highlights the key advantage of the second form of the solution.

Lemma III.2: For any $QM \times N$ block Toeplitz matrix H of the form (8) with a block size of Q-rows, where $Q \in \mathbb{N}$ and $N \geq (2L-1)$, the product matrix H^*H has the form

$$H^*H = \begin{bmatrix} * & * & 0 \\ * & T_{H^*H} & * \\ 0 & * & * \end{bmatrix} \tag{21}$$

where T_{H^*H} is a hermitian positive definite (or semi-definite) Toeplitz matrix that may be written as a product of matrices of the form

$$T_{H^*H} = \hat{H}^*\hat{H} \tag{22}$$

where $\hat{H}$ is the truncated matrix of H with the left and right most $L-1$ columns been deleted.

Proof: Since the i-th row j-th column of (H^*H) is the vector product $(H^*)_i H^j$, from the construction of T_{H^*H} in (22), it is clear that the matrix T_{H^*H} shows up in the center of (H^*H). Thus the hermitian and positive definite (or semi-definite) properties are obvious. To show that it is also Toeplitz, first note that by the assumption on the size of $N \geq (2L-1)$, the size of $\hat{H}$ is at least one or more columns, and from the block Toeplitz structure of H, the non-zero entries in the columns of $\hat{H}$ are the entries of the vector

$$[h_d(1,0), h_d(1,1), \ldots, h_d(1,Q-1), h_d(2,0), h_d(2,1), \ldots, h_d(L,Q-1)]^T \tag{23}$$

shifted from one to another by Q entries. Therefore, $(\hat{H}^*)_i\hat{H}^j = (\hat{H}^*)_u\hat{H}^v$ if $i-j = u-v$, i.e., $\hat{H}^*\hat{H}$ is Toeplitz. ❏

Therefore, alternatively, the MMSE equalizer may also be written as

$$H^{\dagger}_{MMSE} = (H^*H + \sigma^2 I)^{-1}H^*. \tag{24}$$

For the case of oversampling, where the channel matrix H has block Toeplitz structure, the solution in this form is more useful since it leads to the efficient CE equalizers. The main difference between the two different forms of the Wiener Filters are in the structure of the product matrices – the matrix (HH^*) is a block Toeplitz matrix with blocks of size of Q by Q , i.e., the size of the oversampling factor, while the matrix (H^*H), as pointed out in lemma III.2, contains a Toeplitz matrix in the middle.

The class of *circulant-embedding* (CE) *equalizers* is based on the idea of approximating not H but (H^*H), or T_{H^*H} rather, by a circulant matrix C. Thus, the main computational cost of equalization is the cost of circulant matrix inversion which is in the order of $N\log(N)$ due to the efficiency of the FFT algorithm. Variations of CE equalizers deriving from approximating the ZF and MMSE equalizers are the *zero-forcing circulant-embedding* (ZFCE) *equalizer*

$$H^{\dagger}_{ZFCE} = C^{-1}H^* \tag{25}$$

and the *regularized circulant-embedding* (RCE) *equalizer*

$$H^{\dagger}_{RCE} = (C + \sigma^2 I)^{-1}H^*. \tag{26}$$

Now, having a finite Toeplitz matrix T_{H^*H} to work with, we are in a good position to discuss the construction of the circulant matrix C. There are various ways to do this, in this paper we consider the *embedding method.* That is, C is constructed first by embedding T_{H^*H} into a larger matrix and then modifying only the lower-left and upper-right corner entries appropriately so that it becomes a circulant matrix [16], [17]. Therefore we may write

$$C = T_{H^*H} + W \tag{27}$$

for some W that depends on T_{H^*H}. Although H is not Toeplitz, T_{H^*H} is and to compute its row or (column) which is needed to construct C we can still use the FFT algorithm. Thus the overall computational complexity of the ZFCE (or the RCE) equalizer remains in the order of $N\log N$.

B. Rates of Convergence of CE Equalizers

Throughout this section, the size of the matrices is explicitly denoted by using subscript notation. Assuming the $M \times N$ channel matrix H_N has full rank, then the product matrix $(H_N^* H_N)$ is invertible and the k-th symbol ZF FIR equalizer may be expressed as

$$g_N = \left(u_k^T \left(H_N^* H_N\right)^{-1} H_N^*\right)^T. \tag{28}$$

It is clear from the above expression that the full-rank assumption is sufficient and necessary for the existence of g_N. To satisfy such an assumption, the channel matrix H_N has to be at least square. For the case of Q-fold oversampling, the parameters M, N and Q are related by the equation

$$M = Q(N - L + 1), \tag{29}$$

and for the matrix H_N to be square or tall, the minimum N is

$$N_{min} = \left\lceil \frac{Q(L-1)}{Q-1} \right\rceil. \tag{30}$$

For the ZF IIR equalizer:

$$\mathbf{g}_{ZF} = \left(\mathbf{u}_0^T \left(\mathbf{H}^* \mathbf{H}\right)^{-1} \mathbf{H}^*\right)^T, \tag{31}$$

where $\mathbf{u}_0$ is the 0-th unit vector, and the necessary and sufficient condition for it to exist is also the invertibility of $(\mathbf{H}^*\mathbf{H})$. In the infinite dimensional case, however, we may check the existence of $\mathbf{g}_{ZF}$ from the finite sub-channels instead, and this is the well-known "no common zeros" condition, i.e., the sub-channel transfer functions are coprime; for the remaining part of this section, we will make such an assumption so that $\mathbf{g}_{ZF}$ exists. Furthermore, $N > N_{min}$ and H_N having full-rank are also assumed.

In practice, we may only approximate the IIR equalizer (31) with an FIR filter. One approach to this is to approximate the infinite model (5) with a finite model (10) and then proceed to compute the finite model solution with the hope that when the finite model converges to the infinite model, the finite model solution (28) will convergence to the infinite model solution (31) as well. However, we observe that, e.g., when the boundary taps of the discrete sub-channels are very weak, the product matrix $(H_N^* H_N)$ may be very ill-conditioned even when $(\mathbf{H}^*\mathbf{H})$ is not. Thus, we propose to compute the FIR equalizer by inverting a more stable matrix instead and redefine the ZF FIR equalizer as

$$g_N = \left(u_k^T \left(T_{H^*H}\right)_N^{-1} H^*\right)^T \tag{32}$$

where $\left(T_{H^*H}\right)_N$ is a $N \times N$ matrix that is defined in (22). This alternative FIR solution approximates more directly the infinite model solution (31). The advantage of it is that the matrix $\left(T_{H^*H}\right)_N$ is at least as well conditioned as the matrix $(\mathbf{H}^*\mathbf{H})$ by Cauchy's interlacing theorem [18].

For $N \in \mathbb{N}$ and $x \in \mathbb{C}^{2N+1}$, define the *embedding operator* E_N of x as

$$E_N x = (\ldots, 0, 0, x_{-N}, x_{-N+1}, \ldots, x_{N-1}, x_N, 0, 0, \ldots)^T, \tag{33}$$

and we denote $\mathbf{x}_N := E_N x$ and identify the image of E_N with the $2N+1$-dimensional space $\mathbb{C}^{2N+1}$. We also define the *orthogonal projection* $P_N : \ell^2(\mathbb{Z}) \to \mathbb{C}^{2N+1}$ of y by

$$P_N \mathbf{y} = [y_{-N}, \cdots, y_N]^T. \tag{34}$$

For convenience when comparing to the IIR solution, let us also define

$$\mathbf{g}_N = E_N g_N. \tag{35}$$

While in the absence of noise (and assuming we can compute the IIR equalizer) both $\mathbf{g}_{ZF}$ and $\mathbf{g}_N$ give the same result, i.e., the exact solution, in the case of noise it is clear that $\mathbf{g}_{ZF}$ will in general give better results. The question is now: how are $\mathbf{g}_{ZF}$ and g_N related? And how fast does g_N converge to $\mathbf{g}_{ZF}$ when we increase N (if it converges at all), in other words how fast does the FIR equalizer approach the performance of the IIR equalizer? We proceed to give an answer; again, Q-fold oversampling is assumed but the following results can be extended to any oversampling rate.

Lemma III.3: Let $\mathbf{H}$ be as in (5) and assume that the sub-channel transfer functions are coprime. Let κ denote the condition number of $\mathbf{H}$. Furthermore, let

$$P_N\mathbf{H}^*\mathbf{H}P_N x_N = P_N\mathbf{H}^*\mathbf{u}_0. \tag{36}$$

Then the truncated-system vector inverse x_N, or $\mathbf{x}_N = E_N x_N$ rather, converges to the bi-infinite-system inverse $\mathbf{x}$ as $N \to \infty$ and

$$\|\mathbf{x} - \mathbf{x}_N\| \leq C\Big(\frac{\kappa - 1}{\kappa + 1}\Big)^{\frac{N}{2(L-1)}}, \tag{37}$$

where the constant C is independent of N.

Proof: Clearly, $\mathbf{H}^*\mathbf{H}$ and $P_N\mathbf{H}^*\mathbf{H}P_N$ are hermitian positive-definite Toeplitz matrices. Note that the entries of $\mathbf{H}^*\mathbf{H}$ (and thus of $P_N\mathbf{H}^*\mathbf{H}P_N$) can be computed exactly, since $\mathbf{H}$ is a band matrix. Denoting $S := \mathbf{H}^*\mathbf{H}$, $S_N := P_N\mathbf{H}^*\mathbf{H}P_N$, $g = \mathbf{H}^*u_0$ and $g_N = P_N\mathbf{H}^*u_0$, it is easy to see that the setup is now the same as in Theorem 3(ii) in [19] (with some obvious changes in the notation, since the aforementioned theorem has been derived and proved for a different problem). Hence the estimate (37) follows now easily by applying the same proof as for Theorem 3(ii) in [19]. ❏

Theorem III.4: Let $\mathbf{H}^T\mathbf{g}_{ZF} = \mathbf{u}_0$ be given where $\mathbf{H}$ is as in (5) and assume that the sub-channel transfer functions are coprime. Let κ denote the condition number of $\mathbf{H}$. Furthermore, let g_N as define in (32). Then the FIR equalizer g_N converges to the IIR equalizer $\mathbf{g}_{ZF}$ as $N \to \infty$ and

$$\|\mathbf{g}_{ZF} - \mathbf{g}_N\| \leq C\Big(\frac{\kappa - 1}{\kappa + 1}\Big)^{\frac{N}{2(L-1)}}, \tag{38}$$

where the constant C is independent of N.

Proof: Define

$$\mathbf{s}_0 = \mathbf{u}_0^T\left(\mathbf{H}^*\mathbf{H}\right)^{-1}, \tag{39}$$

$$\mathbf{s}_N = E_N\left(u_k^T\left(\left(T_{H^*H}\right)_N\right)^{-1}\right) \tag{40}$$

and

$$\tilde{\mathbf{g}}_N = E_N\left(P_N\left(\mathbf{u}_0^T\left(\mathbf{H}^*\mathbf{H}\right)^{-1}\right)H_N^*\right)^T. \tag{41}$$

Then, it is true that

$$\begin{aligned}\|\mathbf{g}_{ZF} - \mathbf{g}_N\| &\leq \|\mathbf{g}_{ZF} - \tilde{\mathbf{g}}_N\| + \|\tilde{\mathbf{g}}_N - \mathbf{g}_N\| \\ &\leq \|\mathbf{s}_0 - E_N P_N \mathbf{s}_0\|\|\mathbf{H}\| + \|\mathbf{s}_0 - \mathbf{s}_N\|\|H_N\|.\end{aligned} \tag{42}$$

The first inequality is straightforward while the second one follows from the assumption that $\|\mathbf{H}\|$ is finite. Since $\left(\mathbf{H}^*\mathbf{H}\right)^{-1}$ is hermitian, Lemma III.3 applies to row vector linear systems as well as column ones, thus $\|\mathbf{s}_0 - E_N P_N \mathbf{s}_0\|$ goes to zero, as N goes to infinity, with the same exponential decaying rate. By Lemma III.3 and the fact that $\left(T_{H^*H}\right)_N$ is hermitian, $\|\mathbf{s}_0 - \mathbf{s}_N\|$ also converges to zero with the desired decaying rate. ❏

Although we have concentrated on the ZF equalizer, the results are trivially extended to the MMSE case. And having the ZF equalizer result from above, we are now in a good position to give a convergence result for the ZF-CE equalizer.

Theorem III.5: Let $\mathbf{H}^T\mathbf{g}_{ZF} = \mathbf{u}_0$ be given where $\mathbf{H}$ is as in (5) and assume that the sub-channel transfer functions are coprime. Define the embedded ZF-CE equalizer

$$\mathbf{g}_N^{CE} = E_N g_N^{CE}. \tag{43}$$

where g_N^{CE} is the FIR-ZFCE equalizer which is the middle row of $H_{ZFCE}^{\dagger} = C_N^{-1} H_N^*$. Then, for $N > 3L$

$$\|\mathbf{g}_{ZF} - \mathbf{g}_N^{CE}\| \to 0 \qquad \text{as } N \to \infty. \tag{44}$$

Moreover,

$$\|\mathbf{g}_{ZF} - \mathbf{g}_N^{CE}\| \le C_1 \Big(\frac{\kappa - 1}{\kappa + 1}\Big)^{\frac{N}{2(L-1)}}, \tag{45}$$

and

$$\|\mathbf{g}_N - \mathbf{g}_N^{CE}\| \le C_2 \Big(\frac{\kappa - 1}{\kappa + 1}\Big)^{\frac{N}{2(L-1)}}, \tag{46}$$

where κ is the condition number of $\mathbf{H}$ and C_1, C_2 are constants independent of N.

Proof: There holds

$$\|\mathbf{g}_{ZF} - \mathbf{g}_N^{CE}\| \le \|\mathbf{g}_{ZF} - \mathbf{g}_N\| + \|\mathbf{g}_N - \mathbf{g}_N^{CE}\|. \tag{47}$$

The first term on the right hand side of (47) goes to zero by Theorem III.4. the second term goes to zero by Theorem 4.3 in [20]. The error bound (45) follows also from Theorem 4.3 in [20], and (46) follows from combining (38) with (47) and (45). ❏

C. Length of FIR equalizers

The last point to be addressed in this section is the length of FIR equalizers – a key parameter in the FIR equalizer design. Usually, this is a tradeoff between equalization performance and cost – the longer the FIR equalizer, the larger the costs and the better the performance and vice versa.

Given an M-tap FIR equalizer, g_M, for the k-th symbol of x in the system in (10), the MSE as a function of the equalizer length is

$$\text{MSE}(M) = ||g_M^T H||^2 - 2\text{Re}\left\{(g_M^T H)_k\right\} + 1 + \sigma^2 ||g_M||^2 \tag{48}$$

which may be used to predict the FIR equalizer performance and thus to decide the FIR equalizer length.

An example of a MMSE-FIR equalizer's MSE versus its length of a two-fold oversampled system is demonstrated in Fig. 1 below. The sub-channels are 64 taps long and the SNR is 12 dB. Clearly, the performance of the equalizer degrades significantly when it is too short. The minimum N in this case is $N_{min} = 126$ and the corresponding channel matrix size is 126×126. Notice the three different sections of the curve of different characteristics: First, as the equalizer length increases up to the value of N_{min}, the negative slope of the MSE curve is the largest in magnitude, i.e., greatest improvement in performance as the length of the FIR equalizer increases; since the plot is in semilog scale, the decrease in MSE is actually exponentially fast. For the second (middle) section of the curve, the length of the equalizer increases from 128-tap to about twice that value, the slope is also decreasing with an approximate constant slope – this is expected from theorem (III.4). When the length of the FIR equalizer is beyond $2N_{min}$ or so, the MSE is so small that noise dominates – as the last term in (48) approaches $\sigma^2||\mathbf{g}_{MMSE}||^2$, the curve flattens out and will never actually reach zero.

We also see from the convergence results in Section III that the condition number of the channel has a crucial role on the length of the FIR equalizer. In a nutshell, the better the condition number of the channel, the shorter the FIR equalizer can be. If the channel is ill-conditioned the length of the FIR equalizer will need to quite large to achieve reasonable performance. Clearly, these issues become more pronounced for large delay spread channels.

For the remaining part of the paper, the length of the FIR equalizer, i.e., the size of the finite linear system, is assumed to be predetermined and we will drop the subscript notation, which indicates the size of vectors or matrices, for convenience.

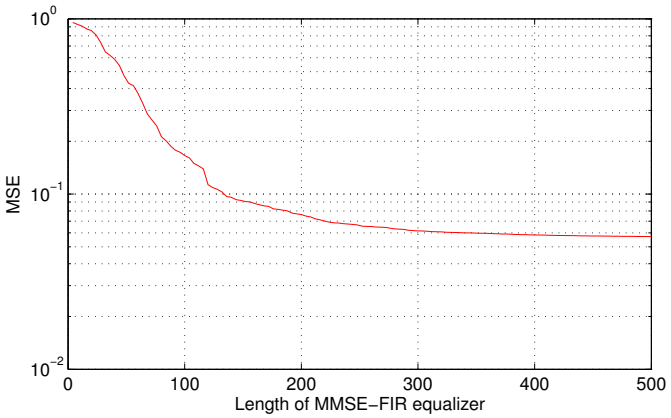

Fig. 1. MSE vs length of the MMSE-FIR equalizer (sub-channel length is 64 taps).

IV. Krylov Subspace Methods in Theory

Solving noisy linear systems is a well established topic in the theory of regularization. The methods may be categorized into direct and iterative schemes. Direct methods cost as much as inverting a matrix so we will only focus on efficient iterative methods that are fast enough to be used in real time. In particular, we will consider Krylov subspace methods based on the conjugate-gradient (CG) algorithm [21].

A. The CG Algorithm

There are various algorithms that are considered as Krylov subspace methods; the first of these is the by now classical CG algorithm, cf. [22] for a detailed discussion of CG. Due to its many nice properties, the CG algorithm is also a natural candidate as an efficient equalization method for single-carrier communication systems. While the CG method has become a standard tool in numerical analysis[2] its practical use in the context of equalization is not straightforward at all and requires some careful modifications and adaptations. And this is the topic for most of the rest of this paper.

For an $N \times N$ hermitian positive definite linear system

$$Ax = y \tag{49}$$

the CG algorithm [18], [23] is merely five lines of code:

$\alpha^{(k)} = \langle r^{(k-1)}, r^{(k-1)} \rangle / \langle p^{(k-1)}, Ap^{(k-1)} \rangle$
$x^{(k)} = x^{(k-1)} + \alpha^{(k)} p^{(k-1)}$
$r^{(k)} = r^{(k-1)} - \alpha^{(k)} Ap^{(k-1)}$
$\beta^{(k)} = \langle r^{(k)}, r^{(k)} \rangle / \langle r^{(k-1)}, r^{(k-1)} \rangle$
$p^{(k)} = r^{(k)} + \beta^{(k)} p^{(k-1)}$

where $x^{(k)}$ is the k-th iterated solution. Most often, the algorithm initializes with

$$x^{(0)} = 0, r^{(0)} = b, \text{ and } p^{(0)} = r^{(0)}.$$

The computational complexity of the algorithm is determined by a matrix vector multiplication in each iteration. When the matrix is not hermitian positive definite, as is the case in (5), the CG algorithm can be applied to the associated normal equations and there are variations of CG like CGNE (CG algorithm applied to normal equations) that avoid the explicit computation of the matrix product H^*H; instead it only costs another matrix vector multiplication with H^*y for some vector y in each

[2]Obviously not aware of the CG method and its many variations, this technique was partially "rediscovered" by Goldstein et al. in the context of Wiener filtering, who called the method "Multistage Wiener filtering". But their paper does not provide new insights into the theory of the method or its practical use for equalization that would be of relevance to our work.

iteration[22]. Throughout the rest of this paper, we will refer both CG and CGNE algorithm as CG algorithm; it should be clear from the corresponding linear systems to which one it refers to.

B. Stopping Criteria

The CG method, when applied to a hermitian positive definite linear system, produces a sequence of iterates $x^{(k)}$, $k = 1, 2, \ldots$, which converge monotonically to the true solution in the noise-free case in a finite number of iterations. It is widely used today due to its nice convergence properties [23], [22]. If the eigenvalues of matrix A are clustered or bounded away from zero, it converges fast. Furthermore, the error measure in matrix norm is non-increasing in the noise-free case. Therefore, the residual error which is defined as

$$r^{(k)} = b - Hx^{(k)} \tag{50}$$

also has the monotone convergence property in the 2-norm. However the convergence properties are more delicate in the presence of noise. As mentioned, in the noise-free case CG produces a sequence $x^{(k)}$ that converges to the true solution x. Unfortunately, in the presence of noise the monotone convergence of this sequence to the true solution is no longer guaranteed, this fact is also true for CGNE [22]. The iterates $x^{(k)}$ may first converge, but later diverge from the true solution. A rule is needed to determine which one in the sequence of iterates is the best choice. Furthermore, the evaluation of such a rule must also be efficient so that it may be applied in real time applications. These are often overlooked aspects when the CG method is proposed for various applications.

One such efficient stopping rule is the so-called *discrepancy principle* [24], [22]. This principle depends on a parameter ϵ, and the solution index k is the smallest integer such that

$$||r^{(k)}||_2 \leq \epsilon \tag{51}$$

is satisfied. The advantage of this rule is that it is simple to compute. However, it requires to chose a parameter ϵ which should be a function of the noise power for example. We found that ϵ as a simple linear function of noise power is not a robust SC to use across a wide range of signal-to-noise ratio (SNR) values. Figure 2 demonstrates the difficulty in using this principle as well as the importance of a good stopping rule.

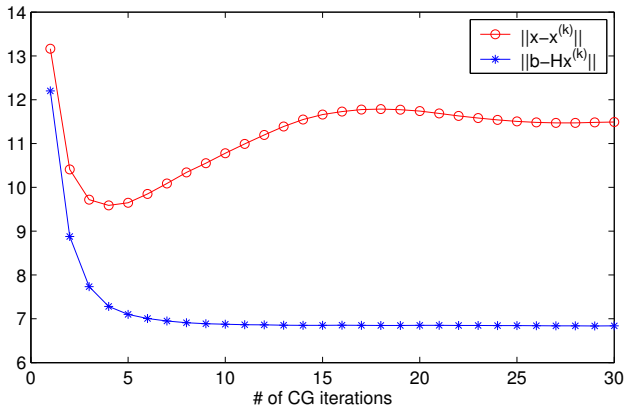

Fig. 2. Actual and residual error.

Another well-known stopping criterion in the theory of regularization is the L-curve [25]. The name comes from the shape of the curve of $\log(||Bx^{(k)}||)$ versus $\log(||r^{(k)}||)$ that resembles the letter L. The matrix B models known properties such as the first or second derivative of the solution. The best solution, according to this rule, is the one corresponding to the one at the corner of the L-curve. For the application of single-carrier communication, e.g., see Figure 3, we found that this rule is also not reliable.

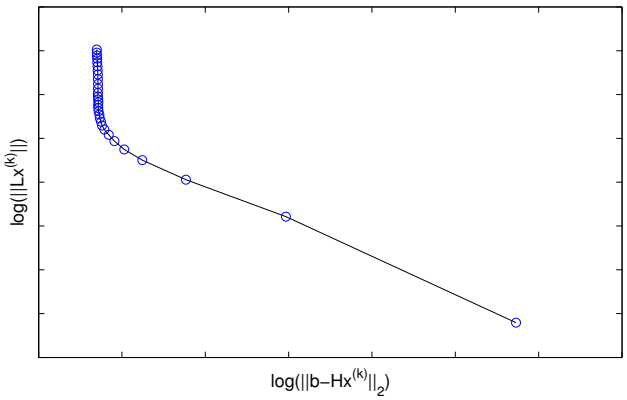

Fig. 3. The L-curce.

C. Two Different Uses of the CG Algorithm

Before discussing optimal stopping criteria for the CG algorithm, we want to point out that by properly formulating the problems, the CG algorithm can be used to compute both the matrix system solution and the FIR solution and these different solutions require different SCs.

Applying the CG algorithm in the straightforward manner, we may use it to solve for x in the noisy linear system (10) directly. On the other hand, we may first compute an FIR equalizer g and then find the desired solution via decoding ($g^T b$). The latter approach motivates the use of the CG algorithm to solve the following linear equation

$$H^T g = u^{(k)}, \tag{52}$$

where $u^{(k)}$ is a vector consisting only of zeros except for the k-th entry, which is equal to 1.

D. Stopping Criteria for the CG Algorithm - Theoretical Considerations

Given full knowledge of the channel information H, for any FIR equalizer g, the BER may be analyzed. Suppose, without lost of generality, x_1 is the desired decoding symbol. Define a vector $d = g^T H$ and assuming $d_1 \neq 0$ otherwise, the BER for x_1 is 0.5. By normalizing the vector g as $g^{(1)} = (1/d_1)g$, we may expand

$$\left(g^{(1)}\right)^T b = x_1 + (1/d_1)\sum_{k=2}^{N} d_k x_k + \sum_{k=1}^{M} g_k^{(1)} \mathbf{n}_k. \tag{53}$$

Thus, the BER may be computed by analyzing the total noise distribution that consists of ISI due to the neighbor symbols x_2, ... x_N (the second term in the above equation) and the equalized noise (the third term). The interference noise may be modeled with a sum of Bernoulli distributions $\boldsymbol{x}_2$, ..., $\boldsymbol{x}_N$ that have zero-mean and variances $|d_2/d_1|^2$, ..., $|d_{N-1}/d_1|^2$. The equalized noise is a sum of normal distribution with zero mean and variance $(|g_1|\sigma)^2$, ... $(|g_M|\sigma)^2$, thus it is still normally distributed with zero mean and variance $(||g||\sigma)^2$. The *total noise* is then modeled as

$$\boldsymbol{n}^{Total} = \sum_{k=2}^{N} \boldsymbol{x}_k + ||g||\sigma\overline{\boldsymbol{n}}, \tag{54}$$

where $\overline{\boldsymbol{n}}$ is a normally distributed random variable with zero-mean and unit variance.

Therefore, for the CG algorithm when it is used to compute FIR equalizers as in (52), an optimal SC is possible, at least in theory. Since for the i-th CG solution, by knowing the corresponding noise distribution function $f_{\boldsymbol{n}_i^{Total}}$ of $\boldsymbol{n}_i^{Total}$, the BER can be computed by integrating the distribution

function, and by minimizing the BERs with respect to the number of CG iterations, optimal BER-performance CG-FIR equalizer can be obtained. More generally, we have the following result.

Theorem IV.1: For BPSK constellation symbols, the best FIR equalizer g_i from the sequence of FIR equalizers $\{g_i\}_{i=1}^N$ for the j-th symbol, with respect to BER, is the one with index i equal to

$$\min_{i\in[1,2,\ldots,N]} \int_1^\infty f_{\boldsymbol{n}_i}(n)dn \tag{55}$$

where $f_{\boldsymbol{n}_i}$ is the i-th total noise distribution function corresponding to the sequence $\{g_i\}_{i=1}^N$.
Proof: The proof is straightforward since

$$\int_1^\infty f_{\boldsymbol{n}_i}(n)dn \tag{56}$$

is simply the BER for the i-th FIR equalizer. ❑

In practice, however, the computational complexity of the BER is in the order of 2^N), due to the summation of Bernoulli distributions in the ISI term. For large N, the exponential complexity renders the optimal SC impractical. A suboptimal SC based on second order statistics, discussed in the next section, is much cheaper to compute and thus may be used in practice.

For direct computation of x by the CG algorithm, from the ML solution, it is reasonable to minimize the metric

$$||b - Hx^{(k)}||^2 \tag{57}$$

as a SC. In practice however, the above expression requires modification, which is also discussed in the next section.

V. Subspace Methods in Practice

Applying the CG algorithm in practice requires some additional consideration such as finding a low cost SC with acceptable performance and implementing the same algorithm more efficiently. These are the topics of this section.

A. MMSE-SC for FIR Equalizers

For the application of wireless communications, we know that the entries of the solution x belong to a finite alphabet set and we should take advantage of this information when developing SCs. Now, consider the case that the entries of $\boldsymbol{x}$ are uncorrelated zero-mean unit energy random variables with Bernoulli distribution again. Assume the transmitted symbols and the receiver noise are uncorrelated as well, i.e.,

$$\mathcal{E}\{\boldsymbol{x}_i \boldsymbol{n}_j\} = 0, \forall\, i, j \in \mathbb{Z}. \tag{58}$$

When the CG algorithm is applied to the linear system in (52), a sequence of approximate FIR equalizers, $\{g^{(k)}\}_{k\in\mathbb{N}}$, is produced. For each of these solutions, say $g^{(n)}$, written as G_n^T for notational clarity, we may analyze them, similar to (53), by separating the signal and noise components as

$$G_n b = (G_n H)_k \boldsymbol{x}_k + \sum_{i\neq k} (G_n H)_i \boldsymbol{x}_i + G_n \boldsymbol{n}. \tag{59}$$

By formulation, we know $(G_n H)_k$ is supposed to converge to 1. We may initialize $g^{(0)}$ to be the matched filter, i.e., the conjugate of the k-th row of H^T. Thus, it is reasonable to make the following approximation

$$(G_n H)_k \approx 1. \tag{60}$$

Therefore, minimizing the (total) noise power may be used as a SC. This suggests the following noise power metric

$$\begin{aligned} P^{(1)}_{FIR-SC} &= \mathcal{E}\left\{\left|\left|\sum_{i\neq k}(G_nH)_i\mathbf{x}_i + G_n\mathbf{n}\right|\right|^2\right\} \\ &= ||G_nH||^2 - |(G_nH)_k|^2 + (||G_n||\sigma)^2. \end{aligned} \tag{61}$$

By the approximation in (60) and the definition of the residual, (61) may be approximated by

$$P^{(2)}_{FIR-SC} = ||u^{(k)} - r^{(n)}||^2 + (||G_n||\sigma)^2 - 1. \tag{62}$$

Computationally, this may still be further simplified since the value $||r^{(n)}||^2$ is known in each iteration and $u^{(k)}$ is non zero only at the k-th entry which is one. An efficient SC is then

$$P^{(3)}_{FIR-SC} = ||r^{(n)}||^2 + (||G_n||\sigma)^2 \tag{63}$$

since both the residual $r^{(n)}$ and the FIR equalizer G_n are computed in each iteration by default.

Now, consider the case that the approximation in (60) is not good due to for example taking the initial approximation to be the zero vector, which is often done in practice so that the residual converges monotonically. In this case, we want to maximize signal minus noise power. This leads to another SC rule

$$\begin{aligned} P^{(4)}_{FIR-SC} &= \mathcal{E}\left\{|(G_nH)_k\mathbf{x}_k|^2\right\} \\ &\quad - \mathcal{E}\left\{\left|\left|\sum_{i\neq k}(G_nH)_i\mathbf{x}_i + G_n\mathbf{n}\right|\right|^2\right\} \\ &= 2|(G_nH)_k|^2 - ||G_nH||^2 - (||G_n||\sigma)^2. \end{aligned} \tag{64}$$

The above SCs are different due to the quality of approximation of $(G_nH)_k$ by one. When normalizing the FIR equalizer G_n such that

$$(G_nH)_k = 1, \tag{65}$$

the *unbiased FIR equalizer condition*, we found that the expected error of the k-th entry of x is

$$\mathcal{E}\left\{|\mathbf{x}_k - \tilde{\mathbf{x}}_k|^2\right\} = (||G_nH||^2 - 1) + (||G_n||\sigma)^2. \tag{66}$$

The proof is a straightforward computation.

Note that, SCs (61) and (64) will give the same solution as SC (66) when (65) is true. Therefore, if the SNR is known, the above mean-square error (MSE) expression may be used as a practical SC. In fact, we found that it is the best SCs among the ones presented in this paper for the CG algorithm for computing FIR equalizers. Such SC actually determines the MSEs from a set of FIR equalizers that are computed by the CG algorithm, thus it is the *minimum mean-square error SC* or MMSE-SC.

From the two terms in (66) we may see that the "best" equalizer, in the presence of noise is the one that balances between equalizing the channel (the first term) and not amplifying the noise too much (the last term). The MMSE-SC rule is only suboptimal since it is possible to have two different (total) noise distributions where the one corresponding to the smaller BER may actually have a larger variance.

One last point to be addressed to complete the discussion on FIR equalizer computation with the CG algorithm is that which symbol (entry) in x is the best to be equalized. That is, what value should we choose for k in $u^{(k)}$ from equation (52). The answer to this question is given in the next subsection since it is related to the (not surprising) observation that the BER for the symbols in the middle entries of x is much better than the ones on the two boundaries.

B. Boundary Effects

In practice, due to the finite model approximation to the infinite model, the truncation of the finite model leads to unequal MSE for different entries in the solution vector x. The full analysis of this BER behavior includes the effect of the noise and it is too involved. For this reason, we will only analyze the loss of signal energy due to truncation in this paper. In the absence of noise, let us consider any one particular transmission of the signal x. In the finite model formulation, cf. eq. (8), the energy in the first transmitted symbol x_1 is spread out by the channel by the taps $h_L^{(1)}$ and $h_L^{(2)}$ and arrives at the receiver, captured by the samples b_1 and b_2. To capture the maximum energy of this symbol, the matched filter $g_{(1)} = [h_L^{(1)}, h_L^{(2)}, 0, \cdots]^T$ is applied, which gives

$$g_{(1)}^T b = \left(|h_L^{(1)}|^2 + |h_L^{(2)}|^2 \right) x_1 + \sum_{k=2}^{L} f_k x_k. \tag{67}$$

Similarly, the analysis for the symbol, w.l.o.g., x_L is

$$g_{(L)}^T b = \left(||h^{(1)}||^2 + ||h^{(2)}||^2 \right) x_L + \sum_{k \neq L} \tilde{f}_k x_k. \tag{68}$$

The first terms in the right hand side of equations (67) and (68) represent the equalized signal while the second terms are interference noises due to neighbor symbols which we are ignoring. Therefore, we see that unless the energy of each sub-channel is well concentrated in the last taps, the equalized signal energy for the entries in the middle of the vector x, e.g., x_L is larger than the boundary entries such as x_1. Again, the MSE of (48) may be used to demonstrate this idea; Figure 4 below shows the MSE as function of different FIR equalizers from the matrix MMSE solution (24). The SNR is 12 dB.

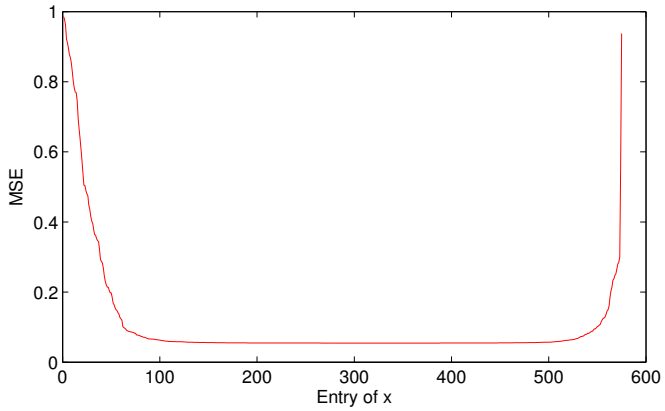

Fig. 4. MSE of different FIR equalizers from the rows of MMSE matrix equalizer (sub-channel length is 64 taps).

Now, we are in a better position to answer the question from the last sub-section – which symbol in x should we equalize? For the matched filter case, we see that it is possible to extract the maximum signal energy for x_L, x_{L+1}, ..., x_{N-L+1}. Thus, equalizing any one of those symbols is equally good. Taking noise and interference into consideration, the matter is not so straightforward. The MSE expression in equation (66) offers an elegant but expensive solution if H is large. For a general guideline, we recommend equalizing exactly the middle entries, though, this is not always the best choice.

C. Matrix Solution

As mentioned in section IV-D, minimizing the ML metric (57) may be used as a SC for the CG algorithm when computing x directly. However, from the last subsection we see that the boundary effect due to truncation of the bi-infinite model, non-equal MSEs in the matrix solutions are unavoidable.

Therefore, it is a good idea to modify the ML metric SC in (57) by truncating the boundary terms. Also, it is a good idea to used the decoded symbols instead of the equalized solution in the ML metric so that the chosen solution is most likely the true solution among the ones from the set of CG iterates. A better stopping rule is therefore the minimization, with respect to k, of the metric

$$||b_t - H_t x_d^{(k)}||^2 \tag{69}$$

where $x_d^{(k)}$ is the k-th decoded and boundary-truncated CG solution of $x^{(k)}$ and H_t and b_t are the appropriate truncated versions of H and b.

Also due to the fact of non-equal MSEs again, the symbols in the boundary of the solutions are not as reliable as the ones in the middle, thus they should be neglected. Therefore, when equalizing the next block of symbols, the next block's head should be overlapped with the tail of the previous block of symbols. If the truncation length in the head and the tail of $x_{(k)}$ is L_h and L_t respectively, then the length of overlapping symbols is $(L_h + L_t)$.

D. Regularized CG Algorithm

Applying the CG algorithm to the noisy linear system (10), with enough iterations, the algorithm will produce the ZF solution. However, this is not a good solution due to the amplification of noise. There are two different ways to deal with the noise enhancement problem and produce a regularized solution similar to the MMSE equalizer.

One way to do this is stopping the CG algorithm earlier with a good stopping rule. For real time applications, the process of solution selection should be done automatically. And to avoid unnecessary iterations we would like to stop the algorithm as soon as the "best" solution is produced. We found that minimizing (69) achieves this purpose. Note that this approach of applying the CG algorithm to an un-regularized system, as opposed to the regularized system that we will be discussed next, relies on the intrinsic regularization effect of the CG algorithm which is still not fully understood when it is used in presence of noise [25].

A "safer" way to avoid too much noise amplification, however, is by explicitly regularizing the linear system with a regularization parameter η [26], i.e., we are applying the CG algorithm to the regularized linear system

$$(H^*H + \eta^2 I)x = H^*b. \tag{70}$$

When η is chosen to be equal to the noise power σ instead of zero, the algorithm converges to the MMSE solution instead of to the ZF solution.

E. Efficient Implementation of the CG Algorithm

For the explicitly regularized system (70), since the noise power for each finite observation is not known, the optimal regularization parameter η needs to be determined (or estimated). This is the parameter selection problem and the traditional approach is actually to compute the solution of various amounts of regularization say from the set $\{\eta_1, \eta_2, \ldots, \eta_s\}$ by brute force and then select a posteriori the best solution from the corresponding solution set $\{x_1, x_2, \ldots, x_s\}$ of different amounts of regularization. This type of brute force approach, is s times more expensive than regularizing the system by simply choosing a priori $\eta = \sigma$.

We find that a more efficient approach to this parameter selection problem for single-carrier equalization is, in *each* CG iteration, to select a best solution from the set of solutions $\{x_1^{(k)}, x_2^{(k)}, \ldots, x_s^{(k)}\}$ that corresponds to different amount of regularization. As for the next CG iteration, the direction of decent as well as the residual (which is used to compute the next step size) is taken to be the one with the amount of regularization η_i corresponding the best solution in the current CG iterate.

The advantage of this *regularized CG-iteration parameter selection method*, as opposed to the traditional brute force parameter selection approach, is its better efficiency – the efficiency comes in two

ways: (1) numerical efficiency and (2) adaptivity. (1) Concerning numerical efficiency we observe the following: When computing the solutions x_i^k's that correspond to different η_i's in the k-th CG iteration, the main computational cost, due to a variant of the regularized CG algorithm which can be found in [27], is still only *one* matrix vector multiplication Hu and *one* matrix vector multiplication H^*v for some vector u and v. In the $(k+1)$-th iteration, only the direction corresponding to the previous best iterate is searched (but again with respect to a range of regularization parameters $\eta_i, i = 1, \ldots, s$). Thus the computational complexity of this regularized CG parameter selection method is of the same order as for the standard CG method and not s times more expensive as the brute force approach. (2) Concerning adaptivity, we note that by selecting the best regularization parameter in each iteration we allow the amount of regularization to change during iterations, and thus due to this adaptivity improve the performance.

Now, notice the similarity in the two different uses of the CG (or CGNE) algorithm, section IV-C, they all require the matrix vector multiplications Hu and H^*v for some block Toeplitz channel matrix H and some vector u and v. It is well-known that these may be computed efficiently with the FFT algorithm [16]. Thus, the FFT algorithm reduces the over all CG algorithm computational complexity to $N\log(N)$, which is a still low even when N is large.

Alternatively, it is well known that preconditioning [16] may be used in conjunction with the CG algorithm to speed up the rate of convergence in order to reduce the required number of iterations. However, we observed from simulations that the number of iterations (which is partially determined by a SC) is rather small, especially in the low SNR region. And since preconditioning requires additional computations, it does not offer complexity reduction in our case, unless perhaps for the case of a channel with deep fades under high SNR conditions.

VI. Simulation Results

The BER performance of various equalizers discussed in this paper have been simulated and the results are shown in Fig. 5 to Fig. 8. These results were simulated with an 8-tap and a 64-tap long sub-channel and the channel data were constructed from measurement data of real channels – courtesy of Intel Corporation. The two 8-tap sub-channels are the odd and even samples of an 16-tap short channel impulse response and the two 64-tap sub-channels are the odd and even samples of a long 512-tap channel impulse response that was down-sampled by 4 first. The measurements were conducted inside an office building with a network analyzer. The environment is an office space ($40m \times 60m$) with many cubicles. Measurements were conducted at several locations, at off-peak hours to ensure channel stationarity, and span a bandwidth of 2-8 GHz with 3.75 MHz frequency resolution.

The FIR equalizer BER curves with the MMSE, RCE and the CG solutions are compared and shown in Fig. 5 and Fig. 7 for the two channels of different length, while the BER performance for the matrix solutions, MMSE and CG-direct methods, are shown in Fig. 6 and Fig. 8 respectively for the 8 and 64-tap length sub-channels. Comparing the FIR equalizer results, the MMSE and RCE BER curves are overlapped due to very good approximation of the CE method. The CG-FIR performance is worse than the MMSE-FIR and this is expected since the CG-FIR equalizer is a low-cost sub-optimal estimated solution to the MMSE-FIR equalizer.

Comparing the matrix solutions, it is interesting to see that the CG-direct solution outperforms the MMSE equalizer. This is not too surprising since the MMSE equalizer does not make use all the information that is available – it uses only the channel state and noise power information. In contrast, when solving the variable x directly, the CG algorithm make uses all the information available, including the received vector b, hence it is able to outperform the MMSE solution. We stress again that for the CG-direct method being able to achieve such a good performance, it is crucial to have a reliable stopping criterion.

From our simulation with various channels, we observed that the number of CG iterations needed to compute the FIR equalizer or to compute the solution x directly is small – two to ten iterations

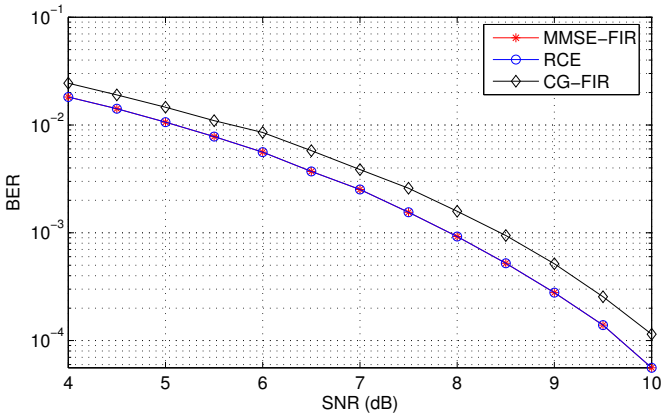

Fig. 5. FIR equalizers performance comparison of an 8-tap sub-channel.

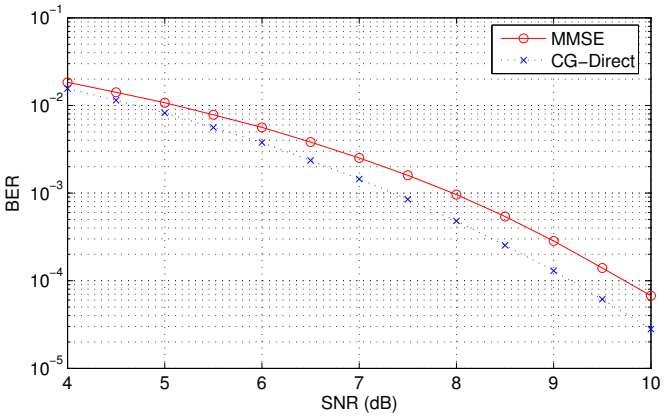

Fig. 6. Matrix equalizers performance comparison of an 8-tap sub-channel.

across a wide range of SNR values. Many researchers also observed this rapid convergent behavior of the CG algorithm that is due to its (still not fully understood) inherent regularization property that the algorithm iterates extract first the dominant directions/subspace(s) of the matrix, and only later those directions that are associated with the small singular values [25].

References

[1] L. Yang and G. B. Giannakis, "Ultra-wideband communications: An idea whose time has come," *IEEE Signal Processing Magazine*, vol. 21, no. 6, pp. 26–54, November 2004.

[2] T. Rappaport, *Wireless Communications: Principles and Practice*. Prentice Hall PTR, 2002.

[3] Z. Wang, X. Ma, and G. Giannakis, "OFDM or single-carrier block transmissions?," *IEEE Transactions on Communications*, vol. 52, no. 3, pp. 2380–394, 2004.

[4] A. Scaglione, G. B. Giannakis, and S. Barbarossa, "Redundant filterbank precoders and equalizers. parts I and II," *IEEE Transactions on Signal Processing*, vol. 47, pp. 1988–2022, July 1999.

[5] L. Deneire, B. Gyselinckx, and M. Engels, "Training sequence vs. cyclic prefix : a new look on single carrier communication," *IEEE Communication Letters*, vol. 5, no. 7, pp. 292–294, 2001.

[6] D. Falconer, S. L. Ariyavisitakul, A. Benyamin-Seeyar, and B. Eidson, "Frequency domain equalization for single-carrier broadband wireless systems," *IEEE Communications Magazine*, pp. 58–66, April 2002.

[7] M. Heikkila, K. Ruotsalainen, and J. Lilleberg, "Space-time equalization using conjugate-gradient algorithm in WCDMA downlink," *Proc. 13th IEEE Intl. Symp. on Personal Indoor Mobile and Radio Comm., Lisbon*, September 2002.

[8] P. Radosavljevic, J. Cavallaro, and A. de Baynast, "ASIP architecture implementation on channel equalization algorithms for mimo systems in WCDMA downlink," *IEEE Vehicular Technology Conference*, September 2004.

[9] C. Dumard, F. Kaltenberger, and K. Freudenthaler, "Low-cost LMMSE equalizer based on Krylov subspace methods for HSDPA," *IEEE Trans. Wireless Comm.*, to appear.

[10] P. P. Vaidyanathan, *Multirate systems and filter banks*. Prentice Hall, Englewood Cliffs, 1993.

[11] G. Harikumar and Y. Bresler, "FIR perfect signal reconstruction from multiple convolutions: Minimum deconvolver orders," *IEEE Transactions on Signal Processing*, vol. 46, January 1998.

[12] J. G. Proakis, *Digital Communications*. New York: McGraw-Hill, 2000.

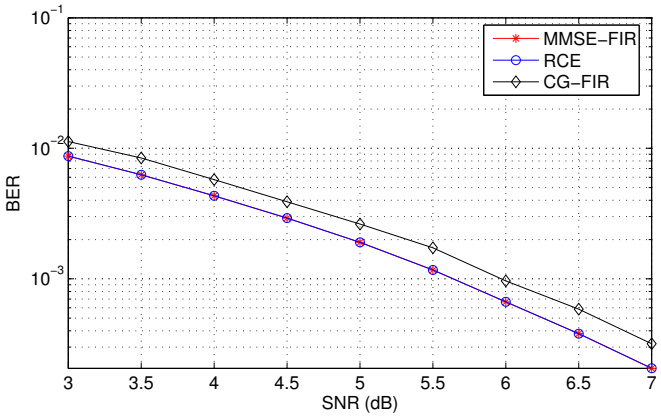

Fig. 7. FIR equalizers performance comparison of a 64-tap sub-channel.

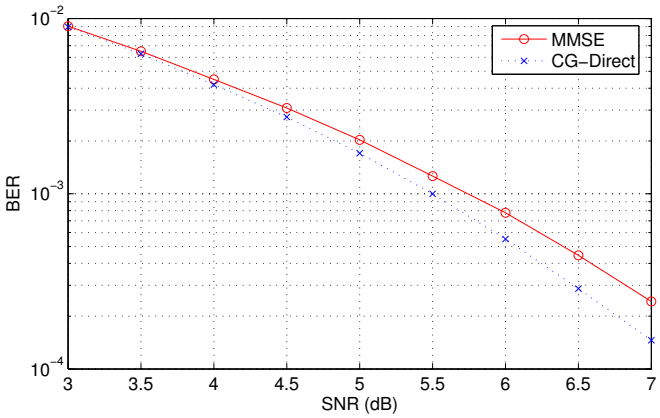

Fig. 8. Matrix equalizers performance comparison of a 64-tap sub-channel.

[13] G. B. Giannakis, Y. Hua, P. Stoica, and L. Tong, *Signal Processing Advances in Wireless and Mobile Communications - Volume I, Trends in Channel Estimation and Equalization.* Prentice-Hall, September 2000.
[14] H. Hassibi, B.; Vikalo, "On the sphere-decoding algorithm I. Expected complexity," *IEEE Trans. Signal Processing*, vol. 53, no. 8, pp. 2806–2818, 2005.
[15] A. Paulraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications.* Cambridge University Press, 2003.
[16] R. H. Chan and M. K. Ng, "Conjugate gradient methods for Toeplitz systems," *SIAM Review*, vol. 38, pp. 427–482, 1996.
[17] M. K. Ng, *Iterative methods for Toeplitz systems.* Oxford University Press, 2004.
[18] G. H. Golub and C. F. Van Loan, *Matrix Computations.* Baltimore, Maryland: Johns Hopkins University Press, 1983.
[19] T. Strohmer, "Rates of convergence for the approximation of dual shift-invariant systems in $\ell^2(\mathbb{Z})$," *J. Four. Anal. Appl.*, vol. 5, no. 6, pp. 599–615, 2000.
[20] T. Strohmer, "Four short stories about Toeplitz matrix calculations," *Linear Algebra Appl.*, vol. 343–344, pp. 321–344, 2002.
[21] M. R. Hestenes and E. L. Stiefel, "Methods of conjugate gradients for solving linear systems," *Journal of Research of the National Bureau of Standards*, vol. 49, pp. 409–436, 1952.
[22] M. Hanke, *Conjugate gradient type methods for ill-posed problems.* Harlow: Longman Scientific & Technical, 1995.
[23] Åke Björck, *Numerical Methods for Least Squares Problems.* Society for Industrial and Applied Mathematics, 1996.
[24] S. Pereverzev and E. Schock, "Morozov's discrepancy principle for Tikhonov regularization of severely ill-posed problems in finite-dimensional subspaces.," *Numerical Functional Analysis and Optimization*, vol. 21, pp. 901–916, 2000.
[25] P. C. Hansen, *Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion.* Society for Industrial and Applied Mathematics, 1999.
[26] A. Tikhonov, "Solution of incorrectly formulated problems and the regularization method," *Sov. Math., Dokl.*, pp. 1035–1038, 1963. English translation of Dokl. Akad. Nauk SSSR 151, pp.501-504, 1963.
[27] A. Frommer and P. Maass, "Fast CG-based methods for Tikhonov–Phillips regularization," *SIAM Journal on Scientific Computing*, vol. 20, no. 5, pp. 1831–1850, 1999.