Fast and Low Complexity Blind Equalization via Subgradient Projections

Alper Tunga Erdogan and Can Kizilkale

***Abstract*—We propose a novel blind equalization method based on subgradient search over a convex cost surface. This is an alternative to the existing iterative blind equalization approaches such as the Constant Modulus Algorithm (CMA), which often suffer from the convergence problems caused by their nonconvex cost functions. The proposed method is an iterative algorithm called SubGradient based Blind Algorithm (SGBA) for both real and complex constellations, with a very simple update rule. It is based on the minimization of the l_∞ norm of the equalizer output under a linear constraint on the equalizer coefficients using subgradient iterations. The algorithm has a nice convergence behavior attributed to the convex l_∞ cost surface as well as the step size selection rules associated with the subgradient search. We illustrate the performance of the algorithm using examples with both complex and real constellations, where we show that the proposed algorithm's convergence is less sensitive to initial point selection, and a fast convergence behavior can be achieved with a judicious selection of step sizes. Furthermore, the amount of data required for the training of the equalizer is significantly lower than most of the existing schemes.**

***Index Terms*—Blind equalization, convex optimization, subgradient.**

I. Introduction

BLIND equalization has been a research focus for several decades. The goal has been the development of effective and low-complexity algorithms that avoid the unnecessary consumption of useful bandwidth by the use of training data. Several approaches have been proposed to achieve this goal. Among them, we can list (Explicit) High-Order Statistics (HOS) Methods (e.g., [1]–[3]), Constant Modulus Type Algorithms (e.g., [4], [5]), Subspace Methods for Multibranch Channels (e.g., [6], [7]), Decision-Directed and Finite Alphabet Methods (e.g., [8]–[11]), and Maximum Likelihood Methods (e.g., [12]).

For the evaluation and the comparison of these blind methods, various properties of the algorithms are to be taken into consideration. From the practical implementation point of view, the complexity of an algorithm is an important issue. The Constant Modulus Algorithm (CMA) stands out as probably the most practical blind algorithm due to its iterative structure with a simple update rule, which is why it has been the blind equalization algorithm of choice in several real-life applications.

Manuscript received December 12, 2003; revised August 19, 2004. This work was supported in part by a TUBITAK Career Award under Contract 104E073. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Nicholas D. Sidiropoulos.

The authors are with the Electrical and Electronics Engineering Department, Koc University, Istanbul, Turkey (e-mail: alper@stanfordalumni.org; alperdogan@ku.edu.tr; ckizilkale@ku.edu.tr).

Digital Object Identifier 10.1109/TSP.2005.849195

Another property of interest is the convergence behavior. It is desirable that the algorithms converge to an "equalizing point" in a small number of iterations so that the training and tracking can be done in a short time interval by "consuming" minimum amount of computational resources and data. Due to the undesired local minima and saddle points of its cost function, the CMA suffers from ill-convergence (convergence to undesirable equilibrium points for arbitrary length sample-spaced finite impulse response (FIR) equalizers [13]) or slow convergence problems (due to capture by saddle points [5], [14]). Various algorithms targeting better convergence behavior have been put forward in the past. Among these, the super-exponential algorithm by Shalvi–Weinstein [15] has a fast convergence behavior. In fact, it is shown in [16] that it can be interpreted as a gradient search algorithm with an optimal step size rule, which maximizes the speed of convergence.

We can list some additional algorithm properties to be considered such as the robustness against the variations in the underlying model assumed by the blind equalization method (an important issue for Subspace Methods) and the ability to equalize single branch sample-spaced channels (i.e., the implicit or explicit use of higher order statistics).

The convergence issue is closely related to the surface structures of the cost functions used by the blind algorithms [5], [13], [14], [17]. The existence of saddle points and local minima in cost functions may cause slow and ill-convergence. Therefore, better behavior can be obtained by the use of convex cost functions as they have only global minima and no saddle points. Reference [18] investigates convex cost functions for blind equalization and proposes minimizing the l_∞ norm of the equalizer output under a linear constraint on the equalizer coefficient vector as the convex blind equalization approach. The choice of the l_∞ cost function solves the slow and ill-convergence issues related to local minima and saddle points. However, as the l_∞ cost function is nondifferentiable, a simple gradient descent approach like the one used in CMA cannot be used. Reference [18] proposes converting the l_∞ cost function to a differentiable one by approximating it using an l_p norm cost function with a large p value. Better performance can be attained by direct l_∞ norm minimization. However, the existing l_∞ norm minimization methods that are based on Linear Programming (LP) [19], [20] are computationally demanding.

The focus of this paper is the development of low-complexity algorithms that minimize the l_∞ cost function. Our approach is based on subgradient methods for nondifferentiable cost functions: Although the l_∞ cost function is nondifferentiable, the subgradient approach can provide us with iterative algorithms with simple update rules. In addition, we can benefit from the step-size selection rules, which have long been investigated in the area of subgradient optimization of convex functions to develop algorithms with fast convergence. We show that the use of subgradient methods with proper step-size selection rules leads to iterative low-complexity blind algorithms with desirable convergence behaviors.

The organization of the article is as follows: Section II provides basic mathematical preliminaries and the notation used in the paper. The blind equalization setup used throughout the paper is also provided in this section. Section III provides background on blind equalization based on the l_∞ norm minimization, and Section IV outlines some of the methods for subgradient optimization. Section V is the major part of the paper, where the application of subgradient methods to the blind equalization problem and the corresponding algorithms are provided. In this section, we first derive the Subgradient based Blind Algorithm (SGBA) and then discuss the step-size selection rules for SGBA. We also provide a variation of SGBA for complex channels. In addition, a Newton-like weighted version of SGBA is also provided with a proof of convergence. The complexity of the proposed SGBA algorithm is also discussed in Section V. In Section VI, we provide examples to illustrate the performance of the proposed algorithms. Finally, Section VII contains the conclusion.

II. Preliminaries and Notation

A. Subgradients

Let $f(\mathbf{x})$ be a convex and possibly nondifferentiable function with domain S. We assume that S is a convex set, and $\mathcal{P}_S$ is the operator that corresponds to the projection to the set S. The subdifferential of $f(\mathbf{x})$ at point $\mathbf{x}$ is defined as the set [21], [22]

$$\partial f(\mathbf{x}) = \{\mathbf{g} \mid f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \mathbf{g}, \mathbf{y}-\mathbf{x}\rangle \ \forall \mathbf{y} \in S\} \quad (1)$$

where $\langle \cdot, \cdot \rangle$ is the inner product. A vector $\mathbf{g}$, which is a member of $\partial f(\mathbf{x})$, is called a subgradient of $f(\mathbf{x})$ at $\mathbf{x}$. Based on the definition provided in (1), a subgradient can be used to construct an affine function that provides a global lower bound to the convex function $f(\mathbf{x})$. This fact is illustrated in Fig. 1, where two affine functions that are tangent to the function f at a point $\mathbf{x}$, where f is not differentiable, are shown. It is clear from this figure that both affine functions and their convex combinations provide global lower bounds for the function f. When the function $f(\mathbf{x})$ is differentiable at point $\mathbf{x}$, then $\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}$, where $\nabla f(\mathbf{x})$ is the gradient of function $f(\mathbf{x})$ at point $\mathbf{x}$, i.e., if a function is differentiable at a point, its subgradient is unique and equivalent to the gradient at that point. As shown in Section IV, subgradient-based iterative algorithms can be developed to minimize convex and possibly nondifferentiable functions.

B. Norms

We use various norm definitions in this paper. For a finite size vector $\mathbf{x} \in \mathcal{C}^N$, we define the p-norm as

$$\|\mathbf{x}\|_p = \begin{cases} \left(\sum_{k=1}^{N} |x_k|^p\right)^{1/p}, & 1 \leq p < \infty \\ \max_{k\in\{1,\ldots,N\}} |x_k|, & p = \infty. \end{cases} \quad (2)$$

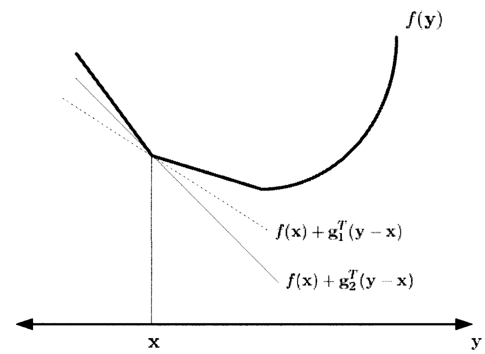

Fig. 1. Supporting hyperplanes and the subgradient.

The extension of the p-norm definition to the vector space of discrete time sequences is achieved by changing the index set $\{1,\ldots,N\}$ to the set of integers $\mathcal{Z}$.

We define the $\mathbf{\Pi}$-weighted norm as

$$\|\mathbf{x}\|_{\mathbf{\Pi}} = \mathbf{x}^H \mathbf{\Pi} \mathbf{x} \quad (3)$$

where $\mathbf{\Pi}$ is a positive definite matrix.

A special property of the norm functions (in particular the ∞ norm) that we frequently quote in the paper is their convexity. More specifically, a function $f(\mathbf{x}) = \|\mathbf{Ax}+\mathbf{b}\|$, which is a composite function of a norm function and an affine function, is convex. We can show the convexity of this type of function through use of the triangle inequality and scaling properties of the norms: For a given $\lambda \in [0,1]$

$$\begin{aligned} f(\lambda\mathbf{x}+(1-\lambda)\mathbf{y}) &= \|\mathbf{A}(\lambda\mathbf{x}(1-\lambda)\mathbf{y})+\mathbf{b}\| \\ &= \|\lambda(\mathbf{Ax}+\mathbf{b})+(1-\lambda)(\mathbf{Ay}+\mathbf{b})\| \\ &\leq \|\lambda(\mathbf{Ax}+\mathbf{b})\| + \|(1-\lambda)(\mathbf{Ay}+\mathbf{b})\| \\ &= \lambda\|\mathbf{Ax}+\mathbf{b}\| + (1-\lambda)\|\mathbf{Ay}+\mathbf{b}\| \\ &= \lambda f(\mathbf{x}) + (1-\lambda) f(\mathbf{y}). \end{aligned}$$

Therefore, the cost functions of this form have only global minima and no saddle points, which make them attractive for adaptive signal processing applications.

C. Blind Equalization Setup

Throughout this paper, we assume the symbol-spaced equalization setup shown in Fig. 2. Here, we have the following.

- $\{x_i \in \{-2\cdot M+1,\ldots,2\cdot M-1\}\}$ is the information sequence sent by the transmitter.
- $\{h_i; i \in \{0,\ldots,N_h-1\}\}$ is the impulse response of the symbol-spaced channel, which is the combination of the linear distortion caused by the communication medium and the pulse-shaping filter.
- $\{y_i\}$ is the input sequence to the equalizer.
- $\{w_i; i=0,\ldots,N_w-1\}$ is the set of equalizer coefficients, and $\mathbf{w} = [w_0\ w_1\ \cdots\ w_{N_w-1}]^T$ is the equalizer coefficient vector.
- $\{z_i\}$ is the equalizer output.

The goal of equalization is to make the equalizer output $\{z_i\}$ as close as possible to a delayed version of the input $\{x_i\}$. No *a priori* knowledge of the channel is assumed. Furthermore, no training sequence is available for adapting the equalizer to the

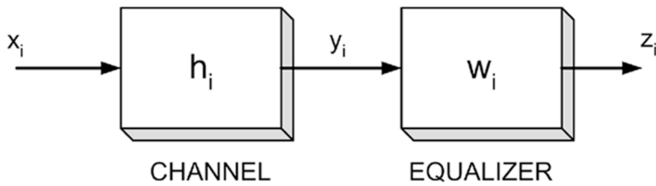

Fig. 2. Equalization setup.

unknown channel. As a result, the blind equalization algorithm can only use the input signal $\{y_i\}$ to train the equalizer.

In order to evaluate the performance of the blind equalization algorithms, we use the following measures defined over the combined channel $c_i = h_i * w_i$:

- *Open eye measure (Normalized peak distortion)*:

$$\rho(c) = \frac{\sum |c_i| - p}{p}; \tag{4}$$

- *ISI (Normalized intersymbol interference energy)*:

$$\text{ISI}(c) = \frac{\sum |c_i|^2 - p^2}{p^2} \tag{5}$$

where $p = \max_i |c_i|$. Both measures are non-negative and attain zero value only under the perfect equalization condition.

III. l_∞ Norm as the Convex Cost Function for Blind Equalization

Due to the undesired local optima and saddle point structure of their cost functions, the CMA and some other blind equalization algorithms suffer from slow and ill convergence problems. This fact led to the search for the cost functions with better geometrical structure. Reference [18] investigates the properties of convex functions as the candidates for the cost functions for blind equalization. In the same reference, minimizing the l_∞ norm of the output of the equalizer, under a linear constraint on equalizer coefficients, is proposed as a blind equalization approach. Considering the equalization setup in Fig. 2, under the assumptions that

1) the input constellation has the maximum magnitude symmetry around zero such that $\max x_k = (2 \cdot M - 1)$ and $\min x_k = -2 \cdot M + 1$;
2) $\{x_k\}$ is sufficiently rich in terms of variations in time;

it can be shown that [18]

$$\|z\|_\infty = (2 \cdot M - 1)\|c\|_1 \tag{6}$$

where $c_i = h_i * w_i$ is the impulse response of the combined channel and equalizer. Therefore, minimizing the l_∞ norm of the output is equivalent to minimizing the l_1 norm of the overall impulse response. A linear constraint of the form

$$\mathbf{u}^T\mathbf{w} = e \quad \mathbf{u} \neq 0, \quad e \neq 0 \tag{7}$$

is needed to avoid the all zeros solution $\mathbf{w} = \mathbf{0}$.

The equivalence between the minimization of the l_1 norm of $\{c\}_i$ and the equalization of the channel $\{h_i\}$ is given by the following proposition, which is proven in [18].

Proposition: Given the impulse response $\{h_i; i \in \mathcal{Z}\}$ and the impulse response of its ideal convolutional inverse $\{\varphi_i; i \in \mathcal{Z}\}$ such that

$$\varphi_n * h_n = \delta_n \tag{8}$$

where $|\varphi_n|$ is assumed to have a single maximum point located at m, then the solution to the problem

$$\begin{aligned} &\text{minimize } \|h_n * w_n\|_1 \quad \text{(Problem 1)} \\ &\text{s.t. } w_0 = 1 \end{aligned}$$

is given by

$$w_n = \frac{\varphi_{n+m}}{\varphi_m} \quad \forall n. \tag{9}$$

As a result, under the previous assumptions made about $\{x_i\}$, the equalizer coefficients $\{w_i; i \in \mathcal{Z}\}$ obtained by solving the convex optimization problem

$$\begin{aligned} &\text{minimize } \|z\|_\infty \quad \text{(Problem 2)} \\ &\text{s.t. } w_L = 1 \end{aligned}$$

for some $L \in \mathcal{Z}$ will be the scaled and time-shifted inverse of $\{h_i\}$ such that

$$h_n * w_n = G\delta_{n-d} \tag{10}$$

for some magnitude G and delay d [18]. Hence, the selection of the fixed tap location is strictly related to the optimal delay selection problem. Given no *a priori* knowledge about the channel, it is reasonable to fix the middle tap as constant.

Note that the solution of *Problem 2* is equivalent to the solution of *Problem 1*, which is the minimization of a convex cost function of $\{c_n\}$, the combined equalizer, and channel impulse response, under a linear constraint on $\mathbf{w}$. Similarly, CMA [4] and Shalvi–Weinstein [15], [23] algorithms have been shown to be equivalent to the minimization of a nonconvex cost function of $\{c_n\}$ under a constant 2-norm constraint on $\{c_n\}$ [17], [24]. Furthermore, [16] shows that the superexponential [15] algorithm can be interpreted as a gradient search algorithm for this constrained nonconvex optimization problem with optimally selected step size. In this paper, we develop a gradient-search-like algorithm that targets to achieve the solution of the constrained convex optimization problem defined in *Problem 1*.

Placing the FIR constraint on the equalizer coefficients will still preserve the convex nature of the problem. However, there will be a performance degradation due to this constraint.

What remains is to develop preferably low-complexity algorithms to solve this convex optimization problem. Since the l_∞ norm is not differentiable, we cannot develop gradient descent algorithms similar to CMA for the cost function of *Problem 2*. Reference [18] proposes the minimization of the l_p norm of the equalizer output (with a large p value), which is a differentiable cost function, as an approximation.

Our approach is to find an iterative algorithm, with a similar complexity to gradient descent algorithms, that targets minimization of the l_∞ norm directly. The real-time implementation requirement places a limit on the data window used by the adaptive algorithms. In order to reflect this requirement, we can modify the previous convex optimization problem as

$$\begin{aligned}&\text{minimize } \|z \odot r\|_\infty \quad \text{(Problem 3)}\\ &\text{s.t. } w_L = 1\end{aligned}$$

where $\{r_n\}$ is a rectangular window function, with

$$r_n = \begin{cases} 1, & 0 \le n \le \Omega - 1 \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

and $z \odot r$ represents the element-by-element multiplication of two discrete time sequences (i.e., $(z \odot r)_k = z_k r_k$). Hence, the final problem would be equivalent to the minimization of the l_∞ norm of the finite size vector

$$\underbrace{\begin{bmatrix} z_0 \\ z_1 \\ \vdots \\ z_{\Omega-1} \end{bmatrix}}_{\mathbf{z}} = \underbrace{\begin{bmatrix} y_0 & y_{-1} & \cdots & y_{-N_w+1} \\ y_1 & y_0 & \cdots & y_{-N_w+2} \\ \vdots & \ddots & \ddots & \vdots \\ y_{\Omega-1} & y_{\Omega-2} & \cdots & y_{\Omega-N_w} \end{bmatrix}}_{\mathbf{Y}} \underbrace{\begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{N_w-1} \end{bmatrix}}_{\mathbf{w}}. \tag{12}$$

Since $\mathbf{z}$ is a linear function of the search vector $\mathbf{w}$ and the constraint $w_L = 1$ is a linear constraint, the corresponding l_∞ norm minimization problem can be casted to the well-known Linear Programming (LP) formulation:

$$\begin{aligned}&\text{minimize } t \quad \text{(Problem 4)}\\ &\text{s.t. } \begin{bmatrix} -\mathbf{1} & \boldsymbol{\Gamma} \\ -\mathbf{1} & -\boldsymbol{\Gamma} \end{bmatrix} \begin{bmatrix} t \\ \mathbf{w_s} \end{bmatrix} \le \begin{bmatrix} -\mathbf{q} \\ \mathbf{q} \end{bmatrix}\end{aligned}$$

where $\boldsymbol{\Gamma}$ is equivalent to $\mathbf{Y}$ matrix with the $(L+1)$th column deleted, $\mathbf{q}$ is the $(L+1)$th column of $\mathbf{Y}, \mathbf{w}_s$ is the $\mathbf{w}$ with the $(L+1)$th element deleted (since $w_L = 1$), and $[t \;\; \mathbf{w_s}^T]^T$ is the search vector. There are various approaches to solve the LP of *Problem 4*, and [25] provides a comprehensive overview of these approaches. However, obtaining the solution of an LP is a computationally demanding task and, therefore, is not suitable for real-time communications applications in general.

As an alternative to the l_p approximation of [18] and the LP approach, we propose a low-complexity iterative method to solve *Problem 3*, which is more suitable for real-time applications. Our approach is based on subgradient optimization techniques for nondifferentiable cost functions. The algorithms based on these techniques have a desirable convergence behavior due to both the convex nature of the cost function and the special step-size selection rules provided by the subgradient approaches. Furthermore, their computational complexities are low, which makes them attractive for practical implementations.

In the next section, we provide an overview of subgradient methods in the context of our discussion. Later, we present the blind equalization algorithms based on these subgradient methods.

IV. Review of Subgradient Methods

The gradient descent methods and their applications in signal processing, especially in adaptive filtering, are fairly well known [26], [27]. These methods use the first-order derivative information to update the search vector such that the objective function value is iteratively improved. The goal is to eventually reach the neighborhood of a global or a local optimal point, depending on the properties of the cost function.

Given that $f(\mathbf{w})$ is a differentiable cost function to be minimized, the gradient descent method consists of iterations of the form

$$\mathbf{w}^{(i+1)} = \mathcal{P}_S\left\{\mathbf{w}^{(i)} - \mu^{(i)} \nabla f\left(\mathbf{w}^{(i)}\right)\right\} \tag{13}$$

where the step size μ is chosen small enough to ensure the decrease in the cost function.

Various adaptive and blind algorithms have been based on this structure due to its low complexity. The most well-known examples are the Least Mean Square (LMS) algorithm in supervised adaptive filtering and the Constant Modulus Algorithm (CMA) in blind equalization.

In the CMA case, for the equalization setup of Fig. 2, the cost function is given by [5]

$$f(\mathbf{w}) = E((\gamma - |z|^2)^2) \tag{14}$$

where γ is the level corresponding to the constant modulus. The update equation corresponding to the CMA cost function of (14) is given by

$$\mathbf{w}^{(i+1)} = \mathbf{w}^{(i)} + \mu^{(i)} \underbrace{\mathbf{y}_i z_i^* (\gamma - |z_i|^2)}_{\hat{\nabla} f(\mathbf{w}^{(i)})} \tag{15}$$

where the stochastic gradient is replaced by its estimate. Note that the projection operator in (13) does not appear in (15). The reason is that the CMA cost function in (14) is defined over the whole $\mathcal{R}^{N_w}$, and therefore, the projection operator is the identity operator.

The gradient-descent methods are applicable to only differentiable functions. The nondifferentiable counterpart of the gradient-descent algorithm is the subgradient projection method in which the gradient vector is simply replaced by a subgradient vector

$$\mathbf{w}^{(i+1)} = \mathcal{P}_S\left\{\mathbf{w}^{(i)} - \mu^{(i)} \mathbf{g}^{(i)}\right\} \tag{16}$$

where $\mathbf{g}^{(i)}$ is a subgradient picked from the subdifferential set $\partial f(\mathbf{w}^{(i)})$. Although the subgradient algorithm looks very much like the gradient descent algorithm, in the subgradient iteration, it may happen that

$$f\left(\mathbf{w}^{(i+1)}\right) > f\left(\mathbf{w}^{(i)}\right) \tag{17}$$

for any $\mu^{(i)} > 0$ [21]. However, if the $\mu^{(i)}$ parameter sequence is properly chosen, then $\mathbf{w}^{(i)}$ can be made to converge to the optimal point $\mathbf{w}^*$. In fact, the selection of the step size is a crucial point, and it has been a research focus for several decades, where various step-size selection schemes have been developed. Below, we provide a summary of these research efforts as we will use different step-size selection schemes in the subgradient-based blind equalization algorithm in Section V.

A. Constant Step Size Rule

The constant step size $\mu^{(i)} = \mu$ is the most primitive choice. However, there exists a clear tradeoff between the final accuracy of the solution, which requires a smaller μ value, and the convergence speed, which requires a larger μ value. In order to balance these requirements, a dynamic step size with diminishing value should be chosen.

B. Zero-Limit-Divergent-Sum (ZLDS) Step-Size Rule

One major result about the selection of the step-size parameter $\mu^{(i)}$ is due to Polyak [28], [29]. If the following two conditions hold:

$$\lim_{i\to\infty} \frac{\mu^{(i)}}{\left\|\mathbf{g}^{(i)}\right\|_2} = 0 \tag{18}$$

$$\sum_{i=0}^{\infty} \frac{\mu^{(i)}}{\left\|\mathbf{g}^{(i)}\right\|_2} = \infty \tag{19}$$

then $\lim_{i\to\infty} \mathbf{w}^{(i)} = \mathbf{w}^*$. Therefore, (18) (zero limit) and (19) (divergent sum) provide sufficient conditions for convergence based on the step-size selection.

A simple example of a step-size-satisfying ZLDS rule is

$$\mu^{(i)} = \frac{\mu_0}{i+1} \tag{20}$$

which satisfies the conditions in (18) and (19), due to the fact that the norm of the subgradient is bounded and nonzero in our problem. This step-size rule has been studied in [30]. Under certain assumptions regarding the cost surface, it is shown that the square distance to the optimal point $\mathbf{w}^*$ decreases with a sublinear convergence behavior, i.e., we can approximately write

$$\left\|\mathbf{w}^* - \mathbf{w}^{(i)}\right\|_2^2 \le \alpha \left\|\mathbf{w}^* - \mathbf{w}^{(0)}\right\|_2^2 \tag{21}$$

where $\alpha \propto (1/(i^p))$ for some parameter p, which is dependent on the cost surface and μ_0.

C. Relaxation Step-Size Rule

A dynamic step-size rule, which is known as the relaxation rule [21], [31], is given by

$$\mu^{(i)} = \gamma^{(i)} \frac{\left(f\left(\mathbf{w}^{(i)}\right) - f^*\right)}{\left\|\mathbf{g}^{(i)}\right\|_2^2} \tag{22}$$

where

$$0 < \gamma^l \le \gamma^{(i)} \le \gamma^u < 2 \tag{23}$$

and f^* is the minimum value of $f(\mathbf{w})$. If the step size is selected based on this rule, then it is guaranteed that

$$\left\|\mathbf{w}^{(i+1)} - \mathbf{w}^*\right\|_2 < \left\|\mathbf{w}^{(i)} - \mathbf{w}^*\right\|_2, \quad \forall i \tag{24}$$

i.e., the distance to the optimal vector decreases monotonically [30]. The choice of μ as in (22) also provides a faster rate of convergence [than the step size in (20)]. The convergence rate of this scheme has been studied in several references (see, for example, [30]–[32]). For this step-size selection, a bound for the convergence rate can be derived [30] based on the assumption that there exists some κ for which

$$f(\mathbf{w}) - f^* \ge \kappa \|\mathbf{w}^* - \mathbf{w}\|_2, \quad \forall \mathbf{w} \in S. \tag{25}$$

Using this assumption, and assuming that $\|\mathbf{g}^{(i)}\|_2 \le C$, it can be shown that [30]

$$\left\|\mathbf{w}^* - \mathbf{w}^{(i)}\right\|_2 \le q^i \left\|\mathbf{w}^* - \mathbf{w}^{(0)}\right\|_2 \tag{26}$$

where

$$q = \sqrt{1 - \gamma^l(2-\gamma^u)\frac{\kappa^2}{C^2}}. \tag{27}$$

For piecewise linear functions of the form $f(\mathbf{w}) = \|A\mathbf{w}+\mathbf{b}\|_\infty$, Brannlund shows that [31] κ in (25) can be computed as the solution of the optimization problem

$$\begin{aligned} &\min z \\ \text{s.t. } &AA^T\boldsymbol{\alpha} \le z\mathbf{1} \\ &\boldsymbol{\alpha}^T AA^T \boldsymbol{\alpha} = 1 \\ &\boldsymbol{\alpha} \ge \mathbf{0}. \end{aligned} \tag{28}$$

Unfortunately, the q factor obtained from (27) provides only a pessimistic lower bound for the rate of convergence.

D. Relaxation Step-Size Rule With Variable Target Values

The relaxation step-size rule in (22) requires *a priori* knowledge of the optimal value of the function to be optimized, which is not reasonable in most cases. The use of an estimate of f^* instead of f^* has been investigated in several references (see, for example, [33]–[35]) such that the subgradient update rule takes the form

$$\mathbf{w}^{(i+1)} = \mathcal{P}_S\left\{\mathbf{w}^{(i)} - \alpha^{(i)}\frac{\left(f\left(\mathbf{w}^{(i)}\right) - \hat{f}^*\right)}{\left\|\mathbf{g}^{(i)}\right\|_2^2}\mathbf{g}^{(i)}\right\}. \tag{29}$$

Bazaraa and Sherali did pioneering work on algorithms that do not require the knowledge of f^* [36]. Along the same lines, Kim *et al.* proposed a variable target method where $\hat{f}*^{(i)}$ is updated as a part of the algorithm [37].

Recently, Goffin and Kiwiel proposed a simple subgradient method [38] that is relatively low complexity and is guaranteed to converge. The algorithm is a variable target method, which is based on the following two simple facts [21].

1) If $\hat{f}^*$ in (29) is an overestimate of the optimal value f^*, then either
   - for some k, $f(\mathbf{w}^{(k)}) \le \hat{f}^*$;
   - $\mathbf{w}^{(i)}$ converges to an $\bar{\mathbf{w}}$ such that $f(\bar{\mathbf{w}}) \le \hat{f}^*$;

   such that the overestimate of f^* is detected.
2) If $\hat{f}^*$ in (29) is an underestimate of the optimal value f^*, then it can be shown that

$$\sum_{i=0}^{\infty} \frac{f\left(\mathbf{w}^{(i)}\right) - \hat{f}^*}{\left\|\mathbf{g}^{(i)}\right\|_2} = \infty \tag{30}$$

so that one can detect underestimate condition by checking whether the finite summation

$$\sum_{i=0}^{K} \frac{f\left(\mathbf{w}^{(i)}\right) - \hat{f}^*}{\left\|\mathbf{g}^{(i)}\right\|_2} \tag{31}$$

is above some threshold parameter.

Therefore, one can start with an arbitrary $\hat{f}^*$ and later update it based on the conditions above.

Nedic and Bertsekas proposed [30] a simpler adaptive step-size algorithm with unknown f^*, where the estimate of the optimal value is given by

$$\hat{f}*^{(i)} = \min_{0\leq j\leq i} f\left(\mathbf{w}^{(i)}\right) - \delta^{(i)}. \tag{32}$$

Here, the $\delta^{(i)}$ is updated according to

$$\delta^{(i+1)} = \begin{cases} \sigma\delta^{(i)}, & \text{if } f\left(\mathbf{w}^{(i)}\right) < \hat{f}*^{(i)} \\ \max\{\beta\delta^{(i)}, \delta\}, & \text{if } f\left(\mathbf{w}^{(i)}\right) \geq \hat{f}*^{(i)} \end{cases} \tag{33}$$

where $\delta^{(0)}, \delta, \sigma, \beta$ are fixed positive constants with $\beta < 1$ and $\sigma \geq 1$.

Sherali *et al.* [39] also proposed an algorithm with a similar complexity to the Goffin's algorithm outlined above. The algorithm proposed in [39] also incorporates the so-called dilation technique to increase the speed of convergence.

V. Iterative Blind Equalization With Subgradient Projections

In order to produce low-complexity iterative algorithms for solving *Problem 3*, we apply the subgradient projection approach, which is outlined in the previous section. We start by writing the subdifferential set for the blind equalization cost function

$$f(\mathbf{w}) = \|\mathbf{z}\|_\infty = \|\mathbf{\Gamma}\mathbf{w}_s + \mathbf{q}\|_\infty \tag{34}$$

which is given by

$$\partial f(\mathbf{w}_s) = \mathbf{Co}\left\{\left\{\mathbf{\Gamma}_{k,:}^T \,\middle|\, z_k = \|\mathbf{z}\|_\infty\right\} \cup \left\{-\mathbf{\Gamma}_{k,:}^T \,\middle|\, z_k = -\|\mathbf{z}\|_\infty\right\}\right\} \tag{35}$$

where $\mathbf{Co}\{\}$ is the convex-hull operation, and $\mathbf{\Gamma}_{k,:}$ is the kth row of the matrix $\mathbf{\Gamma}$, which corresponds to the equalizer input vector-producing output z_k. Here, we obtained the subdifferential set by using the following properties of subgradients.

- Given a set of convex functions $h_i(\mathbf{x}), i = 1\ldots m$, with the corresponding subdifferential sets $\partial h_i(\mathbf{x})$, if

  $$h(\mathbf{x}) = \max_{i=1\ldots m} h_i(\mathbf{x}) \tag{36}$$

  then

  $$\partial h(\mathbf{x}) = \mathbf{Co}\left\{\bigcup\{\partial h_i(\mathbf{x}) \,|\, h_i(\mathbf{x}) = h(\mathbf{x})\}\right\} \tag{37}$$

  where $\mathbf{Co}$ represents the convex hull operation.
- Given $f(\mathbf{x}) = h(\mathbf{A}\mathbf{x} + \mathbf{b})$, where h is convex, then $\partial f(\mathbf{x}) = \{\mathbf{A}^H\mathbf{y} \,|\, \mathbf{y} \in \partial h(\mathbf{A}\mathbf{x} + \mathbf{b})\}$.

On the inspection of the subdifferential set in (35), it can be seen that the search direction for the subgradient projection algorithm is obtained from the equalizer input vectors causing the maximum magnitude equalizer output within the given window. If J is the set of the time instants for which the maximum magnitude is achieved, i.e., $J = \{k \,|\, |z_k| = \|\mathbf{z}\|_\infty\}$, then a possible search direction for the subgradient projection algorithm is given by

$$\mathbf{m} = -\sum_{k\in J} \xi_k \,\mathrm{sign}(z_k)\mathbf{\Gamma}_{k,:}^T \tag{38}$$

where $\sum_{k\in J}\xi_k = 1$, and $\xi_k \geq 0$. For convenience, one may choose $\xi_l = 1$ for some $l \in J$ and $\xi_k = 0$ for $k \neq l$, in which case, the search direction simplifies to

$$\mathbf{m} = -\mathrm{sign}(z_l)\mathbf{\Gamma}_{l,:}^T. \tag{39}$$

As a result, we can write the update rule for the SGBA as

$$\mathbf{w}_s^{(i+1)} = \mathbf{w}_s^{(i)} - \mu^{(i)}\mathrm{sign}\left(z_{l^{(i)}}^{(i)}\right)\mathbf{\Gamma}_{l^{(i)},:}^T \tag{40}$$

where we have the following.

- $l^{(i)} \in \{0, \ldots, \Omega - 1\}$ is the index where maximum magnitude output is achieved at the ith iteration.
- $\mu^{(i)}$ is the step size at the ith iteration.

A. Step-Size Selection for SGBA

There are various possible choices for $\mu^{(i)}$, as mentioned in Section IV.

- **Constant step size** ($\mu^{(i)} = \mu$): This is the simplest choice for the step size. Small values of μ will cause slow convergence, and high values will cause high misadjustment, where the misadjustment is defined as the percent deviation from the optimal value. As shown in [30], for the constant step-size rule

  $$\lim_{i\to\infty} f_i \leq f^* + \frac{\mu C^2}{2} \tag{41}$$

  where C is an upperbound for the norm of subgradients as described in the previous section. We can set C to

  $$C = \max_i \|\mathbf{\Gamma}_{i,:}\|_2 \tag{42}$$

  in our problem as the subgradients are the rows of the $\mathbf{\Gamma}$ matrix. Therefore, for a given constant step size μ, the distance to the optimal point is always bounded, and there is no issue of unstable error growth due to the fact that the norm of the subgradient is always bounded.
- **Normalized Constant Step Size** ($\mu^{(i)} = (\mu/(\|\mathrm{g}^{(i)}\|_2))$: This is in analogy to Normalized-LMS (NLMS), where the step size has a dynamic adjustment based on the norm of the current search vector.
- **Zero-Limit-Divergent-Sum (ZLDS) Step Size** ($\mu^{(i)} = ((\mu_0)/(i+1))$: This step-size rule satisfies the Polyak's conditions in (18) and (19), and therefore, the SGBA using this step size is guaranteed to converge (to the solution of *Problem 3*), as we mentioned in the previous section.
- **Relaxation Rule with Variable Target Value**:

  $$\mu^{(i)} = \alpha\frac{\left|z_{l^{(i)}}^{(i)}\right| - \hat{f}^{*(i)}}{\left\|\mathbf{\Gamma}_{l^{(i)},:}\right\|_2^2} \tag{43}$$

  as in the relaxation rule of (22) with $\alpha \in (0, 2)$. If the target value f^* were known *a priori*, then this scheme would converge with a monotonic decrease in distance to the solution of *Problem 3*. However, as f^* is not known *a priori*, we need to use a variable target value scheme. There are different approaches for updating $\hat{f}^{*(i)}$.

— The use of variable target schemes outlined in the previous section [30], [38], [39]. Among these algorithms, we used the Goffin&Kiwiel algorithm [38] in our simulations in Section VI. This algorithm has a relatively simple rule for updating the target value $\hat{f}^{*(i)}$, and it is guaranteed to converge to f^*.

— As an alternative variable target scheme, for constant modulus signaling, we propose the use of

$$\hat{f}^{*(i)} = \frac{1}{(\Omega)^{1/p}} \left\| \mathbf{\Gamma} \mathbf{w}_s^{(i)} + \mathbf{q} \right\|_p, \quad p = 1, 2 \tag{44}$$

which is the average of the magnitude of the output for $p = 1$ or the RMS of the output for $p = 2$ within the selected window. The resulting μ_i will always be nonnegative since

$$\frac{1}{(\Omega)^{1/p}} \left\| \mathbf{\Gamma} \mathbf{w}_s^{(i)} + \mathbf{q} \right\|_p \leq \left\| \mathbf{\Gamma} \mathbf{w}_s^{(i)} + \mathbf{q} \right\|_\infty \tag{45}$$

with equality if the magnitude of the equalizer output is constant, which refers to the perfect equalization condition. For nonconstant modulus constellations proper adjustments can be made to $\hat{f}^{*(i)}$ by scaling with a constellation-dependent parameter.

B. Variations on SGBA

1) SGBA for Complex Constellations: As shown in [19], the results of [18] for real constellations can be extended to the complex constellations that satisfy the property $\max_k |\Re e\{x_k\}| = \max_k |\mathcal{I}m\{x_k\}|$, where the corresponding optimization problem is defined as

$$\begin{aligned} &\text{minimize } \|\Re e\{\mathbf{z}\}\|_\infty = \|\Re e\{\mathbf{Y}\mathbf{w}\}\|_\infty \quad \text{(Problem 5)} \\ &\text{s.t. } \Re e\{w_L\} = 1 \end{aligned}$$

where $\mathbf{Y}$ and $\mathbf{w}$ are as defined in (12). Therefore, the update rule of the SBGA algorithm in this case becomes

$$\begin{aligned} \mathbf{w}^{(i+1)} &= \mathbf{w}^{(i)} - \mu^{(i)} \text{sign}\left(\Re e\left\{z_{l^{(i)}}^{(i)}\right\}\right) \mathbf{Y}_{l^{(i)},:}^H \\ \Re e\left\{w_L^{(i+1)}\right\} &= 1. \end{aligned}$$

The second equation in the update rule above is the projection to the constraint set. Note that in this case, it is possible to convert *Problem 5*, which is a constrained minimization of the infinity norm of a linear expression, to an unconstrained problem of minimizing the infinity norm of an affine expression as in (34).

2) Weighted SGBA: We introduce the following modification to the update rule in (40):

$$\mathbf{w}_s^{(i+1)} = \mathbf{w}_s^{(i)} - \mu^{(i)} \mathbf{\Pi}^{-1} \mathbf{g}^{(i)} \tag{46}$$

where $\mathbf{g}^{(i)} = \text{sign}(z_{l^{(i)}}^{(i)}) \mathbf{\Gamma}_{l^{(i)},:}^T$ is the subgradient. The only change in (46) compared with the original SGBA is the introduction of the weighting matrix $\mathbf{\Pi}^{-1}$, which is constrained to be positive definite. If we choose the step size as

$$\mu^{(i)} = \gamma_i \frac{\left| z_{l^{(i)}}^{(i)} \right| - f^*}{\left\| \mathbf{\Pi}^{-1} \mathbf{g}^{(i)} \right\|_{\mathbf{\Pi}}^2} \tag{47}$$

where $0 < \gamma_i < 2$, then we can show that, given $f^{(i)} \neq f^*$

$$\left\| \mathbf{w}_s^{(i+1)} - \mathbf{w}_s^* \right\|_{\mathbf{\Pi}} < \left\| \mathbf{w}_s^{(i)} - \mathbf{w}_s^* \right\|_{\mathbf{\Pi}} \tag{48}$$

i.e., the weighted distance to the optimal point decreases monotonically. In order to prove this statement, we write

$$\begin{aligned} &\left\| \mathbf{w}_s^{(i+1)} - \mathbf{w}_s^* \right\|_{\mathbf{\Pi}} \\ &= \left\| \mathbf{w}_s^{(i)} - \mathbf{w}^* - \mu^{(i)} \mathbf{\Pi}^{-1} \mathbf{g}^{(i)} \right\|_{\mathbf{\Pi}} && (49) \\ &= \left\| \mathbf{w}_s^{(i)} - \mathbf{w}^* \right\|_{\mathbf{\Pi}}^2 - 2\mu^{(i)} \mathbf{g}^{(i)T} \left(\mathbf{w}_s^{(i)} - \mathbf{w}_s^* \right) \\ &\quad + \mu^{(i)2} \left\| \mathbf{\Pi}^{-1} \mathbf{g}^{(i)} \right\|_{\mathbf{\Pi}}^2 && (50) \\ &\leq \left\| \mathbf{w}_s^{(i)} - \mathbf{w}_s^* \right\|_{\mathbf{\Pi}}^2 - 2\mu^{(i)} \left(|z_{l^{(i)}}^{(i)}| - f^* \right) \\ &\quad + \mu^{(i)2} \left\| \mathbf{\Pi}^{-1} \mathbf{g}^{(i)} \right\|_{\mathbf{\Pi}}^2 && (51) \\ &= \left\| \mathbf{w}_s^{(i)} - \mathbf{w}_s^* \right\|_{\mathbf{\Pi}}^2 - \gamma_i (2 - \gamma_i) \frac{\left(\left| z_{l^{(i)}}^{(i)} \right| - f^* \right)^2}{\left\| \mathbf{\Pi}^{-1} \mathbf{g}^{(i)} \right\|_{\mathbf{\Pi}}^2} && (52) \end{aligned}$$

where the inequality in (51) is obtained by using the fact that $\mathbf{g}^{(i)}$ is a subgradient, and therefore, we can use the inequality in (1). For $0 < \gamma_i < 2$, the second term in (52) is strictly positive, and therefore, we obtain the inequality in (48). As a result, since the weighted distance to the optimal point decreases monotonically and it is lower bounded by 0, the convergence to the optimal point is achieved in the limit.

Note that this proof of convergence with monotonic decrease in weighted distance to the optimal point is valid for any positive definite $\mathbf{\Pi}$. If we choose $\mathbf{\Pi} = \mathbf{I}$, then the weighted SGBA would be equivalent to the ordinary SGBA algorithm with relaxation step-size rule, and therefore, we have a proof of convergence covered for this case.

Another choice of interest is the selection of $\mathbf{\Pi}$ as the covariance of the observations vector

$$\mathbf{y}_i = [y_i \quad y_{i-1} \quad \cdots \quad y_{i-L+1} \quad y_{i-L-1} \quad \cdots \quad y_{i-N_w+1}]^T \tag{53}$$

i.e.,

$$\begin{aligned} \mathbf{\Pi} &= R_{\mathbf{y}} && (54) \\ &= E\left(\mathbf{y_i} \mathbf{y_i}^T\right) && (55) \end{aligned}$$

which would yield

$$\begin{aligned} \left\| \mathbf{w}_s^{(i)} - \mathbf{w}_s^* \right\|_{\mathbf{\Pi}} &= \left(\mathbf{w}_s^{(i)} - \mathbf{w}_s^* \right)^T R_{\mathbf{y}} \left(\mathbf{w}_s^{(i)} - \mathbf{w}_s^* \right) && (56) \\ &= E\left(\left| \mathbf{w}_s^{(i)T} \mathbf{y}_i - \mathbf{w}_s^{*T} \mathbf{y}_i \right|^2 \right) && (57) \end{aligned}$$

which is the mean square of the error between the outputs of a given $\mathbf{w}^{(i)}$ and the optimal equalizer $\mathbf{w}^*$. Therefore, if the covariance matrix of the observations vector is estimated and its inverse is used as the weighting matrix, then the corresponding weighted SGBA will provide a monotonic decrease in the output mean square error relative to the optimal equalizer $\mathbf{w}^*$. Furthermore, the update rule in (46) will mimic the Newton algorithm.

C. Complexity Analysis

The basic advantage of SGBA is its simple update rule [(40) and (46)], which makes it suitable for hardware-only implementations. The update vector is calculated from the inputs and the outputs of the equalizer with a simple mathematical operation, and therefore, the algorithm is less prone to errors due to fixed-point implementations.

In SGBA, for each iteration, we need to do the following:

- Compute Ω output samples, which requires ΩN_w multiplications and $\Omega(N_w - 1)$ additions.
- Determine the peak location, which requires $\Omega - 1$ subtractions.
- Compute the step size whose complexity depends on the step-size rule:
  —the constant step size: no computation,
  —ZLDS: one division.
  —variable target scheme with $\hat{f}^{*(i)}$ in (44): for the average magnitude case (p=1), N_w multiplications, $\Omega + N_w - 1$ additions, and two divisions.
- Compute the update, which requires N_w multiplications and N_w additions.

Overall, we need about $(\Omega + 1)N_w$ multiplications and additions (and additional computational load depending on the step-size selection scheme) per iteration. Therefore, the overall complexity is $(\Omega + 1)N_w\eta_{\mathrm{SGBA}}$ operations, where η_{SGBA} is the number of iterations. The linear programming solution of *Problem 3* requires $O(\Omega^3 L)$ operations, where L is a precision-based factor [40]. Therefore, considering that the number of iterations is typically much less than the window length, as justified by the examples in the next section, there is a big computational saving obtained by the SGBA algorithm compared to the linear programming solution.

The major computational burden of SGBA is due to the output computation that requires $\Omega N_w\eta_{\mathrm{SGBA}}$ operations, whereas the update part requires (assuming simple step-size selection rule) only $N_w\eta_{\mathrm{SGBA}}$ ($N_w^2\eta_{\mathrm{SGBA}}$ in weighted SGBA) operations. In many physical layer communication system implementations, the equalizer is designed to be a part of the hardware, and therefore, both input and output samples are available to the adaptive algorithm during the data collection stage at no additional cost. As a result, in such systems, unless there is a constraint on the reuse of the same input data window, we can drop the load of the output computations from the overall computation budget. The peak detection [and the magnitude averaging required for (44)] can be also done during the data-collection stage with simple additions. At the end of the data collection, the update can be performed, which requires only N_w (N_w^2 for weighted SGBA) operations per iteration.

If we compare the complexity SGBA with some other iterative methods' complexities, we get the following.

- The CMA algorithm [4], [5] requires $N_w\eta_{\mathrm{CMA}}$ operations. Here, $\eta_{\mathrm{CMA}} \gg \eta_{\mathrm{SGBA}}$ in general.
- The super-exponential algorithm by Shalvi–Weinstein [15] requires $\Omega N_w\eta_{\mathrm{SW}}$ operations for output computation, $6\Omega N_w\eta_{\mathrm{SW}}$ operations for the computation of the empirical cumulant expression, and $N_w^2\eta_{\mathrm{SW}}$ operations for the multiplication with the inverse covariance matrix. As illustrated in Section VI, η_{SW} is typically less than η_{SGBA}; however, the overall computational complexities of the two algorithms are similar.
- The Vembu algorithm [18] (based on l_p approximation of l_∞) requires $N_w\eta_{\mathrm{Vembu}}p$, and typically, $p\eta_{\mathrm{Vembu}} \gg \eta_{\mathrm{SGBA}}$.

When the output computation can be dropped from the computational budget by the hardware implementation of the equalizer, SGBA has significantly lower complexity than all of the algorithms above.

VI. Examples

In this section, we provide examples illustrating the performance of the SGBA algorithm for some sample channels.

In the first example, we compare the convergence performance of different step-size selection rules. We assume

- $h = \{-1.0493 + 0.2305i, 1.4129 - 1.4497i, -0.2540 + 0.2021i, 0.5302 - 0.7732i\}$ as the impulse response of the channel (taken from [41]);
- 4-QAM input constellation;
- $N_\omega = 21, L = 8, \Omega = 400$.

The step-size selection schemes we compared are the following.

1) *Relaxation*: $\mu^{(i)} = 1.7(|\Re e\{z_{l^{(i)}}^{(i)}\}| - f^*)/\|\mathbf{\Gamma}_{l^{(i)},:}\|_2^2$, where f^* is assumed to be known. Here, we precalculated f^* using the LP method. This is not reasonable in real-life applications; however, we include this case to provide a basis of comparison for the other step-size selection schemes.
2) *Relaxation with Average*: $\mu^{(i)} = 1.3((|\Re e\{z_{l^{(i)}}^{(i)}\}| - \hat{f}^{*(i)})/(\|\mathbf{\Gamma}_{l^{(i)},:}\|_2^2))$, where $\hat{f}^{*(i)}$ is selected as the average magnitude in (44).
3) *Normalized Constant*: $\mu^{(i)} = (0.013/(\|\mathbf{\Gamma}_{l^{(i)},:}\|_2))$.
4) ZLDS: $\mu^{(i)} = (10/((i + 1)\|\mathbf{\Gamma}_{l^{(i)},:}\|_2))$.
5) *Goffin–Kiwiel Algorithm*: We updated the target level based on the rules presented in Section IV, following the steps of the Goffin–Kiwiel algorithm in [38]. In our implementation, we chose the threshold for the total path length traversed for the given target level as 0.9 (R parameter in [38] and the initial distance between the minimum and the target level as 4 (δ_1 parameter in [38]).

The plots for the distance to the optimal point and the open-eye measures for the different step-size selection rules are shown in Figs. 3 and 4, respectively. Among different choices, the best performance is achieved by using the relaxation with perfect knowledge of f^* for which the eye becomes open after 40 to 60 iterations. When f^* is replaced with the average magnitude approximation, a similar eye-opening performance is achieved. However, the performance is worse after the eye-opening stage. The degradation is due to fact that only a finite time window is considered, and a symbol-spaced FIR

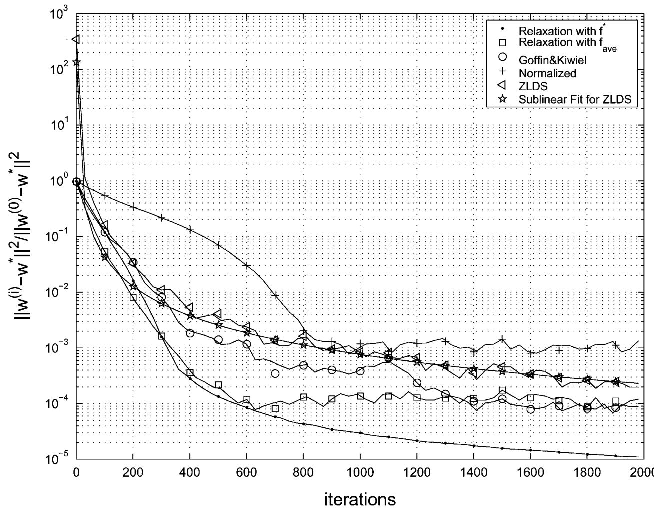

Fig. 3. Distance to the optimal point for different step-size selections.

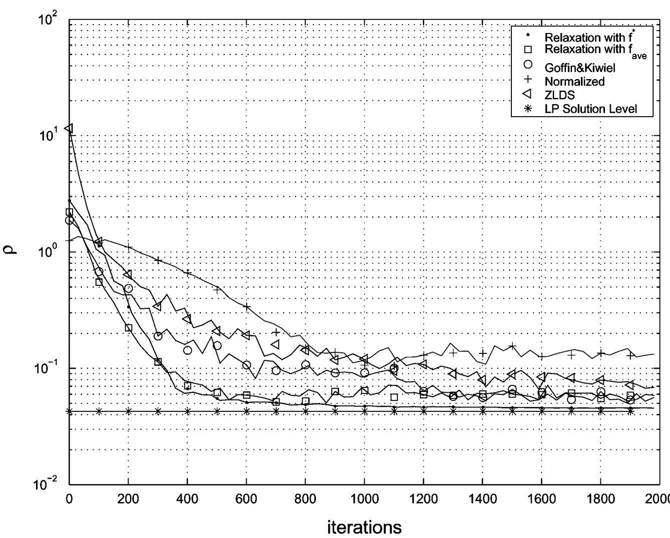

Fig. 4. Open-eye measure for different step-size selections.

equalizer is employed such that the perfect equalization condition is never achieved. If the SBGA algorithm is considered as the initial eye-opener algorithm, which opens the initially closed eye and then leaves the stage to the decision directed blind algorithms, then the relaxation with average magnitude step-size rule can replace the relaxation with f^* as their eye-opening performances are almost identical.

The ZLDS rule, although slower, provides a reasonably fast convergence. The expected sublinear convergence behavior of ZLDS (from Section IV-B) is also confirmed by Fig. 3. The normalized constant step-size rules lead to a slow convergence, as expected. The Goffin–Kiwiel algorithm, after careful tune of its parameters, provides fast convergence. However, the performance of this algorithm tends to be very sensitive to the selection of its parameters. For example, if the threshold parameter for the expression in (31) and the initial target value are not properly chosen, then the algorithm can become very slow. Similarly, we observed that the behavior of Nedic–Bertsekas algorithm is highly dependent on the β parameter in (33), and slight deviations in β cause drastic changes in the algorithm's convergence behavior.

As another example, we consider the same complex channel, and we compare the convergence of SGBA with the following algorithms.

- CMA [5] with constant step size: A high value of step size is chosen to increase the speed of convergence.
- Vembu Algorithm [18]: We have the l_p-based approximation of the l_∞ cost function. We used a dynamic step-size adjustment to increase the speed of convergence for the gradient search algorithm. We selected p parameter according to the following rule:

$$p = \begin{cases} 8, & l < 5000 \\ 12, & 5000 \leq l < 9000 \\ 50, & 9000 \leq l \end{cases} \tag{58}$$

  where l is the iteration index.
- Shalvi–Weinstein (super-exponential) Algorithm [15]. The window length is chosen as 400 (same as SGBA algorithm).

Fig. 5 summarizes the results of the simulations with all four algorithms, where each curve corresponds to a different initialization point. Initial points are chosen randomly from the surface of a sphere around the optimal point so that the initial distance to the optimal point is the same at each run. In SGBA simulations, we used the SGBA with the relaxation step-size rule rule with the average magnitude as the variable target value. The following are the observations based on this figure.

- Convergence to a global optimal point is obtained by SGBA and Vembu algorithms, whereas for the Shalvi–Weinstein and CMA algorithms, $\mathbf{w}$ can converge to different stationary points, depending on the initialization [17]. The ISI level obtained by the LP-based solution is -39.6 dB, which verifies that the point converged by the SGBA algorithm is close to the solution of *Problem 3*.
- When entering into a steady convergence path, the Shalvi–Weinstein algorithm provides fast convergence. However, depending on the initial point, the performance may drag at high ISI levels for a while (which is probably caused by the attraction of saddle points).
- SGBA converges at reasonably low number of iterations and is less sensitive to the choice of initial point due to the convex nature of its cost function.
- Both the CMA and Vembu algorithms are slow and require quite a large number of iterations.

As the next example, we consider a DSL channel, the combination of an ITU G.SHDSL Central Office transmitter mask and a transmission channel defined in standard test cases (CSA4 Channel with two bridge taps) [42]. The corresponding 128-tap impulse response and its frequency response are shown in Fig. 6. We also added white Gaussian noise with SNR level of 35 dB.

In this simulation, we assumed $M = 2, \Omega = 800, N_w = 30, L = 12$ and used the weighted SGBA (with average magnitude estimate of f^*). Fig. 7 summarizes the results.

- The weighted SGBA algorithm opens the eye in 50-to-100 iterations. In order to counteract the effect of noise on convergence, each subgradient vector used in an iteration

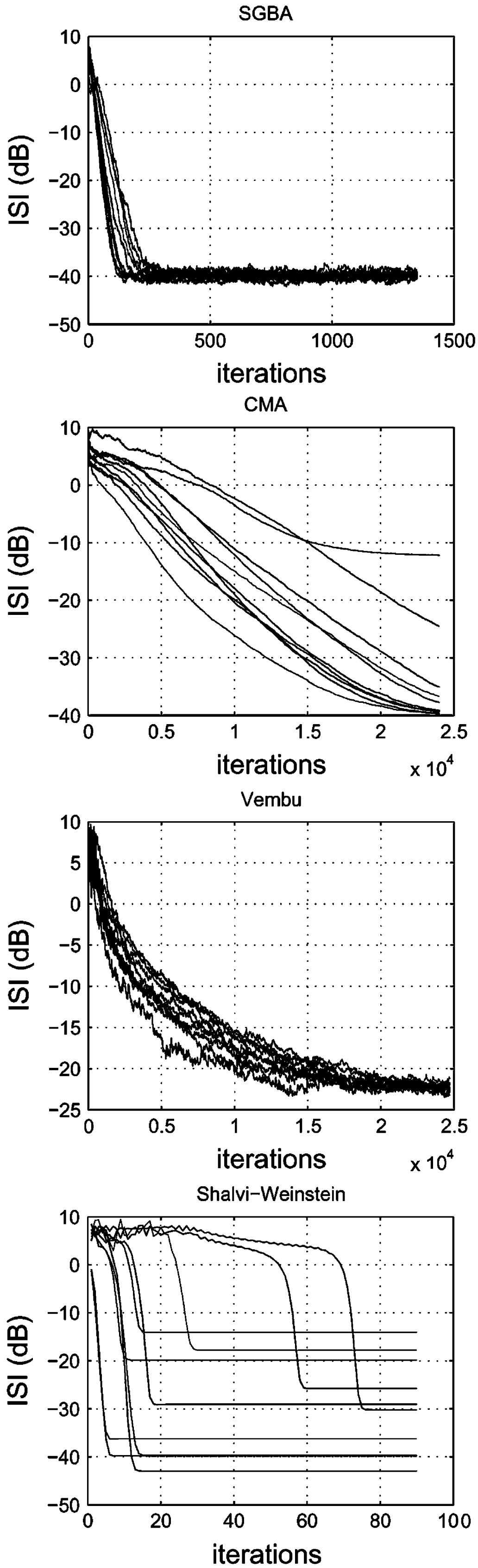

Fig. 5. ISI as a function of iterations for the four-tap complex channel.

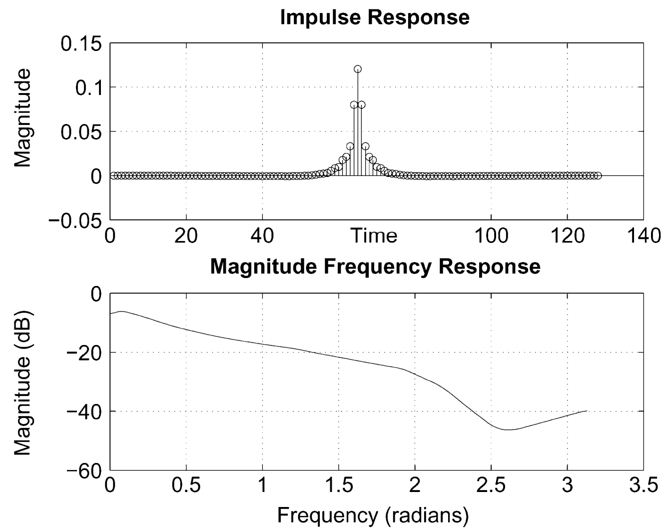

Fig. 6. Impulse response and the frequency response for a sample DSL channel.

is formed as a convex combination (average) of multiple subgradient vector estimates based on (38). These subgradient estimates are the input vectors (modulated by the sign of their corresponding outputs) at the time instants where the output magnitude is 96% of the peak magnitude.

- The super-exponential algorithm convergence curves have initial high ISI-drag sections followed by sharp falls.
- The CMA and Vembu Algorithms again require large numbers of iterations. Furthermore, the stationary point of the CMA algorithm with -10-dB ISI level corresponds to an undesired stationary point with two significant ISI taps (i.e., an undesired minima that is expected in a sample-spaced FIR equalization setting).
- The computations required for each algorithmn are as follows.
  - —SGBA: Assuming 75 iterations on average, the required computation is about 2.1 million operations (1.8 million for the output computations and 0.3 million for the updates). Note that if the output computation's complexity can be dropped (with a hardware equalizer implementation), then the computational requirement reduces to 0.3 million.
  - —Shalvi–Weinstein: Assuming 20 iterations on average, the required computation is about 3.3 million operations (0.5 million for the output computations and 2.8 million for the updates).
  - —CMA: Assuming 10^5 iterations, the required computation is 6 million operations.
  - —Vembu: The required computation is more than 20 million operations.
  - —LP solution of *Problem 3*: The required computation is on the order of 512 Million operations.

VII. Conclusion

We introduced a novel iterative blind equalization approach that is based on the iterative minimization of the l_∞ norm of the equalizer output using the subgradient optimization technique. SGBA updates are very simple and fast, especially for

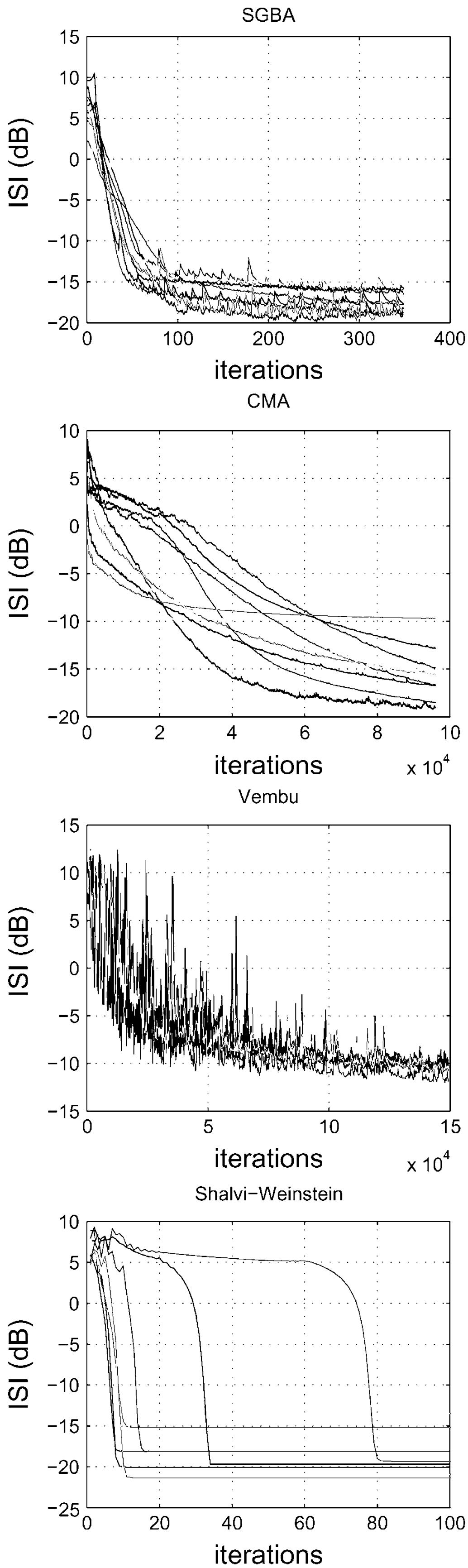

Fig. 7. ISI as a function of iterations for the sample DSL channel.

certain DSP systems in which the equalizer is implemented in hardware. As the computational requirement for the updates is minimal, the algorithm is expected to be insensitive against the roundoff noise effects in fixed-point implementations. Furthermore, due to the convex nature of the l_∞ cost function, the convergence problems that exist in CM-type algorithms caused by the saddle and undesirable equilibria points and arbitrary initialization points do not exist. Further improvement in the convergence rate can be achieved through proper selection of the step size, as illustrated by the examples in the previous section. Moreover, the number of data samples consumed to blindly train the equalizer is less than the other iterative blind equalization algorithms such as CMA. Finally, we should add that the iterative algorithm presented in this paper is an implicit higher order statistics method, and therefore, it is suitable for sample-spaced channels as well as fractionally spaced channels (or general single input multiple output channels).

ACKNOWLEDGMENT

The authors would like to thank to Dr. Sidiropoulos and the anonymous reviewers for their valuable comments, which contributed to the quality of this paper.

REFERENCES

[1] D. Hatzinakos and C. L. Nikias, "Blind equalization using a trispectrum based algorithm," *IEEE Trans. Commun.*, vol. 39, no. 5, pp. 669–681, May 1991.

[2] B. Porat and B. Friedlander, "Blind equalization of digital communications channels using higher order moments," in *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 39, Feb. 1991, pp. 522–526.

[3] J. Tugnait, "Blind equalization and estimation of FIR communications channels using fractional sampling," *IEEE Trans. Commun.*, vol. 44, no. 3, pp. 324–336, Mar. 1996.

[4] ——, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Trans. Commun.*, vol. COM-28, no. 11, pp. 1867–1875, Nov. 1980.

[5] C. Johnson, J. Schniter, T. Endres, J. Behm, R. Casas, V. Brown, and C. Berg, "Blind equalization using the constant modulus criterion: A review," in *Proc. IEEE, Special Issue on Blind System Identification and Estimation*, vol. 47, Jun. 1998, pp. 1927–1950.

[6] H. Liu, G. Xu, L. Tong, and T. Kailath, "Recent developments in blind channel equalization: From cyclostationarity to subspaces," *Signal Process.*, vol. 50, pp. 83–99, Apr. 1996.

[7] E. Moulines, P. Duhamel, J. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Trans. Signal Process.*, vol. 43, no. 2, pp. 516–525, Feb. 1995.

[8] W. Sethares, G. A. Rey, and C. Johnson, "Approaches to blind equalization of signals with multiple modulus," in *Proc. ICASSP*, vol. 2, 1989, pp. 972–975.

[9] C. Papadias and D. Slock, "On the decision-directed equalization of constant modulus signals," in *Proc. 28th Asilomar Conf. Signals, Syst., Comput.*, vol. 2, 1994, pp. 1423–1427.

[10] T.-H. Li, "Finite-alphabet information and multivariate blind deconvolution and identification of linear systems," *IEEE Trans. Inf. Theory*, vol. 49, no. 1, pp. 330–337, Jan. 2003.

[11] A. van der Veen, S. Talwar, and A. Paulraj, "Blind identification of FIR channels carrying multiple finite alphabet signals," in *Proc. ICASSP*, vol. 2, 1995, pp. 1213–1216.

[12] L. Tong and S. Perreau, "Multichannel blind identification: From subspace to maximum likelihood methods," *Proc. IEEE*, vol. 86, no. 10, pp. 1951–1968, Oct. 1998.

[13] Z. Ding and C. R. Johnson, "On the nonvanishing stability of undesirable equilibria for FIR Godard blind equalizers," *IEEE Trans. Signal Process.*, vol. 41, no. 5, pp. 1940–1944, May 1993.

[14] S. Lambotharan, J. Chambers, and C. R. Johnson, "Attraction of saddles and slow convergence in CMA adaptation," *Signal Process.*, vol. 59, pp. 335–340, Jun. 1997.

[15] O. Shalvi and E. Weinstein, "Super-exponential methods for blind deconvolution," *IEEE Trans. Inf. Theory*, vol. 39, no. 2, pp. 504–519, Mar. 1993.

[16] M. Mboup and P. Regalia, "A gradient search interpretation of the super-exponential algorithm," *IEEE Trans. Inf. Theory*, vol. 46, no. 6, pp. 2731–2734, Nov. 2000.
[17] P. Regalia and M. Mboup, "Undermodeled equalization: A characterization of stationary points for a family of blind criteria," *IEEE Trans. Signal Process.*, vol. 47, no. 3, pp. 760–770, Mar. 1999.
[18] S. Vembu, S. Verdu, R. Kennedy, and W. Sethares, "Convex cost functions in blind equalization," *IEEE Trans. Signal Process.*, vol. 42, no. 8, pp. 1952–1960, Aug. 1994.
[19] Z. Ding and Z. Luo, "A fast linear programming algorithm for blind equalization," in *IEEE Trans. Commun.*, vol. 48, Sep. 2000, pp. 1432–1436.
[20] Z.-Q. Luo, M. Meng, K. M. Wong, and J.-K. Zhang, "A fractionally spaced blind equalizer based on linear programming," *IEEE Trans. Signal Process.*, vol. 50, no. 7, pp. 1650–1660, Jul. 2002.
[21] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1999.
[22] J.-B. Hiriart-Urruty and C. Lemarechal, *Convex Analysis and Minimization Algorithms I*. New York: Springer-Verlag, 1996.
[23] O. Shalvi and E. Weinstein, "New criteria for blind deconvolution of nonminimum phase systems (channels)," *IEEE Trans. Inf. Theory*, vol. 36, no. 2, pp. 312–321, Mar. 1990.
[24] Y. Li and Z. Ding, "Global convergence of fractionally spaced Godard (cma) adaptive equalizers," *IEEE Trans. Signal Process.*, vol. 44, no. 4, pp. 818–826, Apr. 1996.
[25] G. A. Watson, "Approximation in normed linear spaces," *J. Comput. Applied Math.*, vol. 121, pp. 1–36, Sept. 2000.
[26] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 2002.
[27] A. H. Sayed, *Fundamentals of Adaptive Filtering*. New York: Wiley, 2003.
[28] B. T. Polyak, "A general method for solving extremal problems," *Doklady Akademii Nauk SSSR*, vol. 174, pp. 33–36, 1967.
[29] N. Shor, *Minimization Methods for Nondifferentiable Functions*. New York: Springer-Verlag, 1985.
[30] A. Nedic and D. Bertsekas, "Convergence rate of incremental subgradient algorithms," in *Stochastic Optimization: Algorithms and Applications*, S. Ursayev and P. M. Pardalos, Eds. Boston, MA: Kluwer, 2000.
[31] U. Brannlund, "On Relaxation Methods for Nonsmooth Convex Optimization," Ph.D. Thesis, Dept. Math., R. Inst. Technol., Stockholm, Sweden, 1993.
[32] J. Goffin, "On convergence rates of subgradient optimization methods," *Math. Programming*, vol. 13, pp. 329–347, May 1977.
[33] M. Held, P. Wolfe, and H. Crowder, "Validation of subgradient optimization," *Math. Programming*, vol. 6, pp. 62–88, Feb. 1974.
[34] B. Polyak, "Minimization of unsmooth functionals," *USSR Comput. Math. Phys.*, vol. 9, pp. 509–521, 1969.
[35] E. Allen, R. Helgason, J. Kennington, and B. Shetty, "A generalization of Polyak's convergence result for subgradient optimization," *Math. Programming*, vol. 37, pp. 309–317, May 1987.
[36] M. Bazaraa and H. Sherali, "On the choice of step size in subgradient optimization," *Eur. J. Oper. Res.*, vol. 7, pp. 380–388, Aug. 1981.
[37] S. Kim, H. Ahn, and S.-C. Cho, "Variable target value subgradient method," *Math. Programming*, vol. 49, pp. 359–369, Jan. 1991.
[38] J. Goffin and K. Kiwiel, "Convergence of a simple subgradient level method," *Math. Programming*, vol. 85, no. 1, pp. 207–211, 1999.
[39] H. Sherali, G. Choi, and C. Tuncbilek, "Variable target value method for nondifferentiable optimization," *Oper. Res. Lett.*, vol. 26, pp. 1–8, Feb. 2000.
[40] D. A. Spielman and S.-H. Teng, "Smoothed analysis of termination of linear programming algorithms," *Math. Programming*, vol. 97, pp. 375–404, Jul. 2003.
[41] C. Papadias, "Methods for Blind Equalization and Identification of Linear Channels," Ph.D. dissertation, EUROCOM Inst., Roissy en Brie, France, 1995.
[42] *G.shdsl: Symmetric High-Bitrate Digital Subscriber Line Standard*, 2001. ITU-T.

Alper Tunga Erdogan was born in Ankara, Turkey, in 1971. He received the B.S. degree from the Middle East Technical University, Ankara, and the M.S. and Ph.D. degrees from Stanford University, Stanford, CA all in electrical engineering, in 1993, 1995, and 1999, respectively.

He was a principal research engineer with Globespan-Virata Corporation (formerly Excess Bandwidth and Virata Corporations), Santa Clara, CA, from September 1999 to November 2001. He has been an assistant professor with the Electrical and Electronics Engineering Department, Koc University, Istanbul, Turkey, since January 2002. His research interests include wireless and wireline communications, adaptive signal processing, optimization, system theory and control, and information theory.

Can Kizilkale (S'03) was born in Ankara, Turkey, in 1981. He received the B.S. degree from the Department of Electrical and Electronics Engineering, Bilkent University, Ankara, in 2002. Currently, he is a graduate student with Koc University, Istanbul, Turkey, and is expecting to receive the M.S. degree in electrical and computer engineering in September 2004.

His research interests are adaptive blind equalization and convex optimization.