

# RELATIVE-GRADIENT BUSSGANG-TYPE BLIND EQUALIZATION ALGORITHMS

Zhengwei Wu, Saleem A. Kassam

Visa Koivunen

Dept. of Electrical and System Engineering,  
University of Pennsylvania, USA  
{zhengwei, kassam}@seas.upenn.edu

Dept. of Signal Processing and Acoustics,  
Aalto University, Finland  
visa.koivunen@aalto.fi

## ABSTRACT

In blind equalization (BE) a cost function based on the fit between the equalizer outputs and the signaling constellation is generally defined. To minimize such a cost function, standard gradient descent learning is commonly used. We exploit the idea of *relative gradient* (RG) learning to modify such standard Bussgang-type algorithms. Instead of one output each time, our method uses a sliding *block* of outputs. Our RG-based block Bussgang algorithms have faster convergence than corresponding Bussgang algorithms based on the standard gradient.

**Index Terms**— Relative gradient learning, Bussgang condition, blind equalization

## 1. INTRODUCTION

In blind equalization (BE) statistical or structural properties of payload data are used for finding the equalizer. BE techniques can be quite useful, for example in pilot decontamination in massive MIMO systems with time-division duplex (TDD) [1], [2]. Standard stochastic gradient descent is commonly used to minimize a cost function that is defined based on the fit of equalizer outputs to some known signaling constellation property. A number of Bussgang-type algorithms have been considered for BE, such as the *constant modulus algorithm* (CMA) [3], [4], [5], the *generalized Sato algorithm* (GSA) [6], [7], the *multimodulus algorithm* (MMA) [8] and the *square contour algorithm* (SCA) [9]. The vector CMA (VCMA) is a version of the CMA where a sliding block of equalizer outputs is processed each time [10], [11].

BE schemes have also been designed using blind source separation (BSS) algorithms [12], [13], [14], [15], for use when the source symbols are independent and identically distributed (i.i.d.). This is a reasonable assumption when source coding is used for payload data. These algorithms also process a block of equalizer outputs at each time. In [16] the BSS algorithm was based on the *relative gradient* (RG). Our recent RG-BSS based scheme for BE in [15] gives better performance relative to standard Bussgang-type BE algorithms for i.i.d. source symbols.

In this paper we propose to modify the Bussgang-type algorithms by using the RG instead of the standard gradient (SG) formulation. A block of outputs are used each time to update the matrix that contains the coefficients of the equalizer vector. Using the RG and forcing Toeplitz structure helps speed up convergence. Unlike BSS-based BE algorithms using the RG, independence of source symbols is not required for our RG Bussgang equalizers. Our proposed algorithms yield faster convergence compared to standard Bussgang-type BE algorithms.

## 2. BLIND EQUALIZATION MODEL AND BUSSGANG-TYPE ALGORITHMS

### 2.1. Blind Equalization

Consider a complex symbol sequence  $\{s(k)\}$  transmitted through an FIR complex channel. For symbol rate sampling, the output of the channel at time  $k$  can be expressed as

$$x(k) = \sum_{l=0}^L h(l)s(k-l) + v(k), \quad (1)$$

where  $\mathbf{h} = [h(0), h(1), \dots, h(L)]^T$  is the channel response, and  $\{v(k)\}$  is an additive white Gaussian noise sequence. The input source sequence is generally but not necessarily i.i.d.

An  $M$ th-order FIR equalizer with impulse response  $\mathbf{w} = [w(0), w(1), \dots, w(M)]^T$  is to be designed so that its output  $y(k) = \sum_{m=0}^M w(m)x(k-m)$  approximates input  $s(k)$  to within a fixed delay  $d$  and possibly a phase shift.

### 2.2. Bussgang-Type Algorithms

In conventional equalization, a popular cost function is the mean square error  $J(\mathbf{w}) = E[|y(k) - s(k-d)|^2]$ . Using *stochastic gradient descent*, the equalizer adaptation is given as

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mu(y(k) - s(k-d))\mathbf{x}_k^*, \quad (2)$$

where  $\mathbf{x}_k = [x(k), x(k-1), \dots, x(k-M)]^T$  is a vector of channel outputs at time  $k$ , of length  $M+1$ . Equation (2) is the well-known least mean square (LMS) update.

The Bussgang technique was first proposed in [17]. With no training symbol  $s(k-d)$  in (2), a memoryless estimator  $\varphi(y(k))$  is used in place of  $s(k-d)$ . The adaptation (2) then becomes

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mu(y(k) - \varphi(y(k)))\mathbf{x}_k^* \quad (3)$$

From (3), we have for the  $i$ -th equalizer coefficient,

$$w_{k+1}(i) = w_k(i) - \mu(y(k) - \varphi(y(k)))x(k-i)^*, 0 \leq i \leq M. \quad (4)$$

The expected value of any equalizer coefficient should tend to a constant under convergence. As a result, we have

$$E[y(k)x(k-i)^*] = E[\varphi(y(k))x(k-i)^*]. \quad (5)$$

For a doubly-infinite equalizer it is easy to see that [18]

$$E[y(k)y(k-m)^*] = E[\varphi(y(k))y(k-m)^*]. \quad (6)$$

A process  $\{y(k)\}$  is called a *Bussgang process* if it satisfies the condition (6), and the algorithm (3) is therefore called a *Bussgang algorithm*. If  $M$  for our FIR equalizer is not too small the Bussgang condition should be well approximated.

In specific *Bussgang-type* algorithms such as the CMA, the GSA, the MMA and the SCA, the cost function can be written as  $J(\mathbf{w}) = E[G(y(k))]$ , and the equalizer adaptation has the form

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mu g(y(k))\mathbf{x}_k^* \quad (7)$$

where  $g(y(k)) = \partial G(y(k)) / \partial y(k)^*$ . Here the nonlinear “estimator” is  $\varphi(y(k)) = y(k) - g(y(k))$ . For example,  $\varphi(y(k))$  for the GSA is  $\varphi(y(k)) = R_{\text{GSA}} \text{csgn}(y(k))$ , where  $R_{\text{GSA}}$  is a scaling constant, and  $\text{csgn}(c) = \text{sgn}(a) + j\text{sgn}(b)$  for  $c = a + bj$ . Upon convergence, the Bussgang condition of (6) is equivalent to

$$E[g(y(k))y(k-m)^*] = 0. \quad (8)$$

### 3. BLOCK BUSSGANG ALGORITHMS WITH STANDARD GRADIENT

In the standard Bussgang-type algorithms, the equalizer is applied to its input vector  $\mathbf{x}_k$  that contains the current and past  $M$  channel outputs. Suppose we apply at each time  $k$  the equalizer  $\mathbf{w}_k$  to a larger block of channel outputs  $\tilde{\mathbf{x}}_k = [x(k), x(k-1), \dots, x(k-M+1), \dots, x(k-P-M+1)]^T$  of size  $P+M$ . The channel is assumed to be quasi-stationary over this observation period. The equalizer  $\mathbf{w}_k$  convolved with this block will produce  $P$  outputs. These  $P$  equalized outputs and the corresponding equalizer input sub-vectors in  $\tilde{\mathbf{x}}_k$  can be used to generate an averaged version of the update term in (7), i.e.

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \frac{\mu}{P} \sum_{i=0}^{P-1} g(\tilde{y}^{(k)}(k-i))\mathbf{x}_{k-i}^* \quad (9)$$

where  $\mathbf{x}_{k-i} = [x(k-i), x(k-i-1), \dots, x(k-i-M)]$  are the length- $(M+1)$  vectors contained in  $\tilde{\mathbf{x}}_k$ , and  $\tilde{y}^{(k)}(k-i)$  is the equalizer output using the *current* equalizer, i.e.  $\tilde{y}^{(k)}(k-i) = \mathbf{w}_k^T \mathbf{x}_{k-i}$ . Note that  $y(k) = \tilde{y}^{(k)}(k)$ . Equation (9) can be considered to be a general block version of the standard Bussgang algorithm.

Define a  $P \times (P+M)$  Toeplitz “equalizer matrix” containing the equalizer coefficient vector as follows:

$$\mathbf{W} = \begin{pmatrix} w(0) & w(1) & \dots & w(M) & \dots & \dots & 0 \\ & \ddots & & \dots & \ddots & & \\ \dots & w(0) & w(1) & \dots & w(M) & \dots & \\ & \ddots & & \dots & \ddots & & \\ 0 & \dots & \dots & w(0) & w(1) & \dots & w(M) \end{pmatrix} \quad (10)$$

The adaptive equalizer coefficient vector update of (9) can be written in matrix form with two steps at each iteration:

$$\begin{aligned} \hat{\mathbf{W}}_{k+1} &= \mathbf{W}_k - \mu \mathbf{g}(\tilde{\mathbf{y}}_k) \tilde{\mathbf{x}}_k^H, \\ \mathbf{W}_{k+1} &= \text{Toeplitz}\{\hat{\mathbf{W}}_{k+1}\}. \end{aligned} \quad (11)$$

Here  $\tilde{\mathbf{y}}_k = [y(k), \tilde{y}^{(k)}(k-1), \dots, \tilde{y}^{(k)}(k-P+1)]^T$  is the block of  $P$  outputs from the current equalizer, and  $\mathbf{g}(\tilde{\mathbf{y}}_k) = [g(y(k)), g(\tilde{y}^{(k)}(k-1)), \dots, g(\tilde{y}^{(k)}(k-P+1))]^T$ ; in the second line of (11) the Toeplitz structure of (10) is forced on  $\hat{\mathbf{W}}_{k+1}$  by taking averages along the descending diagonals of  $\hat{\mathbf{W}}_{k+1}$  after forcing the  $P-1$  diagonals on the upper right and lower left corners of  $\hat{\mathbf{W}}_{k+1}$  to be zero. The resulting  $\mathbf{W}_{k+1}$  matrix contains the updated equalizer coefficients in each row.

It can be shown that the first step of (11) is the standard stochastic gradient descent method to minimize the following cost function of the equalizer matrix  $\mathbf{W}$ :

$$\begin{aligned} L(\mathbf{W}) &= \mathbf{1}_{1 \times P} E[\mathbf{G}(\mathbf{W}\tilde{\mathbf{x}}_k)] = \mathbf{1}_{1 \times P} E[\mathbf{G}(\tilde{\mathbf{y}}_k)] \\ &= \sum_{i=0}^{P-1} E[G(\tilde{y}^{(k)}(k-i))], \end{aligned} \quad (12)$$

where  $\mathbf{G}(\tilde{\mathbf{y}}_k)$  is the vector with function  $G(\bullet)$  applied component-wise on the  $\tilde{\mathbf{y}}_k$ . According to methods for functions of a complex matrix [19], the gradient (SG) of  $L(\mathbf{W})$  can be calculated exactly as

$$\nabla_{\mathbf{w}} L(\mathbf{W}) = E[\mathbf{g}(\tilde{\mathbf{y}}_k) \tilde{\mathbf{x}}_k^H], \quad (13)$$

and the standard stochastic gradient descent adaptation for  $\mathbf{W}$  becomes exactly the same as the first step of (11).

Assuming that channel coherence time is long enough to allow size- $P$  observation blocks, we found from simulation that there is not much difference between standard Bussgang algorithms with  $P=1$  and their corresponding block versions with  $P>1$ ; however, this block structure

turns out to be effective when another gradient, the relative gradient, is used instead in stochastic gradient descent.

#### 4. BLOCK BUSSGANG ALGORITHMS WITH RELATIVE GRADIENT

##### 4.1. Relative Gradient

To minimize a scalar-valued cost function  $L(\mathbf{W})$ , instead of searching over all small deviations  $\delta\mathbf{W}$  of fixed norm Cardoso [16] considered small changes  $\delta\mathbf{W} = \mathcal{E}\mathbf{W}$  proportional to the current  $\mathbf{W}$ , with  $\mathcal{E}$  a “small” matrix. Therefore  $\mathcal{E}$  is a measure of change  $\delta\mathbf{W}$  relative to  $\mathbf{W}$ . With  $\delta\mathbf{W} = \mathcal{E}\mathbf{W}$ , the Taylor expansion of  $L(\mathbf{W})$  is [16], [19]

$$\begin{aligned} L(\mathbf{W} + \mathcal{E}\mathbf{W}) &= L(\mathbf{W}) + 2\text{Re}\langle \mathcal{E}\mathbf{W}, \nabla_{\mathbf{W}} L(\mathbf{W}) \rangle + o(\mathcal{E}) \\ &= L(\mathbf{W}) + 2\text{Re}\langle \mathcal{E}, \nabla_{\mathbf{W}} L(\mathbf{W})\mathbf{W}^H \rangle + o(\mathcal{E}). \end{aligned} \quad (14)$$

where  $\langle \mathbf{M}, \mathbf{N} \rangle = \text{Trace}\{\mathbf{N}^H \mathbf{M}\}$ . From this we find that if  $\mathcal{E}$  is in the direction opposite to  $\nabla_{\mathbf{W}} L(\mathbf{W})\mathbf{W}^H$ , the descent rate of  $L(\mathbf{W})$  is largest. This leads to the definition of the *relative gradient* (RG) for this matrix case as

$$\nabla_{\mathbf{W}}^{(R)} L(\mathbf{W}) = \nabla_{\mathbf{W}} L(\mathbf{W})\mathbf{W}^H. \quad (15)$$

Using the RG of (15), the gradient descent updates for matrix  $\mathbf{W}$  become

$$\begin{aligned} \mathbf{W}_{k+1} &= \mathbf{W}_k - \mu \nabla_{\mathbf{W}}^{(R)} L(\mathbf{W}_k) \mathbf{W}_k \\ &= \mathbf{W}_k - \mu \nabla_{\mathbf{W}} L(\mathbf{W}_k) \mathbf{W}_k^H \mathbf{W}_k. \end{aligned}$$

The concept of RG is closely related to the *natural gradient* (NG) in [20], [21], where the gradient direction depends on the local Riemannian structure of parameter space and the small change is in the Euclidean tangent plane. In the context of BSS with the parameter space of invertible matrices, both the RG and NG give the same matrix updates. The RG provides an alternative to the standard gradient for gradient descent and is easy to apply, and we know from [15] that RG BSS yields good BE performance. Given these considerations, we will now proceed to explore use of the RG in block versions of the Bussgang algorithms.

##### 4.2. Block RG Equalizer Adaptation

For the cost function  $L(\mathbf{W})$  in (12) of the equalizer matrix, the derivation of the RG depends on how we define the small relative change  $\delta\mathbf{W}$  which is proportional to  $\mathbf{W}$ . We may define  $\delta\mathbf{W} = \mathcal{E}\mathbf{W}$  or  $\delta\mathbf{W} = \mathbf{W}\mathcal{E}$  where  $\mathcal{E}$  is a square matrix. We found from our simulations that the first case gives better performance, thus here we only consider the perturbations written as  $\delta\mathbf{W} = \mathcal{E}\mathbf{W}$  to obtain the RG in this block version of our problem. In BSS, defining relative change this way also leads to a performance equivariance property [16], i.e. it is independent of the channel.

For  $\delta\mathbf{W} = \mathcal{E}\mathbf{W}$ , using the SG  $\nabla_{\mathbf{W}} L(\mathbf{W})$  of (13), the Taylor expansion of (14) becomes

$$\begin{aligned} L(\mathbf{W} + \mathcal{E}\mathbf{W}) &= L(\mathbf{W}) + 2\text{Re}\langle \mathcal{E}\mathbf{W}, E\mathbf{g}(\tilde{\mathbf{y}})\tilde{\mathbf{x}}^H \rangle + o(\mathcal{E}) \\ &= L(\mathbf{W}) + 2\text{Re}\langle \mathcal{E}, E\mathbf{g}(\tilde{\mathbf{y}})\tilde{\mathbf{y}}^H \rangle + o(\mathcal{E}). \end{aligned} \quad (16)$$

As a result, when  $\mathcal{E}$  is aligned with  $E[\mathbf{g}(\tilde{\mathbf{y}})\tilde{\mathbf{y}}^H]$ , the change rate is maximum, and the RG for a pre-multiplied matrix change can be written as

$$\nabla_{\mathbf{W}}^{(R)} L(\mathbf{W}) = E[\mathbf{g}(\tilde{\mathbf{y}})\tilde{\mathbf{y}}^H] = E[\mathbf{g}(\tilde{\mathbf{y}})\tilde{\mathbf{x}}^H \mathbf{W}^H]. \quad (17)$$

Comparing (17) with the SG in (13), we see that the RG is just the SG multiplied by  $\mathbf{W}^H$  on the right, as in (15).

The stochastic relative gradient descent adaptation for matrix  $\mathbf{W}$  may be stated as

$$\begin{aligned} \hat{\mathbf{W}}_{k+1} &= \mathbf{W}_k - \mu \mathbf{g}(\tilde{\mathbf{y}}_k) \tilde{\mathbf{y}}_k^H \mathbf{W}_k, \\ \mathbf{W}_{k+1} &= \text{Toepplitz}_z\{\hat{\mathbf{W}}_{k+1}\}. \end{aligned} \quad (18)$$

Since  $\mathbf{W}_k$  is Toeplitz with the equalizer vector in each row, we can show that (18) has the computationally more efficient *equivalent* form

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \frac{\mu}{P} \Psi_k \mathbf{w}_k, \quad (19)$$

where  $\Psi_k$  is an  $(M+1) \times (M+1)$  matrix containing cross-correlation terms from elements of  $\tilde{\mathbf{y}}_k$  and  $\mathbf{g}(\tilde{\mathbf{y}}_k)$ .

For our block RG Bussgang-type algorithm of (18), a stationary point is any matrix  $\mathbf{W}_k$  such that  $E[\mathbf{g}(\tilde{\mathbf{y}}_k)\tilde{\mathbf{y}}_k^H] = \mathbf{0}$  holds. Compared with (11) that uses the SG, the RG  $\mathbf{g}(\tilde{\mathbf{y}}_k)\tilde{\mathbf{y}}_k^H$  that pre-multiplies  $\mathbf{W}_k$  in (18) contains the cross-correlation terms of the Bussgang condition in (8), and therefore we may say that deviation from the Bussgang condition is taken into consideration in the updates of (18). When  $\mathbf{g}(\tilde{\mathbf{y}}_k)\tilde{\mathbf{y}}_k^H$  is large, the adaptation is far from the steady state, and the relative change to  $\mathbf{W}$  is large as a result; while when  $\mathbf{g}(\tilde{\mathbf{y}}_k)\tilde{\mathbf{y}}_k^H$  is small the value of  $\mathbf{W}$  is adjusted with small relative change at each iteration.

The RG  $\mathbf{g}(\tilde{\mathbf{y}}_k)\tilde{\mathbf{y}}_k^H$  contains the cross-correlation terms with time lag up to  $P-1$ . The larger  $P$  is, the more information is used to update matrix  $\mathbf{W}$  at each iteration. However, the larger  $P$  is, the more difficult it is to converge to the stationary point  $E[\mathbf{g}(\tilde{\mathbf{y}}_k)\tilde{\mathbf{y}}_k^H] = \mathbf{0}$ . As a result, the parameter  $P$  needs to be selected carefully to balance good performance and fast convergence. Moreover, channel coherence time needs to be taken into account. From our simulation studies we have found that a reasonable choice is  $P \approx M/2$ . Output phase ambiguity is to be expected for the block RG CMA because it occurs in the CMA. However, with the block RG GSA, MMA or SCA phase ambiguity is reduced to a multiple of  $\pi/2$  that can be easily resolved.

## 5. SIMULATIONS

We now give representative simulation results illustrating the relative performance of the block RG CMA and the standard CMA, as a special case of use of the RG in Busgang BE. Here the  $g$  function in (7) is  $g(y(k)) = (|y(k)|^2 - R_{\text{CMA}})y(k)$  with  $R_{\text{CMA}}$  a scaling constant. Similar results have been obtained for block RG versions of other standard Busgang schemes.

Consider a sequence of 64-QAM i.i.d. source symbols transmitted through a non-minimum phase order-4 FIR channel with SNR = 15dB as an example. The channel has response  $\mathbf{h} = [0.28, 0.9816-0.1911j, 0.5756+0.2451j, 0.3344-0.1385j, 0.1889+0.0625j, 0.0825]$  with one zero outside the unit circle. The equalizer is set to have order  $M = 25$ , and is initialized with center tap  $1+0.5j$  and zero elsewhere. Results for four choices of the size of the equalizer output block are shown. We use *inter-symbol interference* (ISI) in dB to measure the performance of the algo-

rithm, where ISI is defined as  $\text{ISI} = \sum_i \frac{|c_i|^2}{\max_i |c_i|^2} - 1$  for

the cascaded system  $\mathbf{c} = \mathbf{h} * \mathbf{w}$ . The ISI curves in Figs. 1 and 2 are averages over 5 runs.

For Fig. 1 the step-sizes were chosen to make the ISI after convergence approximately the same for all schemes. We see that when  $P = 5$ , the performance of the block RG CMA is worse than that of the standard CMA since not enough information is provided by  $\mathbf{g}(\tilde{\mathbf{y}}_k)\tilde{\mathbf{y}}_k^H$  to update  $\mathbf{W}$  in (18). On the other hand for large  $P$  convergence becomes slow. Based on simulations with different  $P$ , it turns out that  $P = 12$  gives the best performance. When  $P = 12$ , it took about  $2 \times 10^4$  symbols for the block RG CMA to converge, while the standard CMA needed  $7 \times 10^4$ . When  $P$  increases further, for example when  $P = 20$  and  $P = 40$ , the convergence is slower. There is not much difference between  $P = 20$  and  $P = 40$ . Experiments with other channels and different choices for  $P$  and  $M$  also indicated that  $P \approx M/2$  is a good choice. The ISI curve for the adaptation of (18) without the second step for Toeplitz structure is also shown in Fig. 1, for  $P = 12$ . From the figure it can be seen that the Toeplitz structure constraint is important, and helps speed up convergence.

Next, we will see how different choices of equalizer order  $M$  (with  $P = M/2$ ) affects performance. Fig. 2 shows that when  $M = 8$  the block RG CMA performs worse than the standard CMA. There is not much difference between the standard CMA and the block RG CMA when  $M = 15$ ; while when  $M = 20$  and  $M = 25$ , the convergence of the block RG CMA is faster. We know that for the Busgang condition to be well approximated, the equalizer should be long enough. As  $M$  increases, the advantage of using the RG becomes more apparent. In practice, the order  $M$  of the equalizer depends on the nature of the channel. Some

knowledge of channel length and zero locations can be used to set a reasonable value for  $M$ , and we can expect that our block RG CMA will be more likely to perform better than the standard CMA.

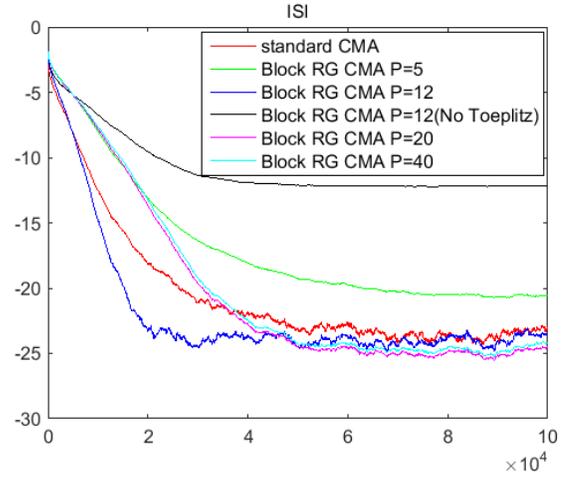


Fig. 1 ISI for CMA and block RG CMA with different  $P$ . 64-QAM i.i.d. source, SNR = 15dB,  $M = 25$ .

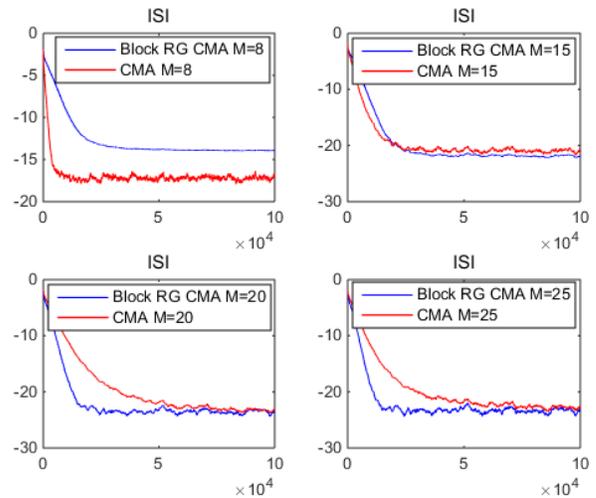


Fig. 2 ISI for CMA and block RG CMA for different equalizer order  $M$  ( $P = M/2$ ).

## 6. CONCLUSIONS

Our new block RG Busgang algorithms use a block of equalizer outputs at each iteration and enforce a Toeplitz condition for faster convergence. With the RG, the Busgang condition appears more explicitly in the equalizer adaptation steps. Simulation results suggest that the block RG Busgang algorithms offer faster convergence compared to their standard counterparts. While the block algorithms have a somewhat higher computational cost, our results suggest that the performance gains obtained are significant.

## 7. REFERENCES

- [1] A. Farhang, A. Aminjavaheri, N. Marchetti, L. E. Doyle, and B. Farhang-Boroujeny, "Pilot Decontamination in CMT-based Massive MIMO Networks," in *Wireless Communications Systems (ISWCS), 2014 11th International Symposium on*, 2014, pp. 589–593.
- [2] A. Aminjavaheri, A. Farhang, N. Marchetti, L. E. Doyle, and B. Farhang-Boroujeny, "Frequency Spreading Equalization in Multicarrier Massive MIMO," arXiv Prepr. arXiv1504.01140, 2015.
- [3] J. Treichler and B. G. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Trans. Acoust.*, vol. 31, no. 2, pp. 459–472, 1983.
- [4] J. R. Treichler and M. G. Larimore, "New Processing Techniques Based on the Constant Modulus Adaptive Algorithm," *IEEE Trans. Acoust.*, vol. 33, no. 2, pp. 420–431, 1985.
- [5] R. Johnson and P. Schniter, "Blind equalization using the constant modulus criterion: A review," *Proc. IEEE*, vol. 86, no. 10, pp. 1927–1950, 1998.
- [6] Y. Sato, "A Method of Self-Recovering Equalization for Multilevel Amplitude-Modulation Systems," *IEEE Trans. Commun.*, vol. 23, no. 6, pp. 679–682, 1975.
- [7] A. Benveniste and M. Goursat, "Blind Equalizers," *IEEE Trans. Commun.*, vol. 32, no. 8, pp. 871–879, 1984.
- [8] J. Yang, J. J. Werner, G. A. Dumont, and Others, "The multimodulus blind equalization and its generalized algorithms," *IEEE J. Sel. Areas Commun.*, vol. 20, pp. 997–1015, 2002.
- [9] T. Thaiupathump, L. He and S. A. Kassam, "Square contour algorithm for blind equalization of QAM signals," *Signal Processing*, vol. 86, pp. 3357–3370, 2006.
- [10] V. Y. Yang and D. L. Jones, "A vector constant modulus algorithm for shaped constellation equalization," *IEEE Signal Process. Lett.*, vol. 5, no. 4, pp. 89–91, 1998.
- [11] T. Abrudan and V. Koivunen, "Blind equalization in spatial multiplexing MIMO-OFDM systems based on vector CMA and decorrelation criteria," *Wirel. Pers. Commun.*, vol. 43, no. 4, pp. 1151–1172, 2007.
- [12] H. H. Yang, "On-line blind equalization via on-line blind separation," *Signal Processing*, vol. 68, no. 3, pp. 271–281, 1998.
- [13] Y. Zhang and S. A. Kassam, "Blind separation and equalization using fractional sampling of digital communications signals," *Signal Processing*, vol. 81, no. 12, pp. 2591–2608, 2001.
- [14] Z. Wu and S. A. Kassam, "Blind Equalization Based On Blind Separation with Toeplitz Constraint," in *Signals, Systems and Computers, 2014 48th Asilomar Conference on*, 2014, pp. 1453–1457.
- [15] Z. Wu and S. A. Kassam, "Symbol-Rate Blind Equalization based on Constrained Blind Separation," in *Information Sciences and Systems (CISS), 2015 49th Annual Conference on*, 2015, no. 4, pp. 1–6.
- [16] J. F. Cardoso and B. H. Laheld, "Equivariant adaptive source separation," *IEEE Trans. Signal Process.*, vol. 44, no. 12, pp. 3017–3030, 1996.
- [17] S. Bellini, "Busgang techniques for blind equalization," in *IEEE Global Telecommunications Conference*, 1986, pp. 1634–1640.
- [18] Z. Ding and Y. Li, *Blind Equalization and Identification*. CRC press, 2001.
- [19] T. Adali and S. Haykin, *Adaptive signal processing: next generation solutions*. John Wiley & Sons, 2010.
- [20] S. Amari, "Natural Gradient Works Efficiently in Learning," *Neural Comput.*, vol. 10, no. 2, pp. 251–276, 1998.
- [21] S. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang, "Multichannel Blind Deconvolution and Equalization Using the Natural Gradient," in *Signal Processing Advances in Wireless Communications, First IEEE Signal Processing Workshop on*, 1997, pp. 101–104.