SUBSTITUTING THE CUMULANTS IN THE SUPER-EXPONENTIAL BLIND EQUALIZATION ALGORITHM

Arie Yeredor

School of Electrical Engineering
Tel-Aviv University, Tel-Aviv, 69978, ISRAEL
e-mail: `arie@eng.tau.ac.il`

ABSTRACT

The Shalvi-Weinstein super-exponential algorithm for blind channel equalization employs empirical high-order cross-cumulants between the equalizer's input and output for iterative updates of the equalizer. When the source signal has (nearly) null cumulants of the required order, the algorithm's performance may be severely degraded. Rather than resort to even higher-order cumulants in such cases, we propose to employ an alternative statistic, based on second-order derivatives (Hessians, evaluated away from the origin) of the joint log-characteristic function of the equalizer's input and output. These Hessians admit straightforward empirical estimates, maintain the "philosophy of operation" of the algorithm, and, as we demonstrate in simulation, can significantly improve its performance in such (and in other) cases.

***Index Terms*—** super-exponential, blind equalization, characteristic function, Hessian, charrelation matrix.

1. INTRODUCTION

The super-exponential (SupEx) algorithm (Shalvi and Weinstein, [4]) for blind channel equalization is a powerful, rather popular tool for fast iterative construction of finite equalizers for unknown (possibly nonminimum-phase) linear, time-invariant (LTI) channels driven by an input of independent, identically distributed (iid) samples (with an unknown distribution). Based on empirical high-order joint cumulants of the equalizer's output and input, the equalizer's taps are modified iteratively, until convergence is attained, usually within a very small number (4-5) of iterations. It has been shown ([4, 5]) that under some regularity conditions, and barring effects of the finite equalizer length, the algorithm is free of local stationary points. Thus, if the relevant cumulants of the channel's input are finite and non-zero, and the equalizer is long enough, then asymptotically (as the observation length tends to infinity), SupEx converges to the exact equalizer (up to irrelevant delay and scaling).

However, if the source signal's cumulants of the required order are null, the algorithm is unable to produce any useful equalizer. Moreover, under non-asymptotic conditions, errors in the estimates of the cumulants naturally affect the performance of the equalizer, and this effect is more adverse for sources with a relatively small (nearly-vanishing) cumulant of the required order.

In such cases it might therefore be desirable to resort to some other statistics, whose estimates can be more informative for the construction of the equalizer. For example, cumulants of an alternative, higher order are a natural choice for the super-exponential algorithm, maintaining its general structure and philosophy. Indeed, if the sources are not Gaussian, they must have non-zero cumulants of some order (higher than second). For symmetrically-distributed sources, all odd-order cumulants vanish, so if, for example, the fourth-order cumulant is zero, it is possible to try to work with the sixth-order cumulant, and so forth. And yet, the estimation of cumulants of very high orders (higher than, say, six) is usually not desirable, since such estimates often (but not always) have a relatively large variance and, in addition, are somewhat cumbersome to construct.

In this paper we propose an alternative statistic as a substitute to cumulants. Our statistic is based on estimates of the Hessian (second-order derivative matrix) of the log characteristic function (LCF) of the joint distribution of the equalizer's input and output. Despite this apparently complex terminology, this statistic turns out to admit very straightforward, intuitively appealing empirical estimates, and, moreover, it maintains the "philosophy of operation" of SupEx. Using simulation results, we demonstrate the potential performance improvement resulting from the use of our alternative statistics for SupEx, in lieu of cumulants.

2. OFF-ORIGIN HESSIANS OF THE LCF

Off-origin Hessians of the LCF are a relatively new emerging tool, offering a "hybrid" statistic, conceptually reconciling second-order statistics with (classical) higher-order statistics. While high-order joint cumulants of any random vector are high-order derivatives of its LCF at the origin, one may consider remaining at the comfortable second-order differentiation, but moving away from the origin. Just like the second-order derivatives (Hessians) at the origin, which are simply the covariance matrices (or correlation matrices for zero-mean vectors), off-origin Hessians maintain the convenient form of matrices, rather than the form of tensors (multi-way arrays, which represent higher-order derivatives). In the sequel we shall refer to these matrices as "charrelation" (pronounced "car-relation") matrices (substituting correlation matrices), reflecting their link to the characteristic function.

The use of off-origin derivatives (of arbitrary order) of the LCF seems to have been first proposed by Gürelli and Nikias in [2] in the context of various array-processing applications, but has not been further pursued by these authors in open literature since. More recently, the use of second-order off-origin derivatives has been proposed by Yeredor *et al.* in various contexts, such as blind source separation [6], Direction of Arrival (DOA) estimation [8] and autoregressive (AR) parameters estimation [9]. Off-origin derivatives have also been used by Kawanabe and Theis [3] and by Comon and Rajih [1].

Let the $K \times 1$ vectors $\boldsymbol{x}$ and $\boldsymbol{\tau}$ denote (respectively) some random vector and an arbitrary (deterministic) vector. We shall refer to $\boldsymbol{\tau}$ as a "processing point". The (generalized) characteristic function (CF) and the LCF are defined, respectively, as

$$\phi_{\boldsymbol{x}}(\boldsymbol{\tau}) \triangleq E[e^{\boldsymbol{\tau}^T\boldsymbol{x}}] \quad , \quad \psi_{\boldsymbol{x}}(\boldsymbol{\tau}) \triangleq \log(\phi_{\boldsymbol{x}}(\boldsymbol{\tau})), \qquad (1)$$

whenever the mean exists. The $K \times 1$ gradients and $K \times K$ Hessian of $\psi_{\boldsymbol{x}}(\boldsymbol{\tau})$ are defined, respectively, as:

$$\boldsymbol{\psi}_{\boldsymbol{x}}(\boldsymbol{\tau}) \triangleq \left.\frac{\partial^T \psi_{\boldsymbol{x}}(\boldsymbol{\tau})}{\partial \boldsymbol{\tau}}\right|_{\boldsymbol{\tau}} \quad , \quad \boldsymbol{\Psi}_{\boldsymbol{x}}(\boldsymbol{\tau}) \triangleq \left.\frac{\partial^2 \psi_{\boldsymbol{x}}(\boldsymbol{\tau})}{\partial \boldsymbol{\tau}^2}\right|_{\boldsymbol{\tau}}. \qquad (2)$$

The Hessian $\boldsymbol{\Psi}_{\boldsymbol{x}}(\boldsymbol{\tau})$ will serve as our alternative "charrelation" matrix. The following general properties (see, e.g., [9] for proof) would be useful in our derivations:

Property 1. *If $\boldsymbol{x}$ can be partitioned into statistically independent groups, then $\boldsymbol{\Psi}_{\boldsymbol{x}}(\boldsymbol{\tau})$ is block-diagonal (with the respective partition(s)). Namely, two statistically independent random vectors are not only uncorrelated, but also "uncharrelated".*

Property 2. *If $\boldsymbol{x}$ can be expressed as a linear transformation of another random vector $\boldsymbol{a} \in \mathbb{R}^L$, namely $\boldsymbol{x} = \boldsymbol{H}\boldsymbol{a}$ where $\boldsymbol{H}$ is any $K \times L$ matrix, then*

$$\boldsymbol{\Psi}_{\boldsymbol{x}}(\boldsymbol{\tau}) = \boldsymbol{H}\boldsymbol{\Psi}_{\boldsymbol{a}}(\boldsymbol{H}^T\boldsymbol{\tau})\boldsymbol{H}^T, \qquad (3)$$

where $\boldsymbol{\Psi}_{\boldsymbol{a}}(\boldsymbol{H}^T\boldsymbol{\tau})$ is the charrelation matrix of $\boldsymbol{a}$ at $\boldsymbol{H}^T\boldsymbol{\tau}$. Thus, the effect of a linear transformation on the charrelation matrix resembles its effect on the correlation matrix.

It is shown in [6, 9] that consistent (and convenient) estimates of the charrelation matrix, based on N realizations $\boldsymbol{x}_n$ of $\boldsymbol{x}$, can be obtained as a "specially weighted" empirical covariance from:

$$\hat{\boldsymbol{\Psi}}_{\boldsymbol{x}}(\boldsymbol{\tau}) = \frac{\sum_{n=1}^N w_n(\boldsymbol{x}_n - \bar{\boldsymbol{x}})(\boldsymbol{x}_n - \bar{\boldsymbol{x}})^T}{\sum_{n=1}^N w_n} \qquad (4)$$

where the "weights" w_n are given by $w_n = \exp(\boldsymbol{\tau}^T\boldsymbol{x}_n)$ and where $\bar{\boldsymbol{x}}$ denotes a similarly weighted (generally non-zero) mean, $\bar{\boldsymbol{x}} \triangleq (\sum_n w_n\boldsymbol{x}_n)/(\sum_n w_n)$.

3. THE SUPEX ALGORITHM

To establish the baseline for our proposed modification, we now provide a brief overview of the SupEx algorithm, emphasizing aspects which are relevant to our proposed modification (for convenience, we shall use the notations of [4]). Let $\mathcal{H} = \{h_\ell\}$ denote the impulse-response of an unknown LTI channel (system), whose input is the zero-mean sequence $\{a_n\}$ of iid variables (with an unknown, non-Gaussian distribution), such that its output is given by $y_n = h_\ell \circ a_n$, where $\circ$ denotes the convolution operation. It is desired to construct an equalizer $\mathcal{C} = \{c_\ell\}$, such that its output (given its input $\{y_n\}$), $z_n = c_\ell \circ y_n \triangleq s_\ell \circ a_n$ is at most a scaled and delayed version of the source signal a_n; Namely the "combined system" $\mathcal{S} = \{s_\ell\} = \{c_\ell \circ h_\ell\}$ is given by $\{s_\ell\} = \{\alpha \cdot \delta_{\ell-k}\}$, where α is some arbitrary (nonzero) constant, k is some arbitrary (integer) delay, and δ_n denotes Kronecker's delta function. For simplicity of the exposition, we shall assume hereinafter that all signals / channels involved are real-valued.

The "philosophy of operation" behind the SupEx algorithm is the following: suppose that we could employ an iterative procedure, such that in each iteration the combined system $\mathcal{S}$ be modified as follows:

$$s'_n := \beta s_n^p \qquad (5)$$

(and then $\mathcal{S} = \{s'_n\}$), where $p > 1$ is some integer, and β is some (indirect) normalization coefficient, which may be different in each iteration. Then evidently, the ratio (in magnitude) between the largest tap and all other taps would be iteratively increased until the smaller taps vanish (due to the scaling constraint), and the desired response is obtained at a stationary point.

Although we do not have direct access to $\mathcal{S}$, but only to $\mathcal{C}$, it turns out that we can effectively approximate this desired process. Assume, from now on, that both the channel $\mathcal{H}$ and the equalizer $\mathcal{C}$ are of finite lengths M and L, respectively. Let an $(M+L-1) \times L$ Toeplitz matrix $\boldsymbol{H}$ be constructed as $H_{ij} = h_{i-j}$ (being zero when $i-j \notin [0, M-1]$), and define an $L \times 1$ vector $\boldsymbol{c} \triangleq [c_0 \; c_1 \; \cdots \; c_{L-1}]^T$. The combined system response vector $\boldsymbol{s} \triangleq [s_0 \; s_1 \; \cdots \; s_{M+L-1}]^T$ is then given, for any equalizer $\boldsymbol{c}$, by

$$\boldsymbol{s} = \boldsymbol{H}\boldsymbol{c}. \tag{6}$$

Now, as mentioned above, in each iteration we would like to change $\boldsymbol{s}$ into $\boldsymbol{s}' = \boldsymbol{g}$, where $\boldsymbol{g}$ is an $(M + L - 1) \times 1$ vector, whose n-th element is s_n^p (prior to the normalization operation). Usually, $\boldsymbol{g}$ does not belong to the range-space of $\boldsymbol{H}$, and therefore exact equality cannot be attained by any equalizer $\boldsymbol{c}$. Nevertheless, the value of $\boldsymbol{c}$ which minimizes the least-squares (LS) distance between $\boldsymbol{s}'$ and $\boldsymbol{g}$ is well known to be given by:

$$\min_{\boldsymbol{c}} \|\boldsymbol{H}\boldsymbol{c}' - \boldsymbol{g}\|^2; \Rightarrow \quad \boldsymbol{c}' = (\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{g}. \tag{7}$$

Luckily, we can obtain consistent estimates of the desired $\boldsymbol{c}'$ (up to irrelevant scaling) as follows. Let us define the $L \times 1$ random vector $\boldsymbol{y}$ and the $(M + L - 1) \times 1$ random vector $\boldsymbol{a}$ such that $\boldsymbol{y} \triangleq [y_n \ y_{n-1} \ \cdots \ y_{n-L+1}]^T$, $\boldsymbol{a} \triangleq [a_n \ a_{n-1} \ \cdots \ a_{n-M-L+2}]^T$ and

$$\boldsymbol{y} = \boldsymbol{H}^T\boldsymbol{a} \tag{8}$$

(where, due to the stationarity, the statistical properties of $\boldsymbol{y}$ and $\boldsymbol{a}$ do not depend on n). We now observe that

- The correlation matrix of $\boldsymbol{y}$ is given by
$$\boldsymbol{R}_{yy} = E[\boldsymbol{y}\boldsymbol{y}^T] = \boldsymbol{H}^T E[\boldsymbol{a}\boldsymbol{a}^T]\boldsymbol{H} = \sigma_a^2 \cdot (\boldsymbol{H}^T\boldsymbol{H}), \tag{9}$$
where σ_a^2 denotes the variance of a_n;
- The joint-cumulants vector $\boldsymbol{d}_{yz}$, whose k-th element is defined as $d_k \triangleq \text{cum}\{z : p; y_k\}$ (here y_k denotes the k-th element of $\boldsymbol{y}$ and cum$\{\cdot\}$ denotes the joint cumulant) is given by (see [4] for a simple proof)
$$\boldsymbol{d}_{yz} = \text{cum}\{a : p + 1\} \cdot \boldsymbol{H}^T\boldsymbol{g}. \tag{10}$$

Since both $\boldsymbol{R}_{yy}$ and $\boldsymbol{d}_{yz}$ can be consistently estimated from the observed signal $\{y_n\}$ and the equalizer's output $\{z_n\}$, the desired updated equalizer $\boldsymbol{c}'$ of (7) can be consistently estimated (up to irrelevant scaling) as well.

Thus, the super-exponential algorithm proceeds as follows (batch version): First, the correlation matrix $\boldsymbol{R}_{yy}$ is empirically estimated from the channel's output sequence $\{y_n\}$. Then, in each iteration the output sequence $\{z_n\} = \{c_n\} \circ \{y_n\}$ of the current equalizer is computed, and the cross-cumulants vector $\boldsymbol{d}_{yz}$ is empirically estimated. An updated equalizer is then computed and normalized using

$$\boldsymbol{c}' := \hat{\boldsymbol{R}}_{yy}^{-1}\hat{\boldsymbol{d}}_{yz} \quad , \quad \boldsymbol{c}'' := \boldsymbol{c}'/\sqrt{\boldsymbol{c}'^T\boldsymbol{c}'}, \tag{11}$$

and a new iteration is applied with the new equalizer $c = c''$.

A common value of p (used especially for symmetrically distributed sources) is $p = 3$, implying the use of fourth-order cross-cumulants in $\boldsymbol{d}_{yz}$. Convergence is typically attained (for sources with strong kurtosis) within $4 - 5$ iterations.

4. THE PROPOSED MODIFICATION

To propose our alternative statistic, let us first define an $(L+1) \times 1$ vector $\boldsymbol{x}$ as follows:

$$\boldsymbol{x} \triangleq \begin{bmatrix} \boldsymbol{y} \\ z \end{bmatrix} = \begin{bmatrix} \boldsymbol{H}^T \\ \boldsymbol{c}^T\boldsymbol{H}^T \end{bmatrix} \cdot \boldsymbol{a} \triangleq \widetilde{\boldsymbol{H}}\boldsymbol{a}. \tag{12}$$

From Property 2 above it follows that at any processing-point $\boldsymbol{\tau}$, the "charrelation" matrix of $\boldsymbol{x}$ is given by

$$\boldsymbol{\Psi}_{\boldsymbol{x}}(\boldsymbol{\tau}) = \widetilde{\boldsymbol{H}}\boldsymbol{\Psi}_{\boldsymbol{a}}(\widetilde{\boldsymbol{H}}^T\boldsymbol{\tau})\widetilde{\boldsymbol{H}}^T. \tag{13}$$

Let us now choose $\boldsymbol{\tau} = [\boldsymbol{0}^T \ \ \tau]^T$, where $\boldsymbol{0}$ denotes an $L \times 1$ all-zeros vector and τ is a scalar parameter. We then have $\widetilde{\boldsymbol{H}}^T\boldsymbol{\tau} = \tau\boldsymbol{H}\boldsymbol{c} = \tau\boldsymbol{s}$. Now, by Property 1, $\boldsymbol{\Psi}_{\boldsymbol{a}}(\widetilde{\boldsymbol{H}}^T\boldsymbol{\tau}) = \boldsymbol{\Psi}_{\boldsymbol{a}}(\tau\boldsymbol{s})$ is a diagonal matrix (since $\boldsymbol{a}$ has iid elements), which we shall denote $\boldsymbol{\Lambda}(\tau\boldsymbol{s}) \triangleq \text{diag}\{\boldsymbol{\lambda}(\tau\boldsymbol{s})\}$, where

$$\boldsymbol{\lambda}(\tau\boldsymbol{s}) = [\psi_a''(\tau s_1) \ \psi_a''(\tau s_2) \ \cdots \ \psi_a''(\tau s_{M+L-1})]^T. \tag{14}$$

We now partition $\boldsymbol{\Psi}_{\boldsymbol{x}}(\boldsymbol{\tau})$ (denoted $\boldsymbol{\Psi}_{\boldsymbol{x}}(\tau)$ for convenience, since $\boldsymbol{\tau}$ now depends on a single parameter τ) as follows,

$$\boldsymbol{\Psi}_{\boldsymbol{x}}(\tau) = \begin{bmatrix} \boldsymbol{P}_{yy}(\tau) & \boldsymbol{p}_{yz}(\tau) \\ \boldsymbol{p}_{zy}(\tau) & p_{zz}(\tau) \end{bmatrix}, \tag{15}$$

where $\boldsymbol{P}_{yy}(\tau)$ is $L \times L$, $\boldsymbol{p}_{yz}(\tau) = \boldsymbol{p}_{zy}^T(\tau)$ is $L \times 1$ and $p_{zz}(\tau)$ is a scalar. Concentrating on $\boldsymbol{p}_{zy}(\tau)$, from (12), (13) and (14) we obtain

$$\boldsymbol{p}_{yz}(\tau) = \boldsymbol{H}^T\boldsymbol{\Lambda}(\tau\boldsymbol{s})\boldsymbol{H}\boldsymbol{c} = \boldsymbol{H}^T\boldsymbol{\Lambda}(\tau\boldsymbol{s})\boldsymbol{s} \triangleq \boldsymbol{H}^T\bar{\boldsymbol{g}} \tag{16}$$

where $\bar{\boldsymbol{g}}$ is an $(M + L - 1) \times 1$ vector whose n-th elements is given by $\psi_a''(\tau s_n)s_n$. Now, let us define

$$\widetilde{\boldsymbol{d}}_{yz} \triangleq \boldsymbol{r}_{yz} - \boldsymbol{p}_{yz}(\tau) = \boldsymbol{H}^T(\sigma_a^2\boldsymbol{s} - \bar{\boldsymbol{g}}) \triangleq \boldsymbol{H}^T\widetilde{\boldsymbol{g}}, \tag{17}$$

where $\boldsymbol{r}_{yz} defeq E[\boldsymbol{y}z] = E[\boldsymbol{H}^T\boldsymbol{a}\boldsymbol{a}^T\boldsymbol{H}\boldsymbol{c}] = \sigma_a^2\boldsymbol{H}^T\boldsymbol{s}$ is the cross-correlation vector between $\boldsymbol{y}$ and z. The n-th element of $\widetilde{\boldsymbol{g}}$ is given by $(\sigma_a^2 - \psi_a''(\tau s_n))s_n$, which can also be expressed as $f_\tau(s_n)s_n$, where

$$f_\tau(s_n) \triangleq \sigma_a^2 - \psi_a''(\tau s_n) \tag{18}$$

is a function of s_n (continuous under some regularity conditions), being zero for $s_n = 0$ (since the second derivative of the LCF at the origin is the variance), with a monotonically increasing magnitude at least in some vicinity of $s_n = 0$. A typical example of $f_\tau(s)$ is depicted in Fig.1.

Thus, $f_\tau(s_n)$ can be regarded as playing in $\widetilde{\boldsymbol{g}}$ the same role that s_n^{p-1} plays in $\boldsymbol{g}$. In other words, if $\widetilde{\boldsymbol{d}}_{yz}$ is used in (11) instead of $\boldsymbol{d}_{yz}$, the effect on the update of $\boldsymbol{s}'$ in (5) becomes

$$s_n' := \beta f_\tau(s_n)s_n, \tag{19}$$

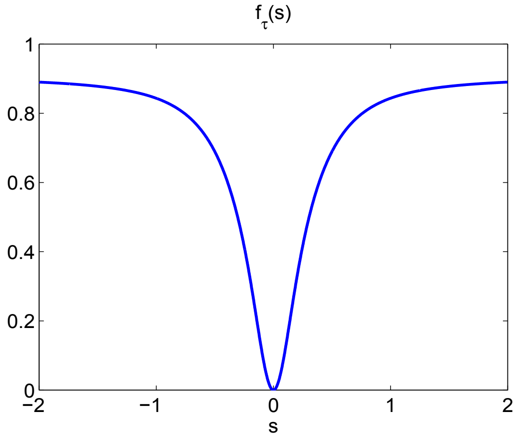

Figure 1: $f_\tau(s)$ (with $\tau = 4.0$) for the source distribution used in the simulation. The exact expression is $f_\tau(s) = \frac{qA^2+(1-q)B^2}{3} - q\left[\frac{1}{(\tau s)^2} - \frac{A^2}{\sinh^2(A\tau s)}\right] - (1-q)\left[\frac{1}{(\tau s)^2} - \frac{B^2}{\sinh^2(B\tau s)}\right]$, see text for values of A, B, q.

rather than $s'_n = \beta s_n^p$; Nevertheless, the "poor gets poorer" effect is similar: since $f_\tau(s_n)$ is monotonically increasing with the magnitude of s_n, it follows that in each iteration any values in $\mathcal{S}$ smaller in magnitude than the peak become (relatively) even smaller, until they converge to zero. This is the same "philosophy of operation" as in the original version of SupEx, but it is not based on cumulants, but rather on estimated "charrelations".

Straightforward empirical estimates of $\widetilde{\boldsymbol{d}}_{yz}$ can be obtained from (recall (4))

$$\hat{\widetilde{\boldsymbol{d}}}_{yz} = \frac{1}{N}\sum_n \boldsymbol{y}_n z_n - \frac{\sum_n w_n(\boldsymbol{y}_n - \bar{\boldsymbol{y}})(z_n - \bar{z})}{\sum_n w_n}, \quad (20)$$

where $\bar{\boldsymbol{y}} \triangleq \frac{\sum_n w_n \boldsymbol{y}_n}{\sum_n w_n}$, $\bar{z} \triangleq \frac{\sum_n w_n z_n}{\sum_n w_n}$ and where, due to the special structure of $\boldsymbol{\tau}$, the "weights" w_n are given by $w_n = \exp(\tau z_n)$.

5. SIMULATION RESULTS, CONCLUSION

To demonstrate the potential improvement in performance, we applied the modified algorithm using the following channel / equalizer setup (similar to that in [4]): the channel was $\{h_\ell\} = \{0.4\ 1\ \ -0.7\ 0.6\ 0.3\ \ -0.4\ 0.1\}$ and the equalizer length was 16, initialized as all-zeros with a 1 at the fourth tap.

The source distribution was $qU(A) + (1-q)U(B)$, where $U(\theta)$ denotes the Uniform symmetric distribution between $-\theta$ and θ. When $A = 3(1+\sqrt{2})$, $B = 3 - \sqrt{2}$ and $q = 0.25$, this zero-mean source has unit variance and a null kurtosis. To salvage the 4th-order SupEx from total failure, we slightly changed these conditions, using the same A and B, but with $q = 0.2$.

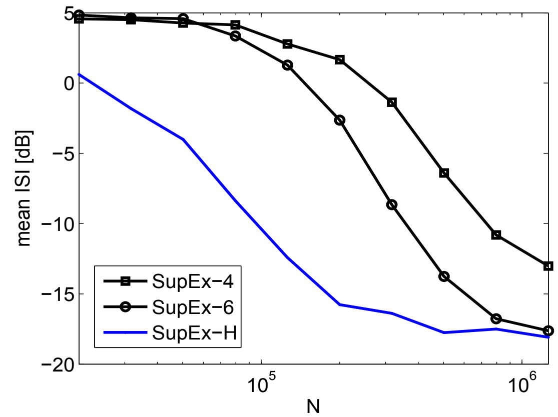

Figure 2: Empirical ISI (in [dB], averaged over 400 trials), for SupEx based on 4th order, on 6th order cumulants, and on the proposed Hessian with $\tau = 4.0$.

Performance is presented in Fig.2 in terms of the resulting (empirical) Inter-Symbol Interference (ISI) vs. the observation length N, for SupEx based on 4-th order cumulants, on 6-th order cumulants and on the proposed Hessian (using $\tau = 4.0$). Significant improvement of the Hessian-based version over the cumulants-based version is evident.

In general, the performance would obviously depend on the selection of τ. Combining several values of τ (namely, using several estimates in parallel) is also possible. However, a strategy for optimizing the selection(s) of τ, as well as analytical performance evaluation, are currently the subjects of on-going further research.

6. REFERENCES

[1] Comon P. and Rajih M., "Blind Identification of Under-Determined Mixtures based on the Characteristic Function," *Signal Processing*, 86(9):2271–2281, 2006.

[2] Gürelli M. I. and Nikias C. L., "Blind identification and array processing applications of generalized higher-order statistics," *Proc. MILCOM*, pp.838–842, 1996.

[3] Kawanabe M. and Theis F.J. "Joint low-rank approximation for extracting non-Gaussian subspaces," *Signal Processing* 87(8):1890-1903, 2007.

[4] Shalvi O. and Weinstein, E., "Super-Exponential methods for blind deconvolution," *IEEE Trans. Information Theory*, 39:504–519, 1993.

[5] Shalvi O. and Weinstein, E., "New criteria for blind deconvolution of non-minimum phase systems (channels)," *IEEE Trans. Information Theory*, 36:312-321, 1990.

[6] Yeredor A. "Blind Source Separation via the Second Characteristic Function," *Signal Processing*, 80(5):897–902, 2000.

[7] Yeredor A., "Blind channel estimation using first and second derivatives of the characteristic function," *IEEE Signal Processing Letters*, 9(3):100–103, 2002.

[8] Yeredor A., "MUSIC using Off-Origin Hessians of the Second Characteristic Function," *Proc. SAM2006*, 2006.

[9] Yeredor A., "Yule-Walker equations applied to Hessians of the characteristic function for improved AR estimation," *Proc. ICASSP2007*, 2007.