entropy

Article

Function Analysis of the Euclidean Distance between Probability Distributions

Namyong Kim

Division of Electronic & Information Communication, Kangwon National University, Samcheok 25913, Korea; namyong@kangwon.ac.kr; Tel.: +82-01-7188-5872

Received: 20 November 2017; Accepted: 8 January 2018; Published: 11 January 2018

Abstract: Minimization of the Euclidean distance between output distribution and Dirac delta functions as a performance criterion is known to match the distribution of system output with delta functions. In the analysis of the algorithm developed based on that criterion and recursive gradient estimation, it is revealed in this paper that the minimization process of the cost function has two gradients with different functions; one that forces spreading of output samples and the other one that compels output samples to move close to symbol points. For investigation the two functions, each gradient is controlled separately through individual normalization of each gradient with their related input. From the analysis and experimental results, it is verified that one gradient is associated with the role of accelerating initial convergence speed by spreading output samples and the other gradient is related with lowering the minimum mean squared error (MSE) by pulling error samples close together.

Keywords: distribution; recursive; gradient; spreading; functions

1. Introduction

Adaptive signal processing is carried out by minimizing or maximizing an appropriate performance criterion for adjusting weights of algorithms designed based on that criterion [1]. The mean squared error (MSE) criterion that measures the average of the squares of the error signal is widely employed in the Gaussian noise environment. However in non-Gaussian noise like impulsive noise, the averaging process of squared error samples that may mitigate the effects of the Gaussian noise is defeated because a single large, impulse can dominate these sums. As recent signal processing methods, the information-theoretic learning (ITL) is based on the information potential concept that data samples can be treated as physical particles in an information potential field where they interact with each other by information forces [2]. The ITL method usually exploits probability distribution functions constructed by the kernel density estimation method with the Gaussian kernel.

Among the ITL criteria, Euclidian distance (ED) between two distributions has been known to be effective in signal processing fields demanding similarity measure functions [3–5]. For training of adaptive systems for medical diagnosis, the ED criterion has been successfully applied to distinguish biomedical datasets [6]. For finite impulse response (FIR) adaptive filter structures in impulsive noise environments, ED between the output distribution and a set of Dirac delta functions has been used as an efficient performance criterion taking advantage of the outlier-cutting effect of Gaussian kernel for output pairs and symbol-output pairs [7]. In this approach with output distribution and delta functions, minimization of the ED (MED) leads to adaptive algorithms that adjust weights so as for the output distribution to be formed into the shape of delta functions located at each symbol point, that is, output samples concentrate on symbol points. Though the blind MED algorithm shows superior performance of robustness against impulsive noise and channel distortions, a drawback of heavy computational burden lies in it. The computational complexity is due in large part to the double summation operations at each iteration time for its gradient estimation. A follow-up study [8], however, shows that the drawback can be reduced significantly by employing a recursive gradient estimation method.

The gradient in ED minimization process of the MED algorithm has two components; one for kernel function of output pairs and the other for kernel function of symbol-output pairs. The roles of these two components have not been investigated or analyzed in scientific literature. In this paper, we analyze the roles of the two components and prove the analysis through controlling each component individually by normalizing each component with component-related input power. Through simulation in multipath channel equalization under impulsive noise, their roles of managing sample pairs are verified, and it is shown that the proposed method of controlling each component through power normalization increases convergence speed and lowers steady state MSE significantly in multipath and impulsive noise environment.

2. MSE Criterion and Related Algorithms

Employing the tapped delay line (TDL) structure, the output y_k becomes $y_k = \mathbf{W}_k^T\mathbf{X}_k$ at time k with the input vector $\mathbf{X}_k = [x_k, x_{k-1}, \ldots, x_{k-L+1}]^T$ and weight $\mathbf{W}_k = [w_{0.k}, w_{1,k}, \ldots, w_{L-1,k}]^T$. Given the desired signal d_k chosen randomly among the M symbol points ($A_1, A_2, \ldots, A_M$), the system error is calculated as $e_k = d_k - y_k$. In blind equalization, the constant modulus error $e_{\text{CME},k} = |y_k|^2 - R_2$ where $R_2 = E[|d_k|^4]/E[|d_k|^2]$ is mostly used [9].

The MSE criterion, one of the most widely used criteria, is the statistical average $E[\cdot]$ of error power e_k^2 in supervised equalization and of CME power $(|y_k|^2 - R_2)^2$ in a blind one. For practical implementation we can use the instant squared error e_k^2 as a cost function in supervised equalization. With the gradient $\frac{\partial e_k^2}{\mathbf{W}} = -2e_k\mathbf{X}_k$ and a step size μ_{LMS}, minimization of e_k^2 leads to the least mean square (LMS) algorithm [1]:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu_{\text{LMS}}\frac{\partial e_k^2}{\partial \mathbf{W}} = \mathbf{W}_k + \mu_{\text{LMS}}2e_k\mathbf{X}_k \tag{1}$$

As an extension of the LMS algorithm, the normalized LMS (NLMS) algorithm has been introduced where the gradient is normalized as proportional to the inverse of the dot product of the input vector with itself $\|\mathbf{X}_k\|^2 = \mathbf{X}_k^T\mathbf{X}_k = \sum_{m=0}^{L-1} x_{k-m}^2$ as a result of minimizing weight perturbation $\|\mathbf{W}_{k+1} - \mathbf{W}_k\|^2$ of the LMS algorithm [1]. Then the NLMS algorithm becomes:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu_{\text{NLMS}}\frac{e_k\mathbf{X}_k}{\sum\limits_{m=0}^{L-1} x_{k-m}^2} \tag{2}$$

The NLMS algorithm is known to be more stable with unknown signals and effective in real time adaptive systems [10,11]. We can see under impulsive noise environments that a single large error sample induced by impulsive noise can generate large weight perturbations. The perturbation becomes zero only when the error e_k is zero. So we can predict that the weight update process (1) may be unstable so that it requires a very small step size in impulsive noise environment. Also the LMS and NLMS algorithms utilizing instant error power e_k^2 may cause instability in an impulsive noise environment.

3. ED Criterion and Entropy

Unlike the MSE based on error power, probability distribution functions can be used in constructing performance criterion. As one of the criteria utilizing distributions, the ED between the distribution of transmitted symbol $f_D(d)$ and the equalizer output distribution $f_Y(y)$ is defined as (3) [3,6].

$$\text{ED} = \int [f_D(\alpha) - f_Y(\alpha)]^2 \mathrm{d}\alpha \tag{3}$$

Assuming that modulation schemes are known to receivers beforehand and all the M symbol points (A_1, A_2, ..., A_M) are equally likely, the distribution of the transmitted symbols can be expressed as:

$$f_D(\alpha) = \frac{1}{M}[\delta(\alpha - A_1) + \delta(\alpha - A_2) + \ldots + \delta(\alpha - A_m) + \ldots + \delta(\alpha - A_M)] \tag{4}$$

The output distribution can be estimated based on kernel density estimation method $f_Y(y) = 1/N\sum_{i=1}^{N} G_\sigma(y - y_i)$ with a set of available N output samples $\{y_1, y_2, \ldots, y_N\}$ [6].

Then the ED can be expressed as:

$$\mathrm{ED} = \frac{1}{M} + \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G_{\sigma\sqrt{2}}(y_j - y_i) - 2\frac{1}{M}\frac{1}{N}\sum_{m=1}^{M}\sum_{i=1}^{N} G_\sigma(A_m - y_i) \tag{5}$$

The first term $1/M$ in (5) is a constant which is not adjustable, so the ED can be reduced to the following performance criterion C_{ED} [7]:

$$C_{\mathrm{ED}} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G_{\sigma\sqrt{2}}(y_j - y_i) - 2\frac{1}{M}\frac{1}{N}\sum_{m=1}^{M}\sum_{i=1}^{N} G_\sigma(A_m - y_i) \tag{6}$$

In ITL methods, data samples are treated as physical particles interacting with each other. If we place physical particles in the locations of y_i and y_j, the Gaussian kernel $G_{\sigma\sqrt{2}}(y_j - y_i)$ produces an exponentially decaying positive value as the distance between the two particles increases. This leads us consider the Gaussian kernel $G_{\sigma\sqrt{2}}(y_j - y_i)$ as a potential field-inducing interaction among particles. Then $\sum_{j=1}^{N} G_{\sigma\sqrt{2}}(y_j - y_i)$ corresponds to the sum of interactions on the i-th particle and $1/N^2\sum_{i=1}^{N}\sum_{j=1}^{N} G_{\sigma\sqrt{2}}(y_j - y_i)$ is the averaged sum of all pairs of interactions. This summed potential energy is referred to as information potential in ITL methods [2]. Therefore, the term $\frac{1}{M}\frac{1}{N}\sum_{m=1}^{M}\sum_{i=1}^{N} G_\sigma(A_m - y_i)$ in (6) is the information potential between symbol points and output samples, and $1/N^2\sum_{i=1}^{N}\sum_{j=1}^{N} G_{\sigma\sqrt{2}}(y_j - y_i)$ in (6) indicates the information potential among output samples themselves.

On the other hand, the information potential can be interpreted in the concept of entropy that can be described in terms of "energy dispersal" or the "spreading of energy" [11]. As one of the convenient entropy definitions, Reny's entropy of order 2, $H_{\mathrm{Reny}}(y)$ is defined in (7) as logarithm of the sum of the power of probability which is much easier to estimate [2]:

$$H_{\mathrm{Reny}}(y) = -\log(\sum_{i=1}^{N} p_i^2) \tag{7}$$

When the Reny's entropy is used along with the kernel density estimation method $f_Y(y) = 1/N\sum_{i=1}^{N} G_\sigma(y - y_i)$, we obtain a much simpler form of Reny's quadratic entropy as:

$$H_{\mathrm{Reny}}(y) = -\log(\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G_{\sigma\sqrt{2}}(y_j - y_i)) \tag{8}$$

This leads to:

$$\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G_{\sigma\sqrt{2}}(y_j - y_i) = \frac{1}{2^{H_{\mathrm{Reny}}(y)}} \tag{9}$$

Likewise:

$$\frac{1}{M}\frac{1}{N}\sum_{m=1}^{M}\sum_{i=1}^{N} G_\sigma(A_m - y_i) = \frac{N}{M}\frac{1}{2^{H_{\mathrm{Reny}}(Am,x)}} \tag{10}$$

Therefore the cost function C_{ED} becomes:

$$C_{\text{ED}} = \frac{1}{2^{H_{\text{Reny}}(y)}} - 2\frac{N}{M}\frac{1}{2^{H_{\text{Reny}}(Am,x)}} \tag{11}$$

Equations (9) and (11) indicate that the entropy of output samples increases as the distance $(y_j - y_i)$ between the two information particles y_j and y_i increases. Therefore, $(y_j - y_i)$ can be referred to as entropy-governing output and we can notice that (9) controls the spreading of output samples. Likewise, the term $2\frac{1}{M}\frac{1}{N}\sum_{m=1}^{M}\sum_{i=1}^{N} G_\sigma(A_m - y_i)$ in (6), that is, $2\frac{N}{M}\frac{1}{2^{H_{Reny}(Am,x)}}$ in (11) governs dispreading or recombining the sample pairs of symbol points and output samples.

4. Entropy-Governing Variables and Recursive Algorithms

When defining $y_{j,i}$ = $(y_j - y_i)$ and $e_{m,i}$ = $(A_m - y_i)$ and $\mathbf{X}_{j.i}$ = $(\mathbf{X}_j - \mathbf{X}_i)$ for convenience's sake, $y_{j,i}$, $e_{m,i}$ and $\mathbf{X}_{j,i}$ can be referred to as entropy-governing output, entropy-governing error and entropy-governing input, respectively. Using these entropy-governing variables and the on-line density estimation method $f_{X.k}(y) = \frac{1}{N}\sum_{i=k-N+1}^{k} G_\sigma(y - y_i)$ instead of $f_Y(y)$, the cost function at time k, $C_{\text{ED},k}$ can be written as:

$$C_{\text{ED},k} = U_k - V_k \tag{12}$$

where:

$$U_k = \frac{1}{2^{H_{\text{Reny}}(y)}} = \frac{1}{N^2}\sum_{i=k-N+1}^{k}\sum_{j=k-N+1}^{k} G_{\sigma\sqrt{2}}(y_{j,i}) \tag{13}$$

$$V_k = 2\frac{N}{M}\frac{1}{2^{H_{Reny}(Am,x)}} = 2\frac{1}{M}\frac{1}{N}\sum_{i=k-N+1}^{k}\sum_{j=k-N+1}^{k} G_\sigma(e_{m,i}) \tag{14}$$

Minimization of $C_{\text{ED},k}$ indicates that U_k forces spreading of output samples and $-V_k$ compels output samples to move close to symbol points. Considering that initial-stage output samples which may have clustered about wrong places due to channel distortion, U_k is associated with the role of getting the output samples to move out in search of each destination, that is, accelerating initial convergence speed. On the other hand, V_k is related with compelling output samples near a symbol point to come close lowering the minimum MSE.

On the other hand, the double summation operations for U_k and V_k impose a heavy computational burden. In the work [8] it has been revealed that each component U_{k+1} and V_{k+1} of $C_{\text{ED},k+1}$ = $U_{k+1} - V_{k+1}$ can be recursively calculated so that the computational complexity of (12) is significantly reduced as in the following equations (15) and (16):

$$\begin{aligned} U_{k+1} = U_k + \tfrac{2}{N^2}\sum_{j=k-N+1}^{k} G_{\sigma\sqrt{2}}(y_{i,k+1}) - \tfrac{2}{N^2}\sum_{j=k-N+1}^{k} \tfrac{1}{2\sigma\sqrt{\pi}}\exp[\tfrac{-(y_{i,k-N+1})^2}{4\sigma^2}] \\ - \tfrac{2}{N^2}\tfrac{1}{2\sigma\sqrt{\pi}}\exp[\tfrac{-(y_{k+1,k-N+1})^2}{4\sigma^2}] + \tfrac{2}{N^2}\tfrac{1}{2\sigma\sqrt{\pi}} \end{aligned} \tag{15}$$

Similarly, V_{k+1} can be divided into the terms with y_{k+1} and the terms with y_{k-N+1}:

$$V_{k+1} = V_k + \frac{2}{NM}\sum_{m=1}^{M}\left[\frac{1}{\sigma\sqrt{2\pi}}\exp[\frac{-(e_{m,k+1})^2}{2\sigma^2}] - \frac{1}{\sigma\sqrt{2\pi}}\exp[\frac{-(e_{m,k-N+1})^2}{2\sigma^2}]\right] \tag{16}$$

The gradients $\frac{\partial U_k}{\partial \mathbf{W}}$ and $\frac{\partial V_k}{\partial \mathbf{W}}$ are calculated recursively by using Equations (15) and (16) as:

$$\begin{aligned}\frac{\partial U_k}{\partial \mathbf{W}} = \frac{\partial U_{k-1}}{\partial \mathbf{W}} &+ \frac{1}{N^2\sigma^2}\sum_{j=k-N}^{k-1}(y_{k,i})\cdot\frac{1}{2\sigma\sqrt{\pi}}\exp[\frac{-(y_{k,i})^2}{4\sigma^2}]\cdot\mathbf{X}_{i,k}\\ &-\frac{1}{N^2\sigma^2}\sum_{j=k-N}^{k-1}(y_{k-N,i})\cdot\frac{1}{2\sigma\sqrt{\pi}}\exp[\frac{-(y_{k-N,i})^2}{4\sigma^2}]\cdot\mathbf{X}_{i,k-N}\\ &-\frac{1}{N^2\sigma^2}(y_{k-N,k})\cdot\frac{1}{2\sigma\sqrt{\pi}}\exp[\frac{-(y_{k-N,k})^2}{4\sigma^2}]\cdot\mathbf{X}_{k,k-N}\end{aligned} \tag{17}$$

Similarly, $\frac{\partial V_k}{\partial \mathbf{W}}$ is calculated recursively as described below:

$$\begin{aligned}\frac{\partial V_k}{\partial \mathbf{W}} = \frac{\partial V_{k-1}}{\partial \mathbf{W}} + \frac{2}{NM\sigma^2}\sum_{m=1}^{M}\Big[&(e_{m,k})\cdot\frac{1}{\sigma\sqrt{2\pi}}\exp[\frac{-(e_{m,k})^2}{2\sigma^2}]\cdot\mathbf{X}_k\\ &-(e_{m,k-N})\cdot\frac{1}{\sigma\sqrt{2\pi}}\exp[\frac{-(e_{m,k-N})^2}{2\sigma^2}]\cdot\mathbf{X}_{k-N}\Big]\end{aligned} \tag{18}$$

Since the argument $y_{k,i}\frac{1}{2\sigma\sqrt{\pi}}\exp\left[\frac{(y_{k,i})^2}{4\sigma^2}\right]$ in (17) is a function of the entropy-governing output $y_{k,i}$, we can define $y_{k,i}\frac{1}{2\sigma\sqrt{\pi}}\exp\left[\frac{(y_{k,i})^2}{4\sigma^2}\right]$ as the modified entropy-output $\hat{y}_{k,i}$, which becomes a significantly mitigated value through the Gaussian kernel when the entropy-governing output $y_{k,i}$ is a large value.

$$\hat{y}_{k,i} = y_{k,i}\frac{1}{2\sigma\sqrt{\pi}}\exp[\frac{-(y_{k,i})^2}{4\sigma^2}] \tag{19}$$

Then (17) becomes

$$\frac{\partial U_k}{\partial \mathbf{W}} = \frac{\partial U_{k-1}}{\partial \mathbf{W}} + \frac{1}{N^2\sigma^2}\sum_{j=k-N}^{k-1}\hat{y}_{k,i}\cdot\mathbf{X}_{i,k} - \frac{1}{N^2\sigma^2}\sum_{j=k-N}^{k-1}\hat{y}_{k-N,i}\cdot\mathbf{X}_{i,k-N} - \frac{1}{N^2\sigma^2}\hat{y}_{k-N,k}\cdot\mathbf{X}_{k,k-N} \tag{20}$$

Similarly, we see that the argument $e_{m,k}\frac{1}{\sigma\sqrt{2\pi}}\exp\left[\frac{-(e_{m,k})^2}{2\sigma^2}\right]$ in (18) is a function of entropy-governing error $e_{m,k}$, so that we have the modified entropy-error $\hat{e}_{m,k}$ as:

$$\hat{e}_{m,k} = e_{m,k}\cdot\frac{1}{\sigma\sqrt{2\pi}}\exp[\frac{-(e_{m,k})^2}{2\sigma^2}] \tag{21}$$

The modified entropy-error $\hat{e}_{m,k}$ also becomes a significantly reduced value through the Gaussian kernel when the entropy-governing error $e_{m,k}$ is large. Then (18) becomes:

$$\frac{\partial V_k}{\partial \mathbf{W}} = \frac{\partial V_{k-1}}{\partial \mathbf{W}} + \frac{2}{NM\sigma^2}\sum_{m=1}^{M}\left[\hat{e}_{m,k}\cdot\mathbf{X}_k - \hat{e}_{m,k-N}\cdot\mathbf{X}_{k-N}\right] \tag{22}$$

Through minimization of $C_{\mathrm{ED},k} = U_k - V_k$ with the gradients $\frac{\partial U_k}{\partial \mathbf{W}}$ and $\frac{\partial V_k}{\partial \mathbf{W}}$ obtained by (20) and (22), the following recursive MED (RMED) algorithm can be derived [7]:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu_{\mathrm{RMED}}\frac{\partial(U_k - V_k)}{\partial \mathbf{W}} = \mathbf{W}_k - \mu_{\mathrm{RMED}}(\frac{\partial U_k}{\partial \mathbf{W}} - \frac{\partial V_k}{\partial \mathbf{W}}) \tag{23}$$

Comparing the gradients of RMED to the gradient $\frac{\partial e_k^2}{\partial \mathbf{W}} = -2e_k\mathbf{X}_k$ of the LMS algorithm in (1) which is composed of error and input, we may find that the gradients $\frac{\partial U_k}{\partial \mathbf{W}}$ and $\frac{\partial V_k}{\partial \mathbf{W}}$ in (20) and (22) have similar terms $\hat{y}_{k,i}\cdot\mathbf{X}_{i,k}$ (modified entropy-output multiplied by entropy-input) and $\hat{e}_{m,k}\cdot\mathbf{X}_k$ (modified entropy-error multiplied by input), respectively. Considering that impulsive noise may induce large entropy-governing output $y_{k,i}$ or entropy-governing error $e_{m,k}$, modified entropy-output $\hat{y}_{k,i}$ and modified entropy-error $\hat{e}_{m,k}$ which are significantly mitigated by the Gaussian kernel can be viewed as playing a crucial role in obtaining stable gradients under strong impulsive noise. Therefore we can anticipate that the RMED algorithm (23) can have a low weight perturbation in impulsive noise environments.

5. Input Power Estimation for Normalized Gradient

For the purpose of minimizing the weight perturbation $\|\mathbf{W}_{k+1} - \mathbf{W}_k\|^2$ of the LMS algorithm in (1), the *NLMS* algorithm has been introduced where the gradient is normalized by the averaged power of the current input samples $\|\mathbf{X}_k\|^2 = \mathbf{X}_k^T\mathbf{X}_k = \sum_{m=0}^{L-1} x_{k-m}^2$ [1].

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu_{\text{NLMS}} \frac{e_k \mathbf{X}_k}{\|\mathbf{X}_k\|^2} \tag{24}$$

Applying this approach to RMED we propose in this section to normalize the gradients in some ways. Since the role of U_k (spreading output samples) is different from that of V_k (moving output samples close to symbol points), the gradients of (23) can be normalized separately as:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu_{\text{RMED}} \frac{\partial U_k}{\partial \mathbf{W}} \frac{1}{P_U(k)} + \mu_{\text{RMED}} \frac{\partial V_k}{\partial \mathbf{W}} \frac{1}{P_V(k)} \tag{25}$$

where $P_U(k)$ is the average power of $\mathbf{X}_{i,k}$ and $P_V(k)$ is the average power of $\mathbf{X}_k$ as:

$$P_U(k) = \frac{1}{N} \sum_{i=k-N+1}^{k} \sum_{j=k-N+1}^{k} \left| x_{i,j} \right|^2 \tag{26}$$

$$P_V(k) = \frac{1}{N} \sum_{i=k-N+1}^{k} |x_i|^2 \tag{27}$$

Since defeating the impulsive noise contained in the input by way of the average operation $\frac{1}{N} \sum\limits_{i=k-N+1}^{k}$ is considered to be ineffective, the denominators of (26) and (27) are likely to be fluctuating under impulsive noise. This may cause the algorithm to be sensitive to impulsive noise. Also the summation operators make the algorithm demand computationally burdensome. To avoid these drawbacks, we can track the average power $P_U(k)$ and $P_V(k)$ recursively with the balance parameter β ($0 < \beta < 1$) as:

$$P_U(k) = \beta P_U(k-1) + (1-\beta) \sum_{j=k-N+1}^{k} \left| x_{i,j} \right|^2 \tag{28}$$

$$P_V(k) = \beta P_V(k-1) + (1-\beta)|x_k|^2 \tag{29}$$

With the recursive power estimation (28) and (29), we may summarize the proposed algorithm in a more formal one as in the Table 1. In the following section, we will investigate the new RMED algorithm (25) with separate normalization by $P_U(k)$ in (28) and $P_V(k)$ in (29) in the aspect of convergence speed and steady state MSE.

Table 1. A summary of the proposed algorithm.

Process	Equations
Initialization	$\frac{\partial U_0}{\partial \mathbf{W}} = 0$, $\frac{\partial V_0}{\partial \mathbf{W}} = 0$, $P_U(0) = 1$, $P_V(0) = 1$, $\mathbf{W}_0 = [0, \ldots, 0, w_{L/2,0} = 1, 0, \ldots, 0]^T$
Update of gradient function $\frac{\partial U_k}{\partial \mathbf{W}}$	$\frac{\partial U_k}{\partial \mathbf{W}} = \frac{\partial U_{k-1}}{\partial \mathbf{W}} + \frac{1}{N^2\sigma^2} \sum_{j=k-N}^{k-1} \hat{y}_{k,i} \cdot \mathbf{X}_{i,k} - \frac{1}{N^2\sigma^2} \sum_{j=k-N}^{k-1} \hat{y}_{k-N,i} \cdot \mathbf{X}_{i,k-N} - \frac{1}{N^2\sigma^2} \hat{y}_{k-N,k} \cdot \mathbf{X}_{k,k-N}$
Update of gradient function $\frac{\partial V_k}{\partial \mathbf{W}}$	$\frac{\partial V_k}{\partial \mathbf{W}} = \frac{\partial V_{k-1}}{\partial \mathbf{W}} + \frac{2}{NM\sigma^2} \sum_{m=1}^{M} \left[\hat{e}_{m,k} \cdot \mathbf{X}_k - \hat{e}_{m,k-N} \cdot \mathbf{X}_{k-N} \right]$
Update of $P_U(k)$	$P_U(k) = \beta P_U(k-1) + (1-\beta) \sum_{j=k-N+1}^{k} \left\| x_{i,j} \right\|^2$
Update of $P_V(k)$	$P_V(k) = \beta P_V(k-1) + (1-\beta) \left\| x_k \right\|^2$
Update of $\mathbf{W}_k$	$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu_{\text{RMED}} \frac{\partial U_k}{\partial \mathbf{W}} \frac{1}{P_U(k)} + \mu_{\text{RMED}} \frac{\partial V_k}{\partial \mathbf{W}} \frac{1}{P_V(k)}$

6. Results and Discussion

A base-band communication system with multipath fading channel and impulsive noise used in the experiment is depicted in Figure 1. The symbol set in the transmitter is composed of equally probable four symbols (−3, −1, 1, 3). The transmitted symbol is to be distorted by the multipath channel $H(z) = 0.26 + 0.93z^{-1} + 0.26z^{-2}$ [12]. The channel output is added by impulsive noise n_k. The distribution function of n_k, $f(n_k)$ is expressed in Table 2 where σ_{IN}^2 is the variance of impulses which are generated according to Poisson process (occurrence rate ε) and σ_{GN}^2 is that of the background Gaussian noise [13]. The simulation setup and parameter values are described in the Figure 1 and the Table 2.

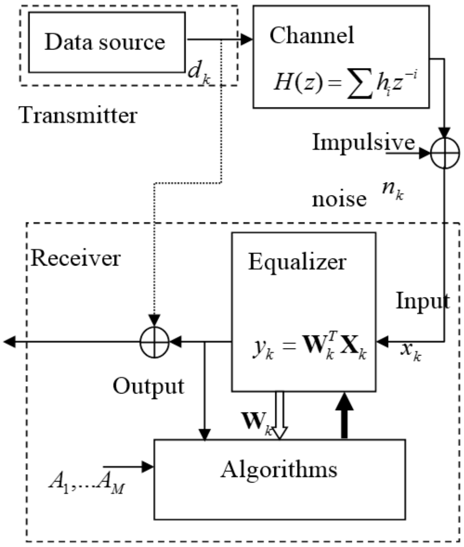

Figure 1. Base-band communication system for simulation.

Table 2. Simulation setup and parameter values.

Features	Parameters
The symbol points in the transmitter	$(A_1,\ A_2,\ A_3,\ A_4) = (-3,\ -1,\ +1,\ +3)$
The channel transfer function $H(z)$	$H(z) = 0.26 + 0.93z^{-1} + 0.26z^{-2}$
The noise distribution function $f(n_k)$	$f(n_k) = \frac{1-\varepsilon}{\sigma_{\text{GN}}\sqrt{2\pi}} \exp[\frac{-n_k{}^2}{2\sigma_{\text{GN}}^2}] + \frac{\varepsilon}{\sqrt{2\pi(\sigma_{\text{GN}}^2+\sigma_{\text{IN}}^2)}} \exp[\frac{-n_k{}^2}{2(\sigma_{\text{GN}}^2+\sigma_{\text{IN}}^2)}]$, $\varepsilon = 0.03$, $\sigma_{\text{GN}}^2 = 0.001$, $\sigma_{\text{GN}}^2 + \sigma_{\text{IN}}^2 = 50.001$
NNumber of weights	11
4 Step size	$\mu_{\text{CMA}} = 0.000001$, $\mu_{\text{LMS}} = 0.0002$, $\mu_{\text{RMED}} = 0.005$
Sample size N	6
Kernel size σ	0.6

An example of impulsive noise being used in this simulation is depicted in Figure 2.

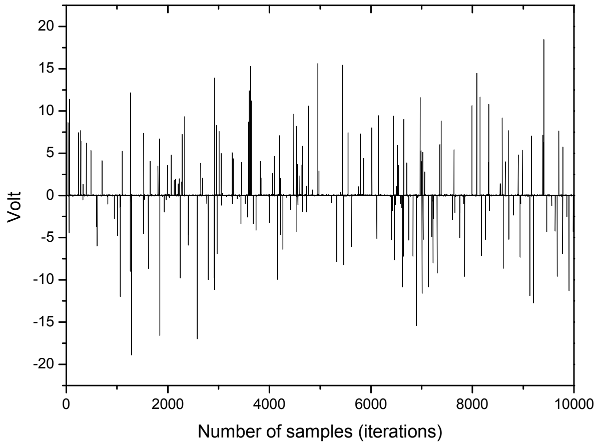

Figure 2. An example of impulsive noise.

It has in Section 4 been analyzed that U_k is associated with the role of spreading output samples which are clustered to wrong positions due to distorted channel characteristics and V_k is related with moving output samples close to symbol points. This process can be explained through initial-stage investigation of what happens in the error distribution and observing how the distribution of output samples changes in the experimental environment.

Figure 3 shows the error distribution in the initial stage with 200 error samples and ensemble average of 500 runs. Considering the four symbol points are $(-3, -1, 1, 3)$, error values greater than 1.0 are associated with output samples which can be decided as wrong symbols. The cumulative probability of initial output samples placed in the wrong regions in this respect is calculated to be 0.35 from the Figure 3 (35% output samples are not in place). The peaks or ridges in the error distribution are about 6 on each side. This observation may indicate that output samples are clustered or grouped in some regions (two groups are within the correct range but 4 groups are in the incorrect positions on each side of the distribution). This result coincides clearly with the initial output distribution in Figure 4. The output distribution showing about 12 peaks indicates that the initial output samples are clustered into 12 groups mostly located out of place, that is, not around $-3, -1, 1, 3$.

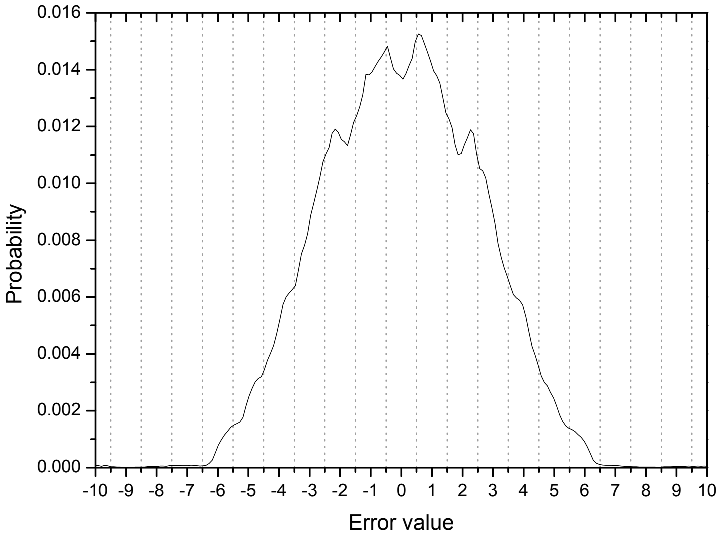

Figure 3. The error distribution at time k = 200 with 200 error samples.

On the 35% output samples clustered in the wrong symbol regions, the spreading force has a positive effect in order for them in blind search to move out for finding their correct symbol positions. This process is observed in the graph of k = 700 in Figure 4. The output distribution at time k = 700 has an evenly spread shape, indicating that the clustered output samples have moved out and mingled with one another. At the sample time k = 1800 the output samples start to position at their correct symbol areas. From this phase, the force moving output samples close to the symbol points is in effect on lowering steady state MSE.

These results imply that U_k is related with convergence speed and V_k with steady state MSE. To verify this analysis we experiment the proposed algorithm in the following three modes with respect to convergence speed and steady state MSE (we assume that steady state MSE is close to minimum MSE):

$$\text{Mode 1} \quad \mathbf{W}_{k+1} = \mathbf{W}_k - \mu_{\text{RMED}} \frac{\partial U_k}{\partial \mathbf{W}} \frac{1}{P_U(k)} + \mu_{\text{RMED}} \frac{\partial V_k}{\partial \mathbf{W}} \tag{30}$$

$$\text{Mode 2} \quad \mathbf{W}_{k+1} = \mathbf{W}_k - \mu_{\text{RMED}} \frac{\partial U_k}{\partial \mathbf{W}} + \mu_{\text{RMED}} \frac{\partial V_k}{\partial \mathbf{W}} \frac{1}{P_V(k)} \tag{31}$$

$$\text{Mode 3} \quad \mathbf{W}_{k+1} = \mathbf{W}_k - \mu_{\text{RMED}} \frac{\partial U_k}{\partial \mathbf{W}} \frac{1}{P_U(k)} + \mu_{\text{RMED}} \frac{\partial V_k}{\partial \mathbf{W}} \frac{1}{P_V(k)} \tag{32}$$

Mode 1 of RMED-SN algorithm in (30) is for observing changes in initial convergence speed by normalizing only $\frac{\partial U_k}{\partial \mathbf{W}}$ by the average power $P_U(k)$ of entropy-input $\mathbf{X}_{i,k}$ compared to the not-normalized RMED. Mode 2 is to observe whether the normalization of $\frac{\partial V_k}{\partial \mathbf{W}}$ by $P_V(k)$ of input $\mathbf{X}_k$ without managing U_k lowers the steady state MSE of RMED. Finally we see if Mode 3 employing normalization of $\frac{\partial U_k}{\partial \mathbf{W}}$ and $\frac{\partial V_k}{\partial \mathbf{W}}$ simultaneously yields both of the two performance enhancements; faster convergence and lowered steady state MSE.

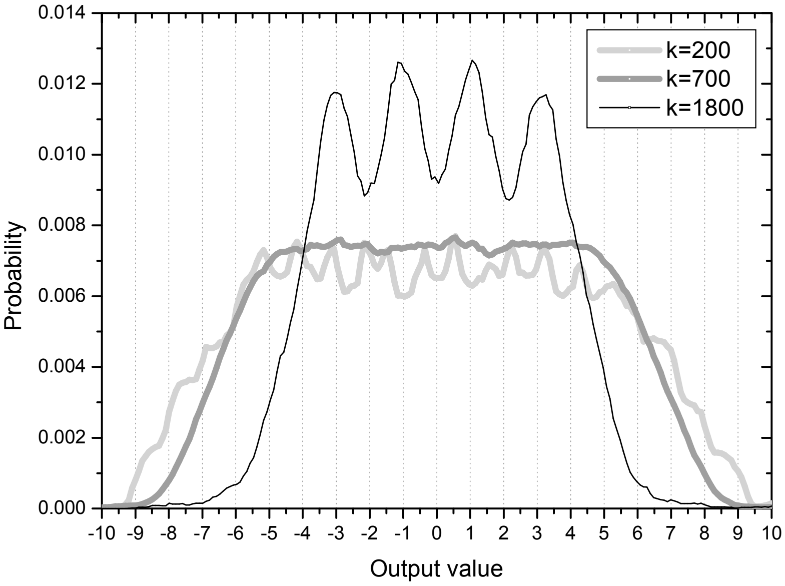

Figure 4. Output distributions in an initial stage.

Figure 5 shows the MSE learning performance for CMA, LMS, RMED and Mode 1 of the proposed algorithm. As discussed in Section 2, the learning curves of the MSE-based algorithms, CMA and LMS do not fall down below −6 dB being defeated by the impulsive noise. On the other hand, the RMED and proposed algorithm show a rapid and stable convergence. The difference of convergence speed between RMED and Mode 1 is clearly observed. While the RMED converges in about 4000 samples, the Mode 1 does in about 2000 samples. Therefore, Mode 1 shows faster convergence than the RMED algorithm by 2 times verifying the analysis of the role of U_k since only $\frac{\partial U_k}{\partial \mathbf{W}}$ is normalized but $\frac{\partial V_k}{\partial \mathbf{W}}$ is not, and we see little difference (about 1 dB) in the steady state MSE.

In Figure 6 RMED and Mode 2 are compared. Both algorithms have similar convergence speed with difference of only 500 samples. But after convergence the Mode 2 yields much lower steady state MSE than the original RMED by over 2 dB. These findings indicate that the role of V_k is definitely related with lowering minimum MSE. This is in accordance with the analysis that U_k plays the role of pulling error samples close together.

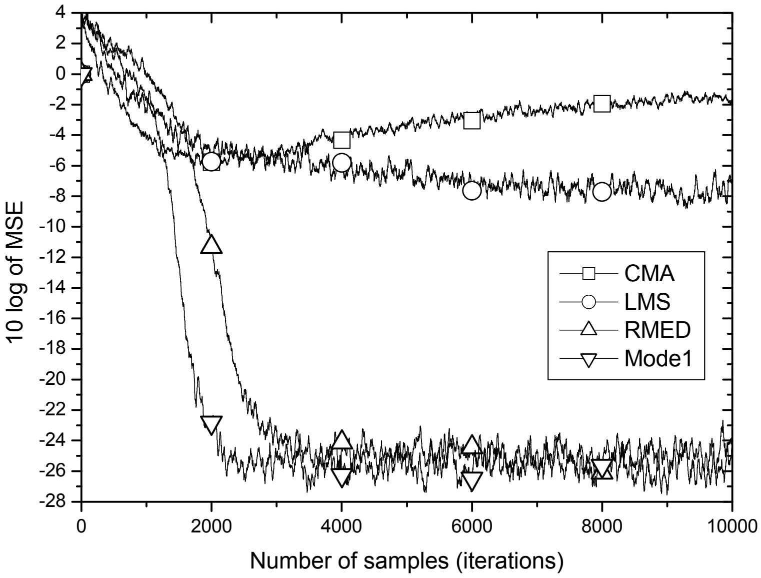

Figure 5. MSE convergence performance for U_k normalization.

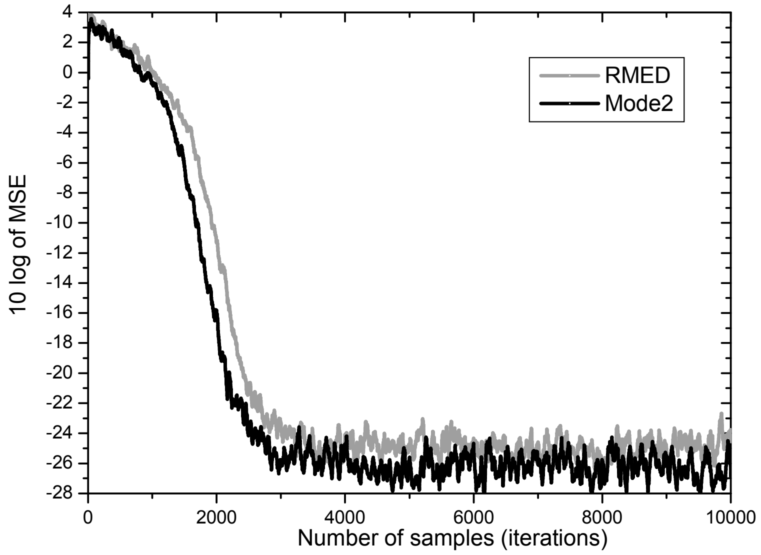

Figure 6. MSE convergence performance for normalization of V_k.

Furthermore, Mode 3 employing normalization of $\frac{\partial U_k}{\partial \mathbf{W}}$ and $\frac{\partial V_k}{\partial \mathbf{W}}$ simultaneously proves to yield the two merits of performance enhancement revealing increased speed and lowered steady state MSE as depicted in Figure 7. While the RMED converges in about 4000 samples and leaves its steady state MSE at about 25 dB, the Mode 3 converges in about 2000 samples and has about 27 dB of steady state MSE. By employing Mode 3, we obtained faster convergence by about 2 times and lower steady state MSE by over 2 dB.

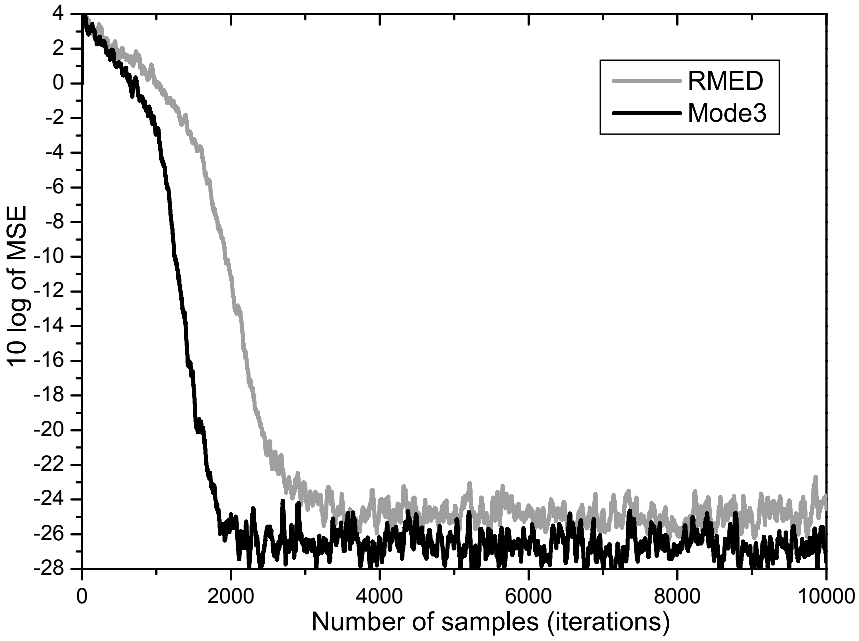

Figure 7. MSE convergence performance for normalization of both U_k and V_k.

In Mode 3, it is still not clear whether the normalization to U_k for speeding up the initial convergence may have a negative influence in later iterations, so we try to reduce the U_k normalization gradually after convergence ($k \geq 3000$) by using $P_U^{\circ}(k)$ in place of $P_U(k)$ as:

$$P_U^{\circ}(k) = P_U(k) \cdot c^{k-3000} + (1 - c^{k-3000}) \tag{33}$$

where $k \geq 3000$ and a constant c is $0 \leq c \leq 1$.

The results for c = 0.8, 0.9, 0.99, 1.0 are shown in Figure 8 in terms of error distribution since the learning curves for the various constant values are not clearly distinguishable.

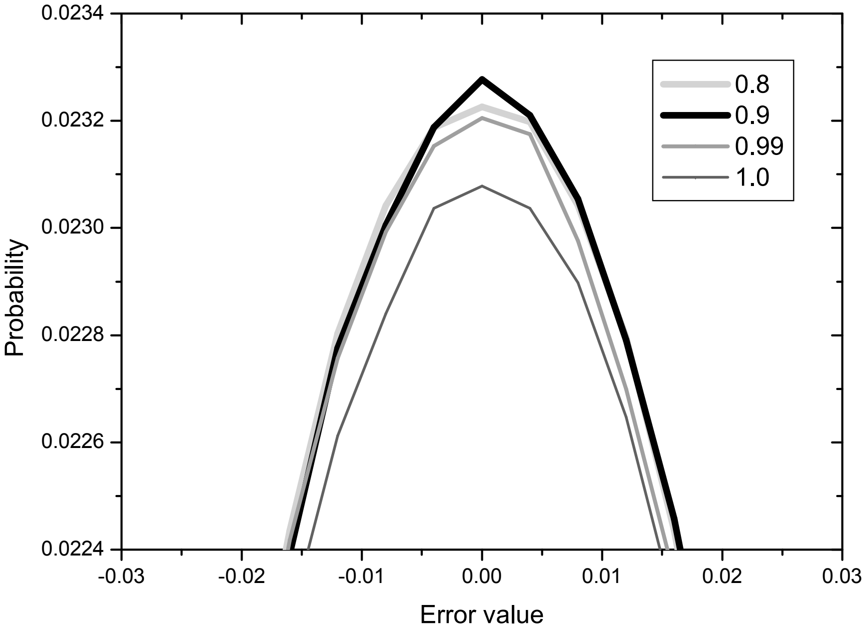

Figure 8. Error distribution with respect to the values of c for normalization of U_k.

The value of c in (33) may be related with the degree of gradual reduction in the normalization to U_k, that is, $c = 1$ indicates no reduction (Mode 3 as it is) and $c = 0.8$ means comparatively rapid reduction. From the Figure 8, we observe that the error performance becomes better and then worse as the degree of reduction decreases from 0.8 to 1.0. This implies that the gradual reduction of the normalization to U_k is effective but not much. We may conclude that the normalization to U_k for speeding up the initial convergence has a slight negative influence in later iterations and this can be overcome by employing the gradual reduction of the U_k normalization.

7. Conclusions

Minimization of the Euclidean distance between output distribution and Dirac delta function as a performance criterion is known to force the distribution of system output to come to a set of delta functions located at each symbol point. In the analysis of the algorithm RMED developed based on that criterion and recursive gradient estimation, it has been revealed in this paper that the minimization process of the cost function uses its two gradients with different functions; one for U_k that forces spreading of output samples and the other one for V_k that compels output samples to move close to symbol points. In order to verify the roles of U_k and V_k explained in the analysis by controlling U_k and V_k separately, we proposed to normalize $\frac{\partial U_k}{\partial \mathbf{W}}$ with the averaged power of entropy-governing input and to normalize $\frac{\partial V_k}{\partial \mathbf{W}}$ with that of input. From the results through simulation for the separate normalization of the gradients of RMED in multipath channel equalization under impulsive noise, faster convergence by about two times through normalization of $\frac{\partial U_k}{\partial \mathbf{W}}$ and lower steady state MSE by over 2 dB by normalization of $\frac{\partial V_k}{\partial \mathbf{W}}$ have been observed. From the analysis and experimental results, we can conclude that U_k is associated with the role of accelerating initial convergence speed by spreading output samples which may have clustered around wrong places in the initial-stage due to channel distortions, and V_k is related with lowering the minimum MSE by pulling error samples close together through the minimization of $C_{\mathrm{ED},k}$. Also it can be concluded that through applying normalization to the two factors $\frac{\partial U_k}{\partial \mathbf{W}}$ and $\frac{\partial V_k}{\partial \mathbf{W}}$ separately with each related input power, significant performance enhancement can be achieved.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Haykin, S. *Adaptive Filter Theory*, 4th ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2001.
2. Principe, J.; Xu, D.; Fisher, J. Information theoretic learning. In *Unsupervised Adaptive Filtering*; Haykin, S., Ed.; Wiley: New York, NY, USA, 2000.
3. Erdogmus, D.; Rao, Y.; Principe, J. Supervised training of adaptive systems with partially labeled data. In Proceedings of the International Conference on ASSP, Marrakech, Morocco, 9–15 April 2005; pp. 321–324.
4. Soleimani, H.; Tomasin, S.; Alizadeh, T.; Shojafar, M. Cluster-head based feedback for simplified time reversal prefiltering in ultra-wideband systems. *Phys. Commun.* **2017**, *25*, 100–109. [CrossRef]
5. Ahmadi, A.; Shojafar, M.; Hajeforosh, S.F.; Dehghan, M.; Singhal, M. An efficient routing algorithm to preserve *k*-coverage in wireless sensor networks. *J. Supercomput.* **2014**, *68*, 599–623. [CrossRef]
6. Jeong, K.; Xu, J.W.; Erdogmus, D.; Principe, J.C. A new classifier based on information theoretic learning with unlabeled data. *Neural Netw.* **2005**, *18*, 719–726. [CrossRef] [PubMed]
7. Kim, N.; Kang, M. Blind signal processing algorithms based on recursive gradient estimation. *Int. J. Electr. Comput. Eng.* **2015**, *5*, 548–561.
8. Treichler, R.; Agee, B. A new approach to multipath correction of constant modulus signals. *IEEE Trans.* **1983**, *31*, 349–372. [CrossRef]
9. Bharani, L.; Radhika, P. FPGA implementation of optimal step size NLMS algorithm and its performance analysis. *Int. J. Res. Eng. Technol.* **2013**, *2*, 885–890.
10. Chinaboina, R.; Ramkiran, D.; Khan, H.; Usha, M.; Madhav, B.; Srinivas, K.; Ganesh, G. Adaptive algorithms for acoustic echo cancellation in speech processing. *Int. J. Res. Rev. Appl. Sci.* **2011**, *7*, 38–42.

11. Leff, H.S. Thermodynamic entropy: The spreading and sharing of energy. *Am. J. Phys.* **1996**, *64*, 1261–1271. [CrossRef]
12. Proakis, J. *Digital Communications*, 2nd ed.; McGraw-Hill: New York, NY, USA, 1989.
13. Santamaria, I.; Pokharel, P.; Principe, J. Generalized correlation function: Definition, properties, and application to blind equalization. *IEEE Trans. Signal Process.* **2006**, *54*, 2187–2197. [CrossRef]