2007 International Symposium on Nonlinear Theory and its Applications
NOLTA'07, Vancouver, Canada, September 16-19, 2007

Recovery of Chaotic Signals Using On-line ICA Algorithm

Song-Ju Kim†, Ken Umeno†‡, and Ryo Takahashi‡

†National Institute of Information and Communications Technology
Nukui-kitamachi 4-2-1, Koganei-shi, Tokyo 184-8795 JAPAN
‡Next Generation Mobile Communications Laboratory, Center for Intellectual Property Strategies, RIKEN
Hirosawa 2-1, Wako-shi, Saitama 351-0198 JAPAN
Email: songju@nict.go.jp, umeno@nict.go.jp, and ryo-takahashi@riken.jp

Abstract—Using chaotic signals, we evaluate the performance of the equivariant adaptive separation via independence (EASI) algorithm. We found that the EASI algorithm in fixed-point (16-bit) arithmetic can recover the chaotic signals successfully as well as the algorithm in floating-point arithmetic. This suggests that the EASI algorithm is suitable for hardware implementation.

1. Introduction

Independent component analysis (ICA) for blind source separation (BSS) has recently attracted much attention in various fields, such as biomedical signal processing (EEG/MEG signals), audio, acoustics, and image enhancement systems, and wireless telecommunication systems [1]. The ICA algorithms can decompose observed signals into statistically independent components. Therefore, we can recover the original source signals $\mathbf{s}(t)$ from the observed signals $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$ if the original source signals are mutually independent ($\mathbf{A}$ is an unknown mixing matrix).

In this paper, we use chaotic signals generated by Chebyshev map as the original source signals because these signals are mutually independent. Recently, we originally found that the chaotic signals recovered by ICA are very useful as spreading sequences in code division multiple access (CDMA) [2]. The signal-to-interference ratio (SIR) of the recovered signals is much larger than those of the original signals although the waveforms of the recovered signals are almost the same as those of the original signals [3].

Using chaotic source signals, we evaluate the performance of the equivariant adaptive separation via independence (EASI) algorithm proposed by Cardoso et al. [4]. The EASI algorithm have simple parallel structure, and may be suitable for hardware implementation. Toward a hardware implementation, we also investigate the performance of the algorithm in 16-bit fixed-point arithmetic.

2. EASI algorithm

In ICA algorithms, the basic goal is to find the separating matrix $\mathbf{W}$, such that $\mathbf{y}(t) = \mathbf{W}\ \mathbf{x}(t)$, without knowing the mixing matrix $\mathbf{A}$. Here, $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$ are observed signals or mixed signals, and $\mathbf{y}(t)$ is a scaled and permuted version of the original source signals $\mathbf{s}(t)$. That is, the equation $\mathbf{WA} = \mathbf{\Lambda P}$ holds, where $\mathbf{\Lambda}$ is an diagonal matrix and $\mathbf{P}$ is a permutation matrix.

Many on-line ICA algorithms have been proposed so far. We focus on the EASI algorithm, which includes the natural gradient [5, 6], because the other effective algorithms are mostly based on this algorithm [7, 8, 9].

Caldoso et al. proposed the following EASI algorithm [4],

$$\begin{aligned}\mathbf{W}(t+1) &= \mathbf{W}(t) - \mu\mathbf{V}(t)\mathbf{W}(t), & (1)\\ \mathbf{V}(t) &= \mathbf{y}(t)\cdot\mathbf{y}(t)^T - \mathbf{I} \\ &+ g(\mathbf{y}(t))\cdot\mathbf{y}(t)^T - \mathbf{y}(t)\cdot g(\mathbf{y}(t))^T. & (2)\end{aligned}$$

We use $g(y) = -tanh(y)$ and $\mu = 0.001953125$ ($= 2^{-9}$).

As the original source signals, we use the chaotic signals generated by Chebyshev map. Each signal is defined as follows:

$$s(t+1) = T_q(s(t)), \quad q \geq 2. \tag{3}$$

Here, $T_q(x)$ is the q-th order Chebyshev polynomial defined by $T_q(\cos\theta) = \cos(q\theta)$. It is known that this Chebyshev map is ergodic and it has the ergodic invariant measure

$$\rho(x)dx = \frac{dx}{\pi\sqrt{1-x^2}}, \tag{4}$$

and it satisfies the orthogonal relation

$$\int_{-1}^{1} T_i(x)T_j(x)\rho(x)dx = \delta_{i,j}\frac{1+\delta_{i,0}}{2}, \tag{5}$$

where $\delta_{i,j}$ is the Kronecker delta function.

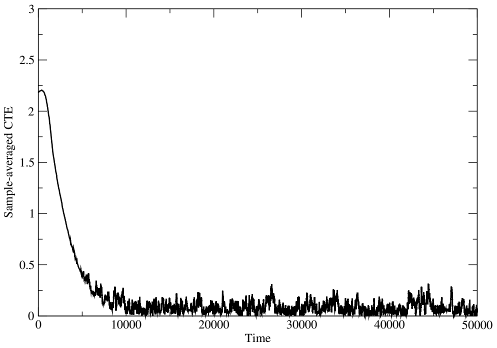

Figure 1: The typical behavior of $< CTE(t) >_{sample}$ in $N = 2$ case. $CTE(t)$s are sample-averaged (50 samples which have different mixing matrix $\mathbf{A}$).

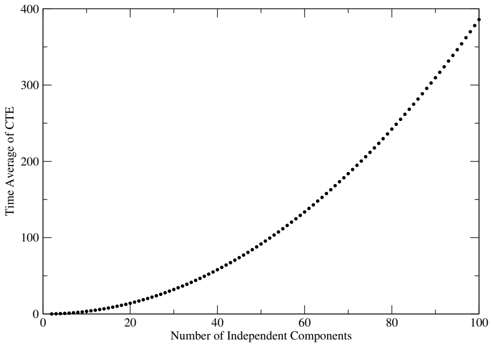

Figure 2: The time-averaged CTEs at each number of independent components N.

3. Performance evaluation

In this paper, we adopt following two indexes in order to evaluate the performance of the EASI algorithm. First index is cross-talking error (CTE) defined as,

$$\begin{aligned} CTE(t) &= \sum_{i=1}^{n}(\sum_{j=1}^{n}\frac{|C_{ij}(t)|}{\max_k |C_{ik}(t)|}-1) \\ &\quad +\sum_{j=1}^{n}(\sum_{i=1}^{n}\frac{|C_{ij}(t)|}{\max_k |C_{kj}(t)|}-1). \end{aligned} \quad (6)$$

Here, $\mathbf{C}(t) = \mathbf{W}(t)\mathbf{A}$. Second index is average distance (AD) defined as,

$$\begin{aligned} AD_{+} &= < |y(t)-s(t)| >_{time} \\ AD_{-} &= < |y(t)+s(t)| >_{time} \\ AD &= min(AD_{+}, AD_{-}). \end{aligned} \quad (7)$$

Here, $<>_{time}$ denotes time average in steady-state. $y(t)$ is a recovered signal of $s(t)$.

As the original source signals, we prepare N chaotic signals, where j-th source signal is generated by the $(j+1)$-th order Chebyshev polynomial and a random initial condition. In case of $N = 2$, we have two original source signals which have the mapping forms $s_1(t+1)$ $=T_2(s_1(t))$ $=2s_1(t)^2 - 1$ and $s_2(t+1)$ $=T_3(s_2(t))$ $=4s_2(t)^3 - 3s_2(t)$, respectively. Here, each source signal $s_j(t)$ is normalized such that $< s_j(t) >_{time}= 0$ and $< s_j^2 >_{time}= 1$.

Figure 1 shows typical behavior of $< CTE(t) >_{sample}$ in $N = 2$ case. After the transition period, $CTE(t)$ becomes stable at low level. Here, we call this state “steady-state”. The time-averaged CTEs in steady-state are shown in figure 2. The dependency on N (number of independent components) is 0.019 $N(N-1)$ as a result of fitting analysis.

If the ICA algorithm successfully recover the original source signals that are mutually independent, the CTE should be zero in general. However, this is not true in our case because the following two reasons: 1) in on-line algorithm, each element C_{ij} (W_{ij}) does not have constant value because there are always brand-new inputs. They always librate around the each convergent level. 2) strictly speaking, the original source signals we are using are not mutually independent because we use finite time sequences. Inner product of the original signals $(\sum_{j=1}^{T} s_1(t_j) \cdot s_2(t_j))$ has very small non-zero value, while inner product of the recovered signals $(\sum_{j=1}^{T} y_1(t_j) \cdot y_2(t_j))$ has zero (to be zero). There are some differences between the original signals $\mathbf{s}(t)$ and the recovered signals $\mathbf{y}(t)$. In this case, $\frac{|C_{ij}|}{\max_k |C_{ik}|}$ $(\frac{|C_{ij}|}{\max_k |C_{kj}|})$ have small non-zero values other than maximum elements that have 1. Therefore, the averaged CTEs of EASI have 0.019 $N(N-1)$, since the number of terms of e.q.(6) is $N(N-1)$.

From the above fact, the EASI algorithm can recover the original source signals successfully although CTEs are not zeros. Figure 3 shows the original source signals and the recovered signals in $N = 2$ case. Fig. 3-(a) shows the original source signal IC-1, which generated by second-order Chebyshev map, and the recovered signal EASI-1. Fig. 3-(b) also shows the original source signal IC-2, which generated by third order Chebyshev map, and the recovered signal EASI-2. The return plots of the original source signals (IC-1, IC-2) and the recovered signals (EASI-1, EASI-2) are shown in figure 4. Each horizontal axis denotes $y(t)$, and vertical axis denotes $y(t+1)$ in fig. 4. The mapping forms (T_2 and T_3) are also conserved. These figures show that the EASI algorithm can recover the original source signals successfully in $N = 2$ case. The fact that EASI-1 has opposite sign to the IC-1 means nothing to the success of recovery.

Even though the CTEs have large number in $N = 100$ case, the EASI algorithm can recover the original source signals successfully. Figure 5 shows the results of ICA simulation in $N = 100$ case. Fig. 5-(a)

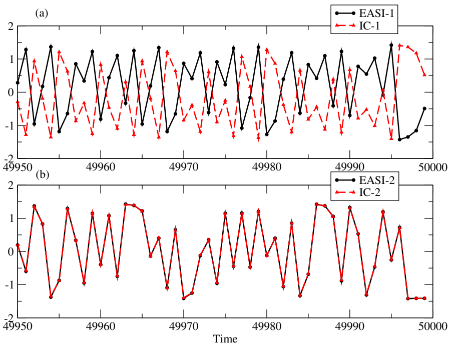

Figure 3: The original source signals and the recovered signals in $N = 2$ case. (a) the original source signal IC-1, which generated by Chebyshev map (second order), and the recovered signal EASI-1. (b) the original source signal IC-2, which generated by Chebyshev map (third order), and the recovered signal EASI-2.

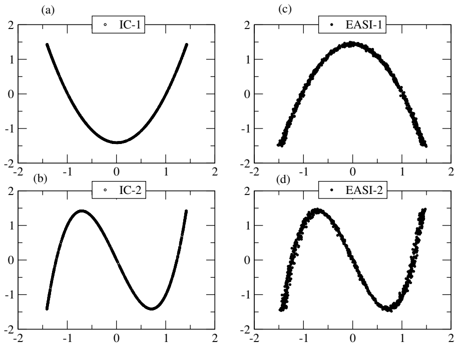

Figure 4: The return plots of the original source signals (a) IC-1, (b) IC-2, and the recovered signals (c) EASI-1, (d) EASI-2, respectively in $N = 2$ case.

shows the correspondence between recovered signals and original signals. For example, first point (1, 67) denotes that the recovered signal EASI-67 is very similar to IC-1. The average distance (AD) of the pair (1, 67) is 0.1714694 as shown in fig. 5-(b). This AD value is minimum in this simulation. The maximum AD is 0.2141588 at number-40 (IC-40 and EASI-2) as shown in fig. 5-(b).

Figure 6 shows signals of these pairs. Fig. 6-(a) shows signals of minimum AD pair (IC-1 and EASI-67), and fig. 6-(b) shows signals of maximum AD pair (IC-40 and EASI-2). We can confirm that the EASI algorithm recover the original source signals successfully even in $N = 100$ case. These ADs are proportional to $0.019\sqrt{N-1}$ as shown in figure 7.

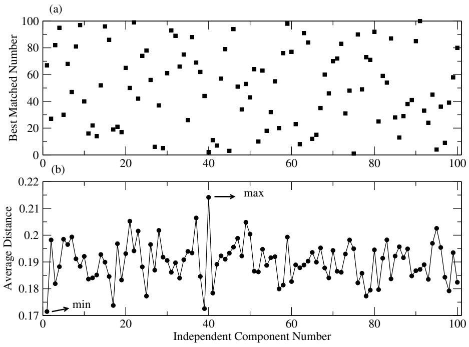

Figure 5: The results of EASI in $N = 100$ case. (a) Correspondence between ICs and EASIs. (b) The average distances (ADs) between ICs and EASIs.

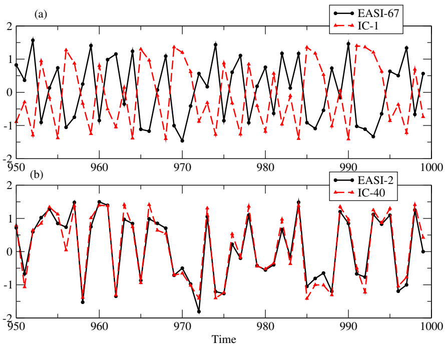

Figure 6: The original source signals and the recovered signals in $N = 100$ case. (a) the original source signal IC-1 and the recovered signal EASI-67 (minimum AD pair). (b) the original source signal IC-40 and the recovered signal EASI-2 (maximum AD pair).

4. Results in fixed-point arithmetic

Toward the hardware implementation of the EASI algorithm, we have to check the performance of the algorithm in fixed-point arithmetic. We used 16-bit fixed-point arithmetic (two's complement arithmetic), and prepared the emulation program written by C. We also used the following approximation function instead of $tanh(y)$, $g(y) = -1$ $(y < -1)$, $g(y) = y$ $(-1 \leq y \leq 1)$, and $g(y) = 1$ $(1 < y)$.

We can confirm that the EASI algorithm recover the original source signals successfully even in fixed-point arithmetic from the comparison between left and right figures in figure 8. The mapping forms (T_2 and T_3) of the recovered signals are conserved in both cases. The ADs in fixed-point arithmetic are almost the same as those in floating-point arithmetic.

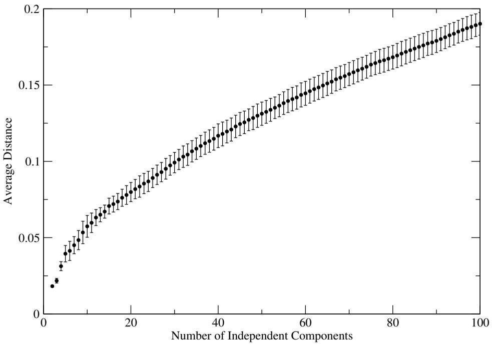

Figure 7: The average distances (ADs) between ICs and EASIs. Each filled circle denotes the average of ADs, and each error-bar denotes the standard deviation of ADs at each number of independent components N.

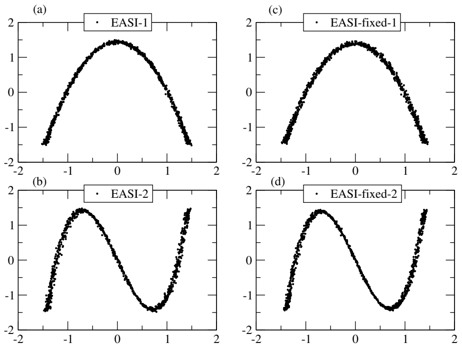

Figure 8: The return plots of the recovered signals in floating-point arithmetic ((a) EASI-1, (b) EASI-2), and in fixed-point arithmetic ((c) EASI-fixed-1, (d) EASI-fixed-2), respectively in $N = 2$ case.

5. Conclusion and discussions

In this paper, we found that the EASI algorithm in fixed-point (16-bit) arithmetic can recover the chaotic signals successfully as well as the algorithm in floating-point arithmetic. This suggests that the EASI algorithm is suitable for hardware implementation.

The chaotic signals recovered by the EASI algorithm are almost the same as the original chaotic signals. This means that properties such as ergodicity and correlation property are conserved. However, its orthogonality is improved due to whitening procedure in the algorithm. Moreover, we can control its orthogonality by changing the learning rate μ [10]. Using the hardware in which the EASI algorithm is implemented, we can prepare the chaotic sequences whose inner products are almost zeros in real-time. We believe that these chaotic sequences are very useful as spreading sequences or channelization codes used for CDMA.

Acknowledgments

This work was financially supported by NEDO Grant for Industrial Technology Research (FY2005-2).

References

[1] K. Umeno, “Chaos and Codes in Communications Systems”, *Proceedings of IPSJ Symposium*, vol.19, pp.312–316, 2005.

[2] R. Takahashi, S. J. Kim, and K. Umeno, “Super Amplification of SNR with an Independent Component Analysis in Chaos CDMA”, *Technical Report of IEICE*, vol.106, no.413, pp.1–6, 2006.

[3] R. Takahashi, S. J. Kim, and K. Umeno, “Analysis of Bit Error Rate Using the Chaos Spreading Sequence with an Independent Component Analysis Filter”, *Technical Report of IEICE*, vol.106, no.573, pp.1–6, 2007.

[4] J.-F. Cardoso and B.H.Laheld, “Equivariant Adaptive Source Separation”, *IEEE Trans. on Signal Processing*, vol.44, pp.3017–3030, 1996.

[5] S. Amari, A. Cichocki, and H. H. Yang, “A New Learning Algorithm for Blind Signal Separation”, *in Advances in Neural Information Processing Systems*, vol.8 (1996), Boston, MA: MIT Press, pp. 752–763.

[6] S. Amari, “Natural Gradient Works Efficiently in Learning”, *Neural Computation*, vol.10 (1998), pp. 251–276.

[7] X. -L. Zhu, J. -F. Chen, and X. -Da Zhang, “Adaptive Blind Source Separation in Order Specified by Stochastic Properties”, *proceedings of ICA2003*, pp.1107–1110, 2003.

[8] W. Y. Leong and J.Homer, “Investigating the Performance Analysis of EASI Algorithm and EKENS Algorithm in Nonlinear Model”, *International Symposium on Communications and Information Technologies*, pp.415–420, 2004.

[9] Z. Zhang and Z. Yi, “An Efficient Independent Component Analysis Algorithm for Sub-Gaussian Sources”, *ISNN 2005, LNCS 3496*, pp.967–972, 2005.

[10] R. Takahashi, S. J. Kim, and K. Umeno, “Generating Complete Orthogonal Spreading Sequences by Equivariant Adaptive Separation via Independence for CDMA”, to appear in proceedings of NDES2007, 2007.