

Linear Complexity Lossy Compressor for Binary Redundant Memoryless Sources

Kazushi MIMURA*

Graduate School of Information Sciences, Hiroshima City University, Hiroshima 731-3194, Japan

(Received June 29, 2011; accepted July 7, 2011; published online August 17, 2011)

A lossy compression algorithm for binary redundant memoryless sources is presented. The proposed scheme is based on sparse graph codes. By introducing a nonlinear function, redundant memoryless sequences can be compressed. We propose a linear complexity compressor based on the extended belief propagation, into which an inertia term is heuristically introduced, and show that it has near-optimal performance for moderate block lengths.

KEYWORDS: nonlinear function, LDGM codes, lossy compression, replica method

Channel coding can be considered as a dual problem of lossy source coding.^{1,2)} Recent research on error correcting codes and lossy source coding has shown that the statistical mechanical approach can be used to explain such problems.³⁾ Lossy compression for memoryless sources has been widely investigated since Matsunaga and Yamamoto showed that it is possible to attain the rate-distortion bound asymptotically using low-density parity-check (LDPC) codes.⁴⁾ The upper⁵⁾ and lower^{6,7)} bounds on its rate-distortion performance of low-density generator-matrix (LDGM) codes for lossy compression with a given check degree are evaluated. Some other lossy compression schemes that have asymptotic optimality have been proposed so far.^{8–19)}

Efficient compressors, on the other hand, are still in the stage of development. Some efficient encoding algorithms, which have near optimal performance, have been proposed, e.g., the nested binary linear codes,²⁰⁾ the inertia-term-introduced belief propagation,^{10,21)} the survey-propagation-based message passing algorithm,²²⁾ the bit-flipping-based algorithm,¹⁴⁾ the exhaustive search of small words into what an original message is divided,²³⁾ the linear-programming-based algorithm,²⁴⁾ and the Markov–Chain Monte–Carlo (MCMC)-based algorithm.²⁵⁾

For redundant memoryless sources, some low complexity compressors, e.g., the near-linear complexity compressor based on the exhaustive search of small words,²³⁾ the quadratic complexity compressor based on the bit-flipping-based algorithm,¹⁴⁾ and the MCMC-based compressor²⁵⁾ whose complexity is independent of the sequence length, have been proposed so far. One of other approaches to obtain near-linear complexity compressors for redundant sources is to apply the inertia-term-introduced belief propagation. Hosaka and Kabashima have proposed an algorithm for redundant sources, whose complexity is $O(N^2)$.¹⁰⁾ In this study, we propose a linear complexity lossy compression algorithm based on an inertia-term-introduced belief propagation by using a nonlinear function and a sparse matrix such as low-density generator matrix (LDGM) codes for binary redundant memoryless sources. This proposed algorithm can be regarded as the perceptron-based one¹⁰⁾ whose edges are extremely deleted to have a finite connectivity and has asymptotic optimality under some constraints. We also show that it has near optimal performance for moderate block lengths.

Let us first provide the concepts of the rate-distortion theory¹⁾ and some notations. Let x be a discrete random variable with an alphabet \mathcal{X} . A source alphabet, a codeword alphabet, and a reproduced alphabet are \mathcal{X} , \mathcal{S} , and $\hat{\mathcal{X}}$, respectively. The compressor \mathcal{F} encodes the M bit source sequence $\mathbf{x} = {}^t(x_1, \dots, x_M) \in \mathcal{X}^M$ into the N ($< M$) bit codeword $\boldsymbol{\xi} = {}^t(\xi_1, \dots, \xi_N) = \mathcal{F}(\mathbf{x}) \in \mathcal{S}^N$. The decompressor \mathcal{G} generates the M bit reproduced sequence $\hat{\mathbf{x}} = {}^t(\hat{x}_1, \dots, \hat{x}_M) = \mathcal{G}(\boldsymbol{\xi}) \in \hat{\mathcal{X}}^M$ from the codeword $\boldsymbol{\xi}$. The code rate then becomes $R = N/M (< 1)$.

A distortion measure is a map $d : \mathcal{X} \times \hat{\mathcal{X}} \rightarrow [0, \infty)$. A distortion between the sequences $\mathbf{x} = {}^t(x_1, \dots, x_M) \in \mathcal{X}^M$ and $\hat{\mathbf{x}} = {}^t(\hat{x}_1, \dots, \hat{x}_M) \in \hat{\mathcal{X}}^M$ is measured by the averaged single-letter distortion as $d(\mathbf{x}, \hat{\mathbf{x}}) = (1/M) \sum_{k=1}^M d(x_k, \hat{x}_k)$. A rate distortion pair (R, D) is said to be achievable if there exists a sequence of rate distortion codes $(\mathcal{F}, \mathcal{G})$ with $\mathbb{E}_x[d(\mathbf{x}, \hat{\mathbf{x}})] \leq D$ in the limit $M \rightarrow \infty$. The rate distortion function $R(D)$ is the infimum of the rate R such that (R, D) is in the rate distortion region of the source for a given distortion D .

We hereafter consider the binary alphabets $\mathcal{X} = \mathcal{S} = \hat{\mathcal{X}} = \{-1, 1\}$ and a redundant binary memoryless source whose distribution is given by $\mu(X = 1) = 1 - p, \mu(X = -1) = p$. The parameter p is a source bias. We use the Hamming distortion

$$d(x, \hat{x}) = \begin{cases} 0, & \text{if } x = \hat{x}, \\ 1, & \text{if } x \neq \hat{x}, \end{cases} \quad (1)$$

as a distortion measure. The rate-distortion function of a Bernoulli(p) i.i.d. source then becomes $R(D) = h_2(p) - h_2(D)$, where h_2 denotes the binary entropy function which is defined by $h_2(x) = -x \log_2(x) - (1-x) \log_2(1-x)$.

Using a parameter $\mathbf{w} = (w_1, w_2) \in (\mathbb{N} \setminus \{0\})^2$, we first introduce the nonlinear function $g : \mathbb{N} \rightarrow \{-1, 1\}$ defined as

$$g_{\mathbf{w}}(z) = g_{(w_1, w_2)}(z) = \begin{cases} 1 & \text{if } w_1 < |z| < w_2, \\ -1 & \text{otherwise} \end{cases}, \quad (2)$$

where the operator tA denotes the transpose of A . For a vector, this function acts componentwise. We here consider the following decompressor:

$$\mathcal{G}(\boldsymbol{\xi}) = g_{\mathbf{w}}(\mathcal{A}\boldsymbol{\xi}), \quad (3)$$

where $\mathcal{A} = (a_{\mu i}) \in \{-1, 0, 1\}^{M \times N}$ denotes a sparse regular matrix whose row weight, i.e., the number of nonzero elements, is C . Each nonzero element of the sparse matrix \mathcal{A} takes ± 1 with equiprobability. The function $g_{\mathbf{w}}(z)$ is adjusted

*E-mail: mimura@hiroshima-cu.ac.jp

so that the bias of the reproduced sequence is close to that of the original message as much as possible. More complex functions can be considered as the function $g_w(z)$. As will be discussed later, the complexity of our proposed algorithm is $O(N)$ but it is proportional to 2^C . So a small C is preferable; therefore, we here consider a simple nonlinear function that can easily be adjusted.

The compressor is then defined by

$$\mathcal{F}(\mathbf{x}) = \operatorname{argmin}_{s \in \{-1,1\}^N} d(\mathbf{x}, \mathcal{G}(s)). \quad (4)$$

This compressor is identical to $\mathcal{F}(\mathbf{x}) = \operatorname{argmax}_{s \in \{-1,1\}^N} p(s; \mathbf{x})$, which is the maximization of the following distribution

$$p(s; \mathbf{x}) \triangleq \frac{1}{Z(\beta)} e^{-\beta M d(s; \mathbf{x})} = \frac{1}{Z(\beta)} \prod_{k=1}^M e^{-\beta G_k(s; x_k)}, \quad (5)$$

where $G_k(s; x_k) = (1/2)(1 - x_k \hat{x}_k)$, $\hat{x}_k = g_w(\sum_{i \in \mathcal{L}(k)} a_{ki} s_i)$ and $\mathcal{L}(k) = \{i | a_{ki} \neq 0\}$ with the parameter $\beta > 0$. Here, $Z(\beta)$ denotes an normalization constant of $p(s; \mathbf{x})$, which is defined by $Z(\beta) = \sum_s e^{-\beta M d(s; \mathcal{G}(s))}$. The function $G_k(s; x_k)$ represents a distortion with respect to the k th bit.

We here consider a large row weight limit, which is a case where $C = N$ holds, to allows us to infer compression performance. In this limit our scheme can be regarded as the perceptron-based code.^{9,10,17,18,26} When $w_2 > N$, these are equivalent to each other. To make the parameters w_1 and w_2 be of order unity, we introduce a constant into the decompressor as $\mathcal{G}(\xi) = g_w(N^{-1/2} \mathcal{A}\xi)$. The achievable distortion D is evaluated as $D = \lim_{\beta \rightarrow \infty} \partial[\beta f(\beta)]/\partial\beta$ via the free energy density $f(\beta) = (-\beta M)^{-1} \mathbb{E}_{\mathcal{A}, \mathbf{x}}[\ln Z(\beta)]$, where \mathbb{E} denotes an expectation operator.

Applying the so called replica method, the free energy density can be evaluated as $f(\beta) = -\beta^{-1}(\ln \{e^{-\beta} + (1 - e^{-\beta})K_w\} + (1 - p) \ln \{e^{-\beta} + (1 - e^{-\beta})(1 - K_w)\} + R \ln 2)$ within the replica symmetric treatment, where $K_w = \int_{\{z \in \mathbb{R} | g_w(z) = -1\}} (2\pi)^{-1/2} e^{-z^2/2} dz$. The parameter K_w is identical to the expectation value $\mathbb{E}_z[g_w(z) = -1]$ with a random variable z which obeys the standard normal distribution $\mathcal{N}(0, 1)$, which originates from the distribution of each element of $N^{-1/2} \mathcal{A}\xi$. It can be considered that the compression performance is given using the distribution of $N^{-1/2} \mathcal{A}\xi$ and g_w in this scheme.

The entropy density of $p(s; \mathbf{x})$ is then obtained as $s(\beta) = \beta(\partial[\beta f(\beta)]/\partial\beta - f)$. The entropy density $s(\beta)$ must be non-negative owing to the definition of $p(s; \mathbf{x})$; however it takes negative values in the large β region. We therefore evaluate the achievable distortion D at β_c which gives a zero entropy density [$s(\beta_c) = 0$], so this analysis is equivalent to the Krauth-Mézard approach which is a kind of one-step replica symmetric breaking (RSB) treatment.²⁷

Using the zero-entropy-density criterion, minimizing the achievable distortion $D = \lim_{\beta \rightarrow \beta_c} \partial[\beta f(\beta)]/\partial\beta$ with respect to w , one can obtain $R(D) = h_2(p) - h_2(D)$, which is identical to the rate-distortion function. From these two conditions, i.e., the zero entropy density and the minimization of the achievable distortion, the following relationships are obtained:

$$e^{-\beta} = \frac{D}{1 - D}, \quad (6)$$

$$K_w = \frac{p - D}{1 - 2D}. \quad (7)$$

The definition of the compressor means that it has exponential complexity. We then utilize a suboptimal algorithm based on message passing to construct the compressor.²¹ Instead of the maximization of $p(s; \mathbf{x})$ we use a symbol MAP encoding scheme, which is maximization of marginal distribution,

$$\xi_i = \operatorname{argmax}_{s_i \in \{-1,1\}} \sum_{s \setminus s_i \in \{-1,1\}^{N-1}} p(s; \mathbf{x}). \quad (8)$$

To evaluate the marginal distribution we apply the belief propagation. Since $\mathcal{G}(-s) = \mathcal{G}(s)$ holds for any s , the expectation value of s_i becomes zero. To avoid this uncertainty we heuristically introduce an inertia term as a prior, which gives the following *inertia-term-introduced belief propagation*.²¹

$$\hat{\rho}_{ki}^t(s_i) = \sum_{s_i' \in \mathcal{L}(k) \setminus i} e^{-\beta G_k(s; x_k)} \prod_{i' \in \mathcal{L}(k) \setminus i} \rho_{i'k}^t(s_{i'}), \quad (9)$$

$$\rho_{ik}^{t+1}(s_i) = \alpha_{ik} r_i^t(s_i) \prod_{k' \in \mathcal{M}(i) \setminus k} \hat{\rho}_{k'i}^t(s_i). \quad (10)$$

A pseudo marginal can be evaluated as

$$q_i^{t+1}(s_i) = \alpha_i r_i^t(s_i) \prod_{k \in \mathcal{M}(i)} \hat{\rho}_{ki}^t(s_i), \quad (11)$$

where α_{ik} and α_i denote normalization constants and $\mathcal{M}(i) = \{k | a_{ki} \neq 0 \forall \mu\}$. Here, the function $r_i^t(s_i)$ is an introduced prior as the inertia term and the superscript t represents an iteration step. We here consider that s_i is binary, so we can safely put $\rho_{ik}^t(s_i) = (1/2)(1 + m_{ik}(t)s_i)$, $\hat{\rho}_{ki}^t(s_i) = (1/2)(1 + \hat{m}_{ik}(t)s_i)$, $q_i^t(s_i) = (1/2)(1 + m_i(t)s_i)$. We here define a prior as $r_i^t(s_i) = e^{s_i \tanh^{-1}[\gamma m_i(t)]}$, where the parameter γ ($0 \leq \gamma < 1$) denotes the amplitude of the inertia term, which is heuristically chosen. When $\gamma = 0$ [$r_i^t(s_i) = 1$], the inertia-term-introduced belief propagation recovers the conventional belief propagation. It should be noted that the performance does not strongly depend on the detailed shape of the function, if it is an increasing function. It has not yet been investigated how the inertia term works in detail so far; however it is known that the inertia term chooses a single peak in the calculation of the pseudo marginal.

We calculate the equations of the belief propagation, which gives

$$\hat{m}_{ki}(t) = \frac{a_{ki} x_k \left(\tanh \frac{\beta}{2} \right) V_{ki}(t)}{1 + x_k \left(\tanh \frac{\beta}{2} \right) U_{ki}(t)}, \quad (12)$$

$$m_{ik}(t+1) = \tanh \left(\sum_{k' \in \mathcal{M}(i) \setminus k} \tanh^{-1} \hat{m}_{k'i}(t) + \tanh^{-1} \gamma m_i(t) \right), \quad (13)$$

$$m_i(t+1) = \tanh \left(\sum_{k' \in \mathcal{M}(i)} \tanh^{-1} \hat{m}_{k'i}(t) + \tanh^{-1} \gamma m_i(t) \right), \quad (14)$$

where

$$U_{ki}(t) = \sum_{s_{i'} \in \mathcal{L}(k) \setminus i} u_w \left(\sum_{i' \in \mathcal{L}(k) \setminus i} a_{ki'} s_{i'} \right) \times \prod_{i' \in \mathcal{L}(k) \setminus i} \frac{1 + m_{i'k}(t) s_{i'}}{2}, \quad (15)$$

$$V_{ki}(t) = \sum_{s_{i'} \in \mathcal{L}(k) \setminus i} v_w \left(\sum_{i' \in \mathcal{L}(k) \setminus i} a_{ki'} s_{i'} \right) \times \prod_{i' \in \mathcal{L}(k) \setminus i} \frac{1 + m_{i'k}(t) s_{i'}}{2}, \quad (16)$$

$$u_w(x) = \mathbb{I}(-w_2 < x < -w_1) + \mathbb{I}(w_1 < x < w_2) - \mathbb{I}(x < -w_2) - \mathbb{I}(w_2 < x) - \mathbb{I}(-w_1 < x < w_1), \quad (17)$$

$$v_w(x) = \mathbb{I}(x = -w_2) - \mathbb{I}(x = -w_1) + \mathbb{I}(x = w_1) - \mathbb{I}(x = w_2), \quad (18)$$

and $\mathbb{I}(\mathcal{P})$ denotes an indicator function that takes 1 if the proposition \mathcal{P} is true, and 0 otherwise. After t_m iterations, the i th bit of the codeword can be obtained as $\xi_i = \text{sgn}[m_i(t_m)]$ using the mean of the pseudo marginal $m_i(t_m)$. To derive these iterative equations, we use the identity $\hat{x}_k = u_w(\sum_{i' \in \mathcal{L}(k) \setminus i} a_{ki'} s_{i'}) + a_{ki} s_i v_w(\sum_{i' \in \mathcal{L}(k) \setminus i} a_{ki'} s_{i'})$, which holds for any $i \in \mathcal{L}(k)$.

The computational cost of the terms $U_{ki}(t)$ and $V_{ki}(t)$ is $O(2^C)$, which depends only on the row weights C , namely, it is $O(1)$ with respect to N . The complexity of this algorithm is therefore $O(N)$ when the number of iterations t_m is fixed.

Utilizing eqs. (6) and (7) which are obtained in the large-row-weight-limit analysis, we can approximately set all parameters C , w_1 , w_2 , and β of our scheme with finite row weights except γ .

We first consider a setting of the parameter β . Using eq. (6) and the rate-distortion function, we set β as $\beta = \beta_c(p, R)$ for the given the source bias p and the code rate R , where $\beta_c(p, R) = \ln([h_2^{-1}(h_2(p) - R)]^{-1} - 1)$. Here, h_2^{-1} denotes the inverse function of the binary entropy function.

We next consider a setting of the parameters C , w_1 , and w_2 . Each element of the vector $\mathcal{A}s$ is the summation of C Bernoulli random variables $1 - 2\text{Bernoulli}(0.5)$, where $s \in \{-1, 1\}^N$ denotes a candidate of a codeword. This is a similar situation to the row weight limit. To keep the row weight finite, we restrict the row weight as $C \leq C_{\max}$. Using eq. (7) and the rate-distortion function, we set (C, w_1, w_2) as $(C, w_1, w_2) = \text{argmin}_{(C', w_1', w_2') \in D(C_{\max})} |\hat{K}(C', w_1', w_2') - K(p, R)|$, where $\hat{K}(C, w_1, w_2) = \sum_{n \in \{0, \dots, C\}; g_w(C-2n)=-1} 2^{-C} \binom{C}{n}$, $K(p, R) = [p - h_2^{-1}(h_2(p) - R)] / [1 - 2h_2^{-1}(h_2(p) - R)]$, and $D(C_{\max}) = \{(C, w_1, w_2) | 2 \leq C \leq C_{\max}, 0 < w_1 \leq C - 1, w_1 < w_2 \leq C + 1\}$ for the given p and R . Note that the parameter that gives second smallest value might provide better performance.

Lastly, γ is determined by trial and error. In this study, we choose γ only within $\{0.2, 0.3, 0.4, 0.5\}$.

The empirical compression performance is shown in Fig. 1. In this figure, the distortion averaged over 10 runs is plotted as a function of the code rate R for the source bias $p \in \{0.6, 0.7, 0.8, 0.9\}$. The length of codewords is fixed at $N = 420$, and the length of original sequence is adjusted. We here choose $C_{\max} = 8$ and $p > 0.5$. In this figure, the

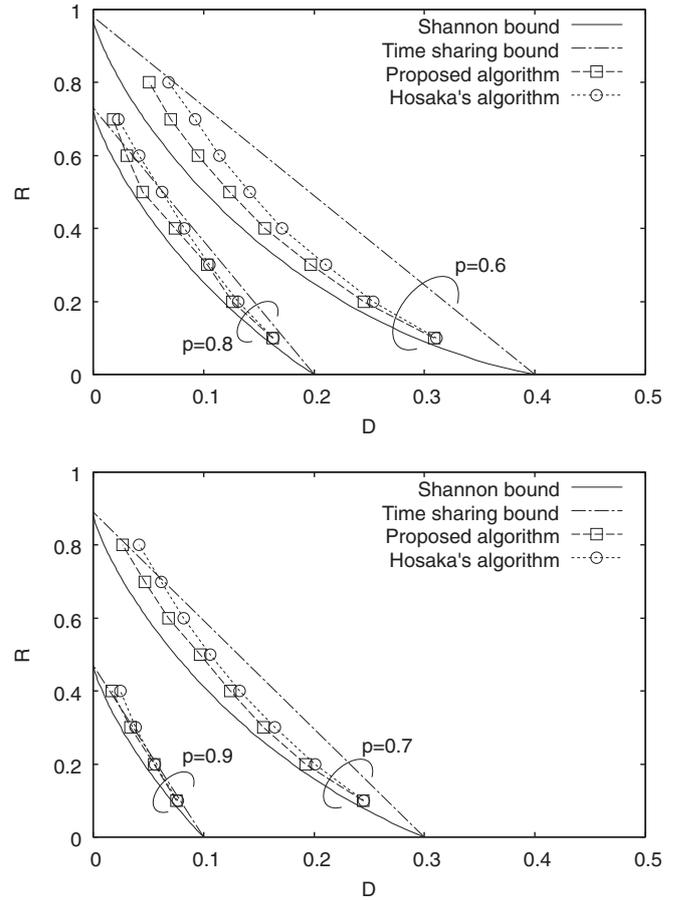


Fig. 1. Empirical compression performance against the code rate R for different source bias p . The proposed algorithm (squares) and Hosaka's algorithm (circles) are shown. The length of the original sequences is $N = 420$, and all the measurements are averaged over 10 runs. The parameter γ is chosen within $\{0.2, 0.3, 0.4, 0.5\}$. The row weight C is chosen within $C \leq 8$ ($C_{\max} = 8$). Top: $p \in \{0.6, 0.8\}$. Bottom: $p \in \{0.7, 0.9\}$.

time sharing bound is also shown. The time sharing bound is given by $R(D) = (1 - D/p)h_2(p)$, which denotes the compression performance achieved by the time sharing scheme of a lossless coding [$R(0) = h_2(p)$] and a trivial encoding that always outputs an all-one vector for any input [$R(p) = 0$]. It can be confirmed that the proposed linear complexity compressor [with $O(N)$ complexity] has slightly better performance than Hosaka's algorithm [with $O(N^2)$ complexity].

We observed a not-so-good performance for the small- p region that the source bias is less than about 0.2. In this region $\min|\hat{K}(C', w_1', w_2') - K(p, R)|$ is not much smaller than that of the large- p region. When we compress the original sequence x of which bias is $p < 0.5$, we can first flip it as $-x$ and then compress. The information for determining whether the sequence flips requires one bit. To reduce $\min|\hat{K}(C', w_1', w_2') - K(p, R)|$, it might be helpful to introduce more complex nonlinear functions.

In this study, we have proposed a scheme using a nonlinear function and a sparse matrix, and as well as a linear complexity message passing compressor based on the inertia-term-introduced belief propagation. The proposed method can treat redundant memoryless sources and has near-optimal compression performance for moderate block

lengths. The adjustment of the column weight distribution of the sparse matrix might enable us to improve the compression performance. The analysis of this scheme with finite row weights is one of our future studies.

Acknowledgments The author would like to thank Tadaaki Hosaka, Tatsuto Murayama, and Jun Muramatsu for their valuable comments and Ankit Gupta, Sergio Verdú, and Tsachy Weissman for providing him valuable preprints. This work was partially supported by a Grant-in-Aid for Scientific Research (C) No. 22500136 from the Ministry of Education, Culture, Sports, Science and Technology of Japan (MEXT).

-
- 1) T. M. Cover and J. A. Thomas: *Elements of Information Theory* (Wiley, New York, 2006) 2nd ed.
 - 2) I. Csiszár and J. Körner: *Information Theory: Coding Theorems for Discrete Memoryless Systems* (Academic Press, New York, 1981).
 - 3) H. Nishimori: *Statistical Physics of Spin Glasses and Information Processing—An Introduction* (Oxford University Press, Oxford, U.K., 2001).
 - 4) Y. Matsunaga and H. Yamamoto: *IEEE Trans. Inf. Theory* **49** (2003) 2225.
 - 5) A. G. Dimakis, M. J. Wainwright, and K. Ramchandran: Proc. Information Theory Workshop, 2005, p. 650.
 - 6) E. Martinian and M. J. Wainwright: Proc. Workshop Information Theory and Its Applications, 2006.
 - 7) M. J. Wainwright: *IEEE Signal Process. Mag.* **24** [5] (2007) 47.
 - 8) T. Murayama and M. Okada: *J. Phys. A* **36** (2003) 11123.
 - 9) T. Hosaka, Y. Kabashima, and H. Nishimori: *Phys. Rev. E* **66** (2002) 066126.
 - 10) T. Hosaka and Y. Kabashima: *Physica A* **365** (2006) 113.
 - 11) S. Ciliberti and M. Mézard: *J. Stat. Mech.* **2005** (2005) P10003.
 - 12) S. Ciliberti, M. Mézard, and R. Zecchina: *Complex Syst. Methods* **3** (2006) 58.
 - 13) K. Mimura and M. Okada: *Phys. Rev. E* **74** (2006) 026108.
 - 14) A. Gupta and S. Verdú: *IEEE Trans. Inf. Theory* **55** (2009) 1961.
 - 15) S. Miyake and J. Muramatsu: *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **E91-A** (2008) 1488.
 - 16) J. Muramatsu and S. Miyake: *Proc. IEEE Int. Symp. Information Theory (ISIT2008)*, 2008, p. 424.
 - 17) F. Cousseau, K. Mimura, T. Omori, and M. Okada: *Phys. Rev. E* **78** (2008) 021124.
 - 18) F. Cousseau, K. Mimura, and M. Okada: *Proc. IEEE Int. Symp. Information Theory (ISIT2008)*, 2008, p. 509.
 - 19) K. Mimura: *J. Phys. A* **42** (2009) 135002.
 - 20) T. Wadayama: Proc. 3rd Int. Symp. Turbo Codes and Related Topics, 2003, p. 231.
 - 21) T. Murayama: *Phys. Rev. E* **69** (2004) 035105(R).
 - 22) M. J. Wainwright and E. Maneva: *Proc. IEEE Int. Symp. Information Theory (ISIT2005)*, 2005, p. 1493.
 - 23) A. Gupta, S. Verdú, and T. Weissman: *Proc. IEEE Int. Symp. Information Theory (ISIT2008)*, 2008, p. 847.
 - 24) S. Miyake, J. Honda, and H. Yamamoto: Proc. IEEE Int. Symp. Information Theory and Its Applications (ISITA2008), 2008, p. 589.
 - 25) S. Jalali and T. Weissman: *Proc. IEEE Int. Symp. Information Theory (ISIT2008)*, 2008, p. 852.
 - 26) K. Mimura, F. Cousseau, and M. Okada: *J. Phys. Soc. Jpn.* **80** (2011) 034802.
 - 27) W. Krauth and M. Mézard: *J. Phys. (Paris)* **50** (1989) 3057.