Squeezing the Arimoto-Blahut Algorithm for Faster Convergence

Yaming Yu, *Member, IEEE*

***Abstract*—The Arimoto–Blahut algorithm for computing the capacity of a discrete memoryless channel is revisited. A so-called "squeezing" strategy is used to design algorithms that preserve its simplicity and monotonic convergence properties, but have provably better rates of convergence.**

***Index Terms*—alternating minimization; channel capacity; discrete memoryless channel; rate of convergence.**

I. Introduction

The Arimoto–Blahut Algorithm [1], [2] (ABA) plays a fundamental role in numerical calculations of channel capacities. This iterative scheme has an appealing geometric interpretation ([6]), and possesses a desirable monotonic convergence property. We refer to [3], [14], [18], [16], [20], [17], [8], [12] for extensions and improvements.

We study variants of ABA with an aim to speed up the convergence while maintaining the simplicity. The focus is on the discrete memoryless channel, and on theoretical properties; extensions and further numerical results will be reported in future works. Our investigation relies on certain reformulations that slightly generalize the original capacity calculation problem. Each formulation leads to an Arimoto-Blahut-type algorithm, which is monotonically convergent, and typically as easily implemented as the original ABA. A formula for the rate of convergence provides valuable insight as to when ABA is slow. Comparison theorems show that our constructions are at least as fast as the usual ABA as measured by the global convergence rate. Numerical examples show that the improvement can be substantial.

Our approach differs from other acceleration methods for ABA (e.g., the proximal point formulations of [12], [15]) in that we focus on preprocessing or "reparameterizing" the problem (Sections III and IV). Such reparameterizations, referred to as "squeezing strategies," aim at reducing the overlap between rows of the channel matrix. Our technical contributions include the monotonic convergence theorem of Section IV, and the convergence rate comparison theorems of Section V. These theoretical results are illustrated with simple examples.

II. Variants of Arimoto-Blahut

A discrete memoryless channel is associated with an $m\times n$ transition matrix $W = (W_{ij})$, where W_{ij} specifies the probability of receiving the output letter j if the input is i. Mathematically W is a stochastic matrix satisfying $W_{ij}\geq 0$ and $\sum_j W_{ij}=1$ for all i. The information capacity is defined as

$$\sup_{p\in\Omega} I(p),\quad I(p)=\sum_i p_i D(W_i\|pW). \tag{1}$$

See [9], [4] for interpretations of this fundamental quantity. Throughout Ω denotes the probability simplex

$$\Omega=\{p=(p_1,\ldots,p_m):\ p_i\geq 0,\ p1_m=1\},$$

1_m denotes the $m\times 1$ vector of ones, W_i denotes the ith row of W, i.e., $W_i = (W_{i1},\ldots,W_{in})$, and $D(q\|r) = \sum_i q_i\log(q_i/r_i)$ for nonnegative vectors $q=(q_i)$ and $r=(r_i)$. We use natural logarithm and adopt the convention $0\log(0/a) = 0,\ a\geq 0$. Let us also define $H(q) = -\sum_i q_i\log q_i$ for a nonnegative vector $q=(q_i)$. It is not required that $\sum_i q_i = 1$. Without loss of generality assume that not all rows of W are equal, and that none of its columns is identically zero.

Iterative algorithms are usually needed for finding the capacity and an optimal input distribution $\hat{p}$, i.e., any maximizer of $I(p)$ in (1). An example among our general class of algorithms is as follows. Let $\lambda\in\mathbf{R}$ satisfy

$$1\leq\lambda\leq\frac{1}{1-\sum_j\min_i W_{ij}}. \tag{2}$$

Algorithm I: Singly Squeezed ABA. Choose $p^{(0)}\in\Omega$ such that $p_i^{(0)}>0$ for all i. For $t=0,1,\ldots$, calculate $p^{(t+1)}$ as

$$p_i^{(t+1)}=\frac{p_i^{(t)}\exp\left(\lambda z_i^{(t)}\right)}{\sum_l p_l^{(t)}\exp\left(\lambda z_l^{(t)}\right)};\quad z_i^{(t)}=D\left(W_i\left\|p^{(t)}W\right.\right). \tag{3}$$

Upon convergence, output $\hat{p}=p^{(\infty)}$.

One recognizes Algorithm I as a generalization of the original Arimoto-Blahut Algorithm, which corresponds to $\lambda=1$. This simple generalization has been considered before (see, e.g., [12]). What is new is the constraint (2). Under this constraint, Algorithm I is guaranteed to converge monotonically (Section IV), and its convergence rate is no worse than that of ABA (Section V). The nickname reflects our intuitive interpretation of Algorithm I and is explained near the end of Section III.

Example 1. Consider the channel matrix

$$W=\begin{pmatrix}0.7 & 0.2 & 0.1\\ 0.1 & 0.2 & 0.7\end{pmatrix}$$

Yaming Yu is with the Department of Statistics, University of California, Irvine, CA, 92697-1250, USA (e-mail: yamingy@uci.edu). Part of this work was presented in the 3rd EPFL/UMLV Workshop on Random Matrices, Information Theory and Applications (Paris, France, Dec. 2009), and in the 2010 IEEE Information Theory Workshop (Cairo, Egypt, Jan. 2010).

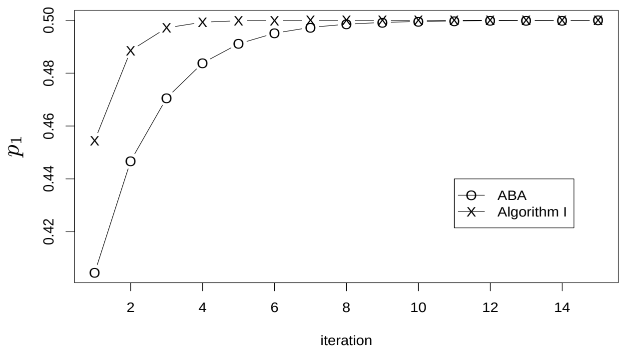

Fig. 1. Iterations of $p_1^{(t)}$ for ABA and Algorithm I with $\lambda = 5/3$.

which is also used by [12] as an illustration. Let us choose $\lambda = 5/3$, which attains the upper bound in (2). Fig. 1 compares the iterations $p_1^{(t)},\ t = 1, 2, \ldots,$ produced by ABA and by Algorithm I with $\lambda = 5/3$. Each algorithm is started at $p^{(0)} = (1/3,\ 2/3)$. Algorithm I, however, appears to approach the target $p^* = (1/2,\ 1/2)$ faster than ABA. Different starting values give similar comparisons. Matz and Duhamel [12] consider an adaptive choice of λ and also report significant improvements over ABA. $\diamondsuit$

Algorithm I is a special case of the following class of algorithms. Henceforth define

$$\Omega(r) = \{p \in \Omega :\ p \geq r\}$$

for any $1 \times m$ vector $r \geq 0$. For vectors (matrices) A and B of the same dimension, $A \geq B$ means every entry of $A - B$ is nonnegative. The $m \times m$ identity matrix is written as I_m.

Let r be a nonnegative $1 \times m$ vector such that

$$W_i \geq rW \quad \text{for all } i = 1, \ldots, m. \tag{4}$$

Define $r_+ = r1_m$. Let λ (a scalar) satisfy

$$\frac{1}{1 - r_+} \leq \lambda \leq \frac{1}{1 - \sum_j \min_i W_{ij}}. \tag{5}$$

Algorithm II: Doubly Squeezed ABA. Choose $p^{(0)} \in \Omega(r)$ such that $p_i^{(0)} > 0$ for all i. For $t = 0, 1, \ldots$, calculate

$$p_i^{(t+1)} = \max \left\{ r_i,\ \delta^{(t)} p_i^{(t)} \exp\left(\lambda z_i^{(t)}\right) \right\} \tag{6}$$

where

$$z_i^{(t)} = D\left(W_i \middle\| q^{(t)} W\right), \quad q^{(t)} = \frac{p^{(t)} - r}{1 - r_+},$$

and $\delta^{(t)}$ is chosen such that $\sum_i p_i^{(t+1)} = 1$. Upon convergence, output

$$\hat{p} = \frac{p^{(\infty)} - r}{1 - r_+}.$$

A stopping criterion for practical implementation is ($\epsilon > 0$)

$$\max_i z_i^{(t)} - \sum_i q_i^{(t)} z_i^{(t)} \leq \epsilon. \tag{7}$$

This is the same criterion as often used for ABA ([2]), and it is convenient since the quantities $z_i^{(t)}$ are readily available at each iteration.

A key requirement is (4). It implies, for example,

$$r_+ \leq \sum_j \min_i W_{ij} < 1,$$

assuming that not all rows of W are equal. When $m = 2$, (4) becomes

$$\frac{r_1}{1 - r_1 - r_2} \leq \min_{j:\ W_{1j} > W_{2j}} \frac{W_{2j}}{W_{1j} - W_{2j}}, \quad \text{and} \tag{8}$$

$$\frac{r_2}{1 - r_1 - r_2} \leq \min_{j:\ W_{2j} > W_{1j}} \frac{W_{1j}}{W_{2j} - W_{1j}}. \tag{9}$$

For general m, the restrictions on r are less clear. See Section V for further discussion.

If $r \equiv 0$, then (6) reduces to (3), showing Algorithm II as a generalization of Algorithm I. Compared with Algorithm I, Algorithm II is only slightly more difficult to implement. In (6), determining $\delta^{(t)}$ is a form of waterfilling ([4]), which can be implemented in $O(m \log m)$ time. (A simple implementation is included in Appendix A for completeness.) Hence the additional cost per iteration is minor. The improvement in convergence rate, however, can be substantial.

Example 1 (continued). Consider Algorithm II with $\lambda = 5/3$ and $r = (1/8,\ 1/8)$. Then (8) and (9) are satisfied with equalities. Inspection of (6) reveals that we have $p^{(1)} = (1/2,\ 1/2)$, regardless of the starting value $p^{(0)}$. (It is easier to verify this with the equivalent form of Algorithm II in Section III.) That is, with this choice of λ and r, Algorithm II converges in one step. $\diamondsuit$

The general validity of Algorithm II is verified in Section IV. The critical issue of which values of r and λ lead to fast convergence is studied in Section V, where theoretical justifications are provided for the following guideline. For fast convergence, we should

- set λ at the upper bound in (5), and
- let $r/(1 - r_+)$ be as large as possible, subject to the restriction (4).

For $m = 2$, this means that r should satisfy the equalities in (8) and (9). Although Example 1 already hints at such a recommendation, we also conduct a simulation for illustration.

Example 2. A channel matrix W with $m = 2$ and $n = 8$ is generated according to $W_{ij} = u_{ij} / \sum_k u_{ik}$ where u_{ij} are independent uniform$(0, 1)$ variates. The original ABA, Algorithm I, and Algorithm II are compared. For Algorithm I, we set λ at the upper bound in (2); for Algorithm II, we choose $r/(1 - r_+)$ to satisfy the upper bounds in (8)–(9), and set λ at the upper bound in (5). The starting values are $p^{(0)} = (1/2, 1/2)$ for ABA and Algorithm I, and $p^{(0)} = (1 - r_+)(1/2, 1/2) + r$ for Algorithm II. We record the number of iterations until the common criterion (7) is met with $\epsilon = 10^{-8}$. The experiment is replicated 100 times.

The improvement in speed by using Algorithm I or Algorithm II is evident from Fig. 2, which displays two bivariate plots of the numbers of iterations. While ABA sometimes takes hundreds of iterations, Algorithm I takes no more than 46, and Algorithm II no more than 20, throughout the 100

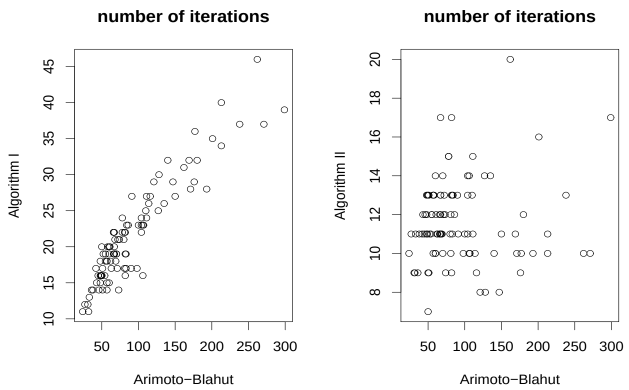

Fig. 2. Comparing the numbers of iterations for three algorithms in Example 2.

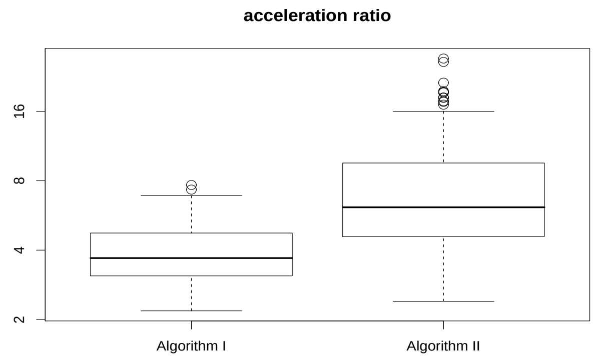

Fig. 3. Acceleration ratios in Example 2.

replications. The large reduction in the number of iterations is also shown in Fig. 3, which summarizes the acceleration ratios relative to ABA, defined as $N_{\mathrm{ABA}}/N_{\mathrm{I}}$ for Algorithm I, for example. Here N_{ABA} (resp. N_{I}) denotes the number of iterations for ABA (resp. Algorithm I). Fig. 3 uses the log scale because the acceleration ratio has a skewed distribution. The median acceleration ratio is 3.7 for Algorithm I, and around 6.1 for Algorithm II. The minimum acceleration ratio is 2.2 for Algorithm I and 2.4 for Algorithm II. Overall this supports the preference for large values of λ and $r/(1-r_+)$, subject to (4) and (5), in implementing Algorithm II. $\diamondsuit$

Remark. One may still implement Algorithm II with some $r,\ \lambda$ that do not satisfy (4) or (5). For example, it is conceivable that values of λ slightly exceeding the upper bound in (5) could lead to even faster convergence. However, our theoretical results only guarantee convergence under (4) and (5). A potentially difficult question is to determine the largest range of (r, λ) that ensures the convergence of Algorithm II.

III. Equivalent Form of Algorithm II

Although Algorithm II is convenient for practical implementation, we write it in an equivalent form (Algorithm III) to study the theoretical properties.

Let r $(1\times m)$ and f $(1\times n)$ be nonnegative vectors that satisfy

$$\tilde{W} \equiv (1+f_+)\frac{I_m - 1_m r}{1-r_+}W - 1_m f \geq 0, \quad r_+ \equiv r 1_m < 1, \tag{10}$$

and $f_+ \equiv f 1_n$. Set

$$c_i = H(\tilde{W}_i) - \frac{1+f_+}{1-r_+}H(W_i), \quad 1 \leq i \leq m. \tag{11}$$

Algorithm III: Doubly Squeezed ABA. Choose $p^{(0)} \in \Omega(r)$ such that $p_i^{(0)} > 0$ for all i. For $t = 0, 1, \ldots$, calculate

$$\Phi_{ji}^{(t)} = \frac{p_i^{(t)}\tilde{W}_{ij}}{f_j + \sum_l p_l^{(t)}\tilde{W}_{lj}}; \tag{12}$$

$$p_i^{(t+1)} = \max\left\{r_i,\ \alpha^{(t)} e^{c_i + \sum_j \tilde{W}_{ij}\log\Phi_{ji}^{(t)}}\right\}, \tag{13}$$

where $\alpha^{(t)}$ is chosen such that $\sum_i p_i^{(t+1)} = 1$. Upon convergence, output

$$\hat{p} = \frac{p^{(\infty)} - r}{1-r_+}.$$

The restriction (10) can be broken down as

$$r_+ < 1, \quad W^* \equiv \frac{I_m - 1_m r}{1-r_+}W \geq 0, \tag{14}$$

and

$$(1+f_+)W^* - 1_m f \geq 0. \tag{15}$$

The restriction (14) is a restatement of (4), while (15) is equivalent to

$$f_j \leq (1+f_+)\min_i W_{ij}^*, \quad j = 1, \ldots, n. \tag{16}$$

If we set

$$\lambda = \frac{1+f_+}{1-r_+}, \tag{17}$$

then Algorithm III reduces to Algorithm II. Indeed, by summing over j, (16) leads to

$$f_+ \leq \frac{1+f_+}{1-r_+}\sum_j\left[\min_i W_{ij} - (rW)_j\right],$$

from which we obtain the upper bound in (5). Moreover, after some algebra, the mapping $p^{(t)} \to p^{(t+1)}$ as specified by (12)–(13) reduces to (6). (A useful identity in this calculation is $p\tilde{W} + f = \lambda(p-r)W$; see also Proposition 1 in Section IV.) Thus Algorithm III reduces to Algorithm II with λ given by (17).

Conversely, suppose r and λ satisfy (4) and (5). If we define

$$f_j = [\lambda(1-r_+) - 1]\frac{\min_i W_{ij}^*}{\sum_k \min_i W_{ik}^*},$$

with W^* given by (14), then (17) is satisfied. We also deduce $f_j \geq 0$ and (16) from (5). Thus Algorithm II is equivalent to Algorithm III with this choice of f.

We shall show that Algorithm II/III converges monotonically, and its convergence rate is no worse than that of ABA. Intuitively, ABA is slow when there exists a heavy overlap between rows of the channel matrix W. Algorithm III, which works with $\tilde{W}$ rather than W, can be seen as trying to reduce this overlap. Its nickname is derived from the transformation (10), which subtracts, or "squeezes out," a nonnegative vector from each row of W. If $r \equiv 0$, then only a vector proportional to f is subtracted. But Algorithm III with $r \equiv 0$ is equivalent to Algorithm II with $r \equiv 0$, which is simply Algorithm I. Hence Algorithm I is called "Singly Squeezed ABA". For general r and f, we squeeze out both a vector proportional to f and another one proportional to rW. Hence Algorithm II/III is called "Doubly Squeezed ABA". The vector r also modifies the space Ω we work on, thus making rW separate from f.

Example 1 (continued). Consider Algorithm III with $r = (1/8, 1/8)$ and $f = (0, 1/4, 0)$. This corresponds to Algorithm II with the same r and $\lambda = 5/3$. By (10) we have

$$\tilde{W} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

The rows of $\tilde{W}$ no longer overlap, i.e., $\tilde{W}_{1j}\tilde{W}_{2j} = 0$ for all j. Inspection of (12) and (13) reveals that we have $\Phi^{(1)} = \tilde{W}^\top$ and $p^{(1)} = (1/2, 1/2)$, regardless of the starting value $p^{(0)}$. Thus, as mentioned earlier, Algorithm II/III converges in one step. $\diamondsuit$

IV. Validity of Algorithm II/III

Given an $m \times n$ stochastic matrix V, a $1 \times n$ vector $f \geq 0$, a $1 \times m$ vector c, and $p \in \Omega$, let us define

$$I(p|V, f, c) = \sum_i p_i(D(V_i \| f + pV) + c_i) + D(f \| f + pV).$$

Equivalently,

$$I(p|V, f, c) = H(pV + f) + \sum_i p_i(c_i - H(V_i)) - H(f). \quad (18)$$

We have $I(p|W, 0, 0) = I(p)$ as in (1). However, there exist less obvious relations. Proposition 1 is key to our derivation of Algorithm III.

Proposition 1: Let $r,\ f,\ \tilde{W}$, and $c = (c_1, \ldots, c_m)$ satisfy (10) and (11). Then

$$I(p|W, 0, 0) = \frac{I(\tilde{p}|\tilde{W}, f, c) + H(f)}{1 + f_+} + \log(1 + f_+) + \frac{\sum_i r_i H(W_i)}{1 - r_+}, \quad (19)$$

where $\tilde{p} = (1 - r_+)p + r$.

Proof: Noting

$$\tilde{p}\tilde{W} + f = (1 + f_+)pW,$$

the claim follows from (18) and routine calculations. ■

Relation (19) implies that, in order to maximize $I(p|W, 0, 0)$ over $p \in \Omega$, we may equivalently maximize $I(\tilde{p}|\tilde{W}, f, c)$ over $\tilde{p} \in \Omega(r)$, and then set $p = (\tilde{p} - r)/(1 - r_+)$. Let us consider solving this slightly more general problem.

Problem I. Let $\tilde{W}$ be an $m \times n$ stochastic matrix, let $f \geq 0$ be a $1 \times n$ vector, and let $r,\ c$ be $1 \times m$ vectors. Assume $r \geq 0$ and $r_+ \equiv r1_m < 1$. Maximize $I(p|\tilde{W}, f, c)$ over $p \in \Omega(r)$.

Problem I can be handled by a straightforward extension of ABA. Following [1], [2], we note that maximizing $I(p|\tilde{W}, f, c)$ is equivalent to maximizing

$$I(p, \Phi) = \sum_{i \geq 1, j} p_i \tilde{W}_{ij} \log \frac{\Phi_{ji}}{p_i} + \sum_j f_j \log \Phi_{j0} + \sum_{i \geq 1} c_i p_i$$

over $p \in \Omega(r)$ and Φ (an $n \times (m+1)$ stochastic matrix) jointly. This holds because, for fixed $p,\ I(p, \Phi)$ is maximized by

$$\Phi_{ji} = \frac{p_i \tilde{W}_{ij}}{f_j + \sum_l p_l \tilde{W}_{lj}}, \quad \Phi_{j0} = \frac{f_j}{f_j + \sum_l p_l \tilde{W}_{lj}}, \quad (20)$$

and the maximum value is $I(p|\tilde{W}, f, c)$. On the other hand, for fixed $\Phi,\ I(p, \Phi)$ is maximized by

$$p_i = \max\left\{r_i,\ \alpha e^{c_i + \sum_j \tilde{W}_{ij} \log \Phi_{ji}}\right\}, \quad (21)$$

where α is chosen such that $\sum_i p_i = 1$. This verifies the Karush-Kuhn-Tucker conditions. If $r \equiv 0$, then (21) reduces to

$$p_i = \frac{\exp(c_i + \sum_j \tilde{W}_{ij} \log \Phi_{ji})}{\sum_{l \geq 1} \exp(c_l + \sum_j \tilde{W}_{lj} \log \Phi_{jl})}. \quad (22)$$

Algorithm III simply alternates between (20) and (21). At each iteration, the function $I(p|\tilde{W}, f, c)$ never decreases. Theorem 1 shows that Algorithm III converges to a global maximum. The proof uses the alternating minimization interpretation of [6]; see Appendix B.

Theorem 1 (Monotonic Convergence): Let $p^{(t)}$ be a sequence generated by Algorithm III. Then $\lim_{t\to\infty} p^{(t)} \equiv p^{(\infty)}$ exists and, as $t \nearrow \infty$,

$$I(p^{(t)}|\tilde{W}, f, c) \nearrow \sup_{p \in \Omega(r)} I(p|\tilde{W}, f, c).$$

By Proposition 1, $(p^{(\infty)} - r)/(1 - r_+)$ is a global maximizer of $I(p|W, 0, 0)$ over $p \in \Omega$. That is, Algorithm III correctly solves the optimization problem (1) in the limit.

V. Rate of Convergence

Throughout this section the notation of Algorithm III is assumed. For example, $\tilde{W}$ is defined via (10). We derive a general formula (Theorem 2) for the rate of convergence. Comparison results (Theorems 3 and 4) show that Algorithm III is at least as fast the original ABA. Based on the comparison theorems, a general recommendation is to let r and f ("the squeezing parameters") be as large as permitted for fast convergence.

Assume the iteration (12)–(13) converges to some p^* in the interior of $\Omega(r)$, i.e., $p_i^* > r_i$ for all i. This rather strong assumption makes our analysis tractable. Denote the mapping from $p^{(t)} \to p^{(t+1)}$ by M. Then $p^* = M(p^*)$, i.e., p^* is a fixed point. We emphasize that, because p^* is assumed to lie in the interior of $\Omega(r)$, so are all $p^{(t)}$ for large enough t. Hence (13) eventually takes the form of (22), i.e.,

$$p_i^{(t+1)} = \frac{\exp(c_i + \sum_j \tilde{W}_{ij} \log \Phi_{ji}^{(t)})}{\sum_{l \geq 1} \exp(c_l + \sum_j \tilde{W}_{lj} \log \Phi_{jl}^{(t)})}.$$

We call $R(p^*) = \partial M(p^*)/\partial p$ the $(m \times m)$ *matrix rate of convergence* of Algorithm III, because

$$p^{(t+1)} - p^* \approx (p^{(t)} - p^*)R(p^*)$$

for $p^{(t)}$ near p^*. The spectral radius of $R(p^*)$, written as $\rho(R(p^*))$, is called the *global rate of convergence*. (The smaller the rate, the faster the convergence.) Such notions are not uncommon in analyzing fixed point algorithms (see, e.g., [7] and [13]). Technically, the global rate should be defined as the spectral radius of a restricted version of $R(p^*)$, because $(p^{(t)} - p^*)1_m = 0$. However, the spectral radius of $R(p^*)$ is the same without this restriction (see Appendix D).

The matrix $R(p^*)$ admits a simple formula (Theorem 2); see Appendix C for its proof.

Theorem 2 (Rate of Convergence): We have

$$R(p^*) = I_m - \tilde{W}\Psi, \quad (23)$$

where the $n \times m$ matrix $\Psi = (\Psi_{ji})$ is specified by

$$\Psi_{ji} = \Phi_{ji}(p^*) + p_i^* \Phi_{j0}(p^*), \quad 1 \le j \le n,\ 1 \le i \le m, \quad (24)$$

and $\Phi_{ji}(p^*)$ is Φ_{ji} as in (20) when taking $p = p^*$.

For the original ABA, we have $R(p^*) = I_m - W\Phi(p^*)$, which can be broadly interpreted as a measure of how noisy the channel is. If $m = n$ and W approaches I_m, then so does $\Phi(p^*)$, and $R(p^*)$ approaches zero. At the opposite end, if rows of W overlap almost entirely, then $W\Phi(p^*)$ is nearly singular, leading to a large $\rho(R(p^*))$, and slow convergence for ABA. See Corollary 1 for a more quantitative statement.

Example 1 (continued). The maximizer of $I(p)$ is $\hat{p} = (1/2,\ 1/2)$. The matrix rates are calculated for ABA (R_0) and for Algorithm III with $r \equiv 0$ and $f = (1/6,\ 1/3,\ 1/6)$ (R_1), which is equivalent to Algorithm I with $\lambda = 5/3$:

$$R_0 = 0.275 \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}; \quad R_1 = 0.125 \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}.$$

The global rates are $\rho(R_0) = 0.55$ and $\rho(R_1) = 0.25$. Thus we confirm the advantage of this choice of λ for Algorithm I. For Algorithm III with $r = (1/8,\ 1/8)$ and $f = (0,\ 1/4,\ 0)$, the global rate is zero. $\diamondsuit$

Propositions 2 and 3 explore basic properties of $R(p^*)$; see Appendix D for the proofs.

Proposition 2: We have

1 $R(p^*)1_m = 0$;
2 if $f \equiv 0$, then $R(p^*)$ is diagonalizable.

Proposition 3: If d is an eigenvalue of $R(p^*)$, then d is real and $0 \le d \le 1$.

Propositions 2 and 3 are used in deriving our main comparison results for convergence rates. Let us write the *global rate* for Algorithm III as $R(r, f)$ to highlight its dependence on the vectors r and f. It is implicit that r and f satisfy (10). When we discuss "the optimal (r, f)," for example, such optimality is always under the constraint (10). The global rate for ABA is simply $R(0, 0)$. The different algorithms under comparison are assumed to deliver the same final output $\hat{p}$.

Theorem 3 presents an exact relation between the global rates for the same r but different f; see Appendix E for its proof.

Theorem 3: We have

$$R(r, f) = (1 + f_+)R(r, 0) - f_+. \quad (25)$$

Consequently, $R(r, f) \le R(r, \tilde{f})$ if $f_+ \ge \tilde{f}_+$.

For fixed r, Theorem 3 recommends large values of f_+ for fast convergence. In view of the constraint (16), this implies that, given r, $R(r, f)$ is minimized by

$$f_j = \frac{\min_i W^*_{ij}}{1 - \sum_k \min_i W^*_{ik}}, \quad 1 \le j \le n, \quad (26)$$

where W^* is defined in (14). This also leads to a nontrivial bound on $R(0, 0)$, the global rate of ABA.

Corollary 1: $R(0, 0) \ge \sum_j \min_i W_{ij}$.

Proof: We have $R(r, 0) \ge f_+/(1 + f_+)$ from (25), since $R(r, f) \ge 0$. The claim follows by choosing $r = 0$ and f as in (26). ■

Corollary 1 formalizes the intuition that ABA is likely to be slow when there exists a heavy overlap between rows of W. The quantity $\sum_j \min_i W_{ij}$ is, in a sense, a conservative measure of this overlap.

To compare the global rates for different values of r, it is convenient to write

$$g = \frac{1 + f_+}{1 - r_+} rW + f. \quad (27)$$

Then f can be recovered from g via

$$f = g - (1 + g_+)rW, \quad g_+ = g1_n. \quad (28)$$

Let us define

$$\tilde{R}(r, g) = R(r, f)$$

in view of this correspondence.

Corollary 2: For fixed r, $\tilde{R}(r, g)$ decreases in g_+.

Proof: Noting

$$f_+ = (1 + g_+)(1 - r_+) - 1, \quad (29)$$

the claim follows from Theorem 3. ■

An advantage of using g is that its optimal choice does not depend on r.

Proposition 4: For fixed r under the constraint (10), $\tilde{R}(r, g)$ is minimized by

$$g_j = \frac{\min_i W_{ij}}{1 - \sum_k \min_i W_{ik}}, \quad 1 \le j \le n. \quad (30)$$

Proof: By direct calculation, (30) follows from (27) and (26). ■

Theorem 4 compares the global rates as a function of r when g is fixed. The proof is presented in Appendix E.

Theorem 4: For fixed g, $\tilde{R}(r, g)$ decreases in $r/(1 - r_+)$, i.e.,

$$\frac{r}{1 - r_+} \ge \frac{\tilde{r}}{1 - \tilde{r}_+} \implies \tilde{R}(r, g) \le \tilde{R}(\tilde{r}, g).$$

Theorem 4 is relatively strong. It implies Corollary 3, as can be verified from Theorem 3 and (27).

Corollary 3: For fixed f, $R(r, f)$ decreases in $r/(1 - r_+)$. Consequently $R(r, f)$ decreases in r.

Overall the function $\tilde{R}(r, g)$ decreases in both $r/(1 - r_+)$ and g. Since the original ABA corresponds to $(r, g) = (0, 0)$, Algorithm III is never worse than the original ABA in terms of the global rate.

Corollary 4: We have

$$R(r, f) \equiv \tilde{R}(r, g) \leq \tilde{R}(0, 0) \equiv R(0, 0).$$

Theorem 4 and Proposition 4 lead to a general rule for choosing the "squeezing parameters". One should choose the largest allowable g as specified by (30), and then choose a large $r/(1 - r_+)$ subject to (4). For $m = 2$ this resolves the optimal choice (subject to (10)) of (r, g) completely.

Corollary 5: If $m = 2$, then $\tilde{R}(r, g)$ is minimized when g satisfies (30) and r satisfies the equalities in (8) and (9).

For general $m > 2$, finding the optimal r appears non-trivial. Fortunately, the optimal r is not strictly necessary for achieving substantial improvements. In Examples 1 and 2, Algorithm I, i.e., $r \equiv 0$, is already considerably faster than ABA. If the optimal r is difficult to find, an option is to fix some $q \in \Omega$, and set $r = \delta q$, where δ is a nonnegative scalar. The constraint (4) reduces to

$$\delta \leq \min_{i,j} \frac{W_{ij}}{(qW)_j}.$$

Then we can set δ at this upper bound. The choice of q, however, remains an open problem.

Remark. Results in this section carry over to Algorithm II since Algorithm III is equivalent to Algorithm II with λ given by (17). By (17), for example, (29) simply says $1 + g_+ = \lambda$. Hence Corollary 2 recommends setting λ at its upper bound in (5). In view of (17), it is not surprising that in Theorem 3 and Corollary 2, the vectors f and g enter the picture only through f_+ and g_+.

VI. Summary and Discussion

A simple "squeezing" strategy is studied for speeding up the Arimoto-Blahut algorithm for discrete memoryless channels. This strategy introduces auxiliary vectors r and f and reformulates the problem so as to reduce the overlap between rows of the channel matrix W. A desirable feature of the resulting Algorithm II/III is that it improves ABA without sacrificing its simplicity or monotonic convergence properties.

The effectiveness of Algorithm II/III is limited by the availability of large values of r and f. If the constraint (10) forces both r and f to be close to zero, then we can expect little improvement from Algorithm II/III. Simply put, some channel matrices are not very "squeezable." Nevertheless, modifications can conceivably be designed for such situations. For example, suppose the input alphabet is ordered so that the overlap between conditional distributions W_i is most severe between adjacent i's. Then a natural strategy is to apply Algorithm II to update the probabilities for one neighborhood of i's at a time, holding the remaining components fixed. Potential applications, e.g., to the discrete-time Poisson channel ([19], [11]), will be reported in future works.

An open problem is to determine the optimal squeezing parameters, i.e., the values of r and f that produce the fastest Algorithm III. While the results in Section V paint a general picture, further theoretical studies may lead to extensions and refinements. If the optimal choice is difficult to derive or to implement, empirical studies may suggest effective rules.

Acknowledgments

The author would like to thank Don Rubin, Xiao-Li Meng and David van Dyk for introducing him to statistical computing and related fields. He is also grateful to the associate editor and the referees for their valuable comments.

Appendix

A: Waterfilling for (6)

We need to determine $\delta \equiv \delta^{(t)}$ such that

$$\sum_i \max\{r_i,\ \delta x_i\} = 1,$$

where $x_i = p_i^{(t)} \exp\left(\lambda z_i^{(t)}\right)$ as in (6). This is feasible with $\delta > 0$ because $\sum_i r_i < 1$.

Step 1. Sort r_i/x_i, say

$$\frac{r_1}{x_1} \leq \frac{r_2}{x_2} \leq \cdots \leq \frac{r_m}{x_m}.$$

Step 2. Calculate the cumulative sums $r_i^* = \sum_{j=i}^m r_j$ and $x_{i*} = \sum_{j=1}^i x_j,\ i = 1, \ldots, m$. By convention $r_{m+1}^* = x_{0*} = 0$.

Step 3. Locate the largest index $i \in \{1, \ldots, m\}$ such that

$$\frac{r_i}{x_i} x_{i*} + r_{i+1}^* \leq 1.$$

Set $\delta = (1 - r_{i+1}^*)/x_{i*}$.

The overall time cost is $O(m \log m)$ due to Step 1.

B: Proof of Theorem 1: Monotonic Convergence

Algorithm III is seen as an alternating divergence minimization procedure between convex sets of measures ([6], [5]). Let $\mathcal{X} = \{0, 1, \ldots, m\}$ and $\mathcal{Y} = \{1, \ldots, n\}$. Let $\mathcal{P}$ be the set of measures on $\mathcal{X} \times \mathcal{Y}$ of the form $P = (P_{ij})$,

$$P_{ij} = \begin{cases} p_i \tilde{W}_{ij}, & 1 \leq i \leq m \\ f_j, & i = 0 \end{cases}$$

where $p = (p_1, \ldots, p_m) \in \Omega(r)$. Let $\mathcal{Q}$ be the set of measures on $\mathcal{X} \times \mathcal{Y}$ of the form $Q = (Q_{ij})$,

$$Q_{ij} = \begin{cases} \Phi_{ji} \tilde{W}_{ij} e^{c_i}, & 1 \leq i \leq m \\ f_j \Phi_{j0}, & i = 0 \end{cases}$$

where $\Phi_{ji} \geq 0$ and $\sum_{i=0}^m \Phi_{ji} = 1$. Observe that (i) both $\mathcal{P}$ and $\mathcal{Q}$ are convex; (ii) $I(p, \Phi) = -D(P\|Q)$; and (iii) (20) and (21) correspond to minimizing $D(P\|Q)$ over Q for fixed P, and over P for fixed Q, respectively. The claim then follows from Theorem 3 of Csiszár and Tusnády [6].

C: Proof of Theorem 2: Convergence Rate

With a slight abuse of notation let $\Phi_{ji}(p)$ and $p_i(\Phi)$ be functions given by (20) and (22) respectively. Then $(1 \leq i, k \leq m,\ 1 \leq j \leq n)$

$$\frac{\partial \Phi_{ji}(p)}{\partial p_k} = \begin{cases} \Phi_{ji}(p)(1 - \Phi_{ji}(p))p_i^{-1}, & k = i; \\ -\Phi_{ji}(p)\Phi_{jk}(p)p_k^{-1}, & k \neq i. \end{cases} \tag{31}$$

$$\frac{\partial p_i(\Phi)}{\partial \Phi_{jk}} = \begin{cases} p_i(\Phi)(1-p_i(\Phi))\tilde{W}_{ij}\Phi_{ji}^{-1}, & k=i; \\ -p_i(\Phi)p_k(\Phi)\tilde{W}_{kj}\Phi_{jk}^{-1}, & k \neq i. \end{cases} \tag{32}$$

We calculate $R(p^*)$ as

$$R(p^*) = \left.\frac{\partial p(\Phi(p))}{\partial p}\right|_{p=p^*}.$$

Write $\Phi^* = \Phi(p^*)$. Then $p(\Phi^*) = p^*$. These relations and (31) and (32) are used repeatedly.

For $i \neq k,\ 1 \leq i, k \leq m$, we have

$$\begin{aligned} \frac{\partial p_i(\Phi(p^*))}{\partial p_k} &= \sum_j \left[\frac{\partial p_i(\Phi^*)}{\partial \Phi_{ji}}\frac{\partial \Phi_{ji}(p^*)}{\partial p_k} + \frac{\partial p_i(\Phi^*)}{\partial \Phi_{jk}}\frac{\partial \Phi_{jk}(p^*)}{\partial p_k}\right] \\ &\quad + \sum_j \sum_{l\geq 1,\, l\neq i,\, l\neq k} \frac{\partial p_i(\Phi^*)}{\partial \Phi_{jl}}\frac{\partial \Phi_{jl}(p^*)}{\partial p_k} \\ &= -\sum_j p_i^* \left[\frac{1-p_i^*}{p_k^*}\tilde{W}_{ij}\Phi_{jk}^* + \tilde{W}_{kj}(1-\Phi_{jk}^*)\right] \\ &\quad + \sum_j \sum_{l\geq 1,\, l\neq i,\, l\neq k} \frac{p_i^* p_l^*}{p_k^*}\tilde{W}_{lj}\Phi_{jk}^* \\ &= -\sum_j \left[(1-p_i^*)\tilde{W}_{kj}\Phi_{ji}^* + p_i^*\tilde{W}_{kj}(1-\Phi_{jk}^*)\right] \\ &\quad + \sum_j p_i^*(1-\Phi_{j0}^* - \Phi_{ji}^* - \Phi_{jk}^*)\tilde{W}_{kj} \qquad (33) \\ &= -\sum_j \tilde{W}_{kj}(\Phi_{ji}^* + p_i^*\Phi_{j0}^*), \qquad (34) \end{aligned}$$

where (33) uses (20).

For $1 \leq k \leq m$, a similar calculation yields

$$\frac{\partial p_k(\Phi(p^*))}{\partial p_k} = 1 - \sum_j \tilde{W}_{kj}(\Phi_{jk}^* + p_k^*\Phi_{j0}^*). \tag{35}$$

Alternatively, (35) can be derived from (34) and

$$\sum_i \frac{\partial p_i(\Phi(p^*))}{\partial p_k} = \frac{\partial \sum_i p_i(\Phi(p^*))}{\partial p_k} = 0. \tag{36}$$

The identity (23) is just (34) and (35) in matrix format.

D: Convergence Rates: Basic Properties

This section proves Propositions 2 and 3. The notation is the same as in Section V. We refer to [10] for tools from matrix analysis.

Part 1 of Proposition 2 follows from (36). For further analysis, define

$$W^* = \frac{I_m - 1_m r}{1-r_+}W, \quad s = p^*W^*, \quad \Delta_s = \mathrm{Diag(s)}.$$

That is, Δ_s is the diagonal matrix with s as the diagonal entries. Similarly, write $\Delta_{p^*} = \mathrm{Diag}(p^*)$. From (20) and (24), we obtain

$$\Psi = \Delta_s^{-1}W^{*\top}\Delta_{p^*}.$$

Thus (23) can be written as

$$R(p^*) = I_m - (1+f_+)K + L \tag{37}$$

where

$$K = W^*\Delta_s^{-1}W^{*\top}\Delta_{p^*}; \quad L = 1_m f\Psi.$$

Observe that $\Delta_{p^*}^{1/2}K\Delta_{p^*}^{-1/2}$ is symmetric and nonnegative definite. Thus K is diagonalizable and has only nonnegative eigenvalues. When $f \equiv 0$, we have $R(p^*) = I_m - K$. Thus $R(p^*)$ is diagonalizable in this case. This proves Proposition 2.

Define a space of row vectors $\Gamma = \{\gamma \in \mathbf{R}^m :\ \gamma 1_m = 0\}$. For an $m\times m$ matrix A such that $\gamma A \in \Gamma$ whenever $\gamma \in \Gamma$, we write $\rho_0(A)$ as the spectral radius of A when restricted as a linear transformation on Γ. Suppose A satisfies $A1_m = 0$, and suppose d is a nonzero eigenvalue of A, with a corresponding left eigenvector γ. Then

$$0 = \gamma A 1_m = d\gamma 1_m \quad \Longrightarrow \quad \gamma \in \Gamma.$$

Hence the set of nonzero eigenvalues is unchanged when A is restricted to Γ. In particular,

$$\rho(A) = \rho_0(A). \tag{38}$$

We have $\gamma L = 0$ for any $\gamma \in \Gamma$. Thus $R(p^*)$ and $I_m - (1+f_+)K$ represent the same linear transformation when restricted to Γ. Also, $R(p^*)1_m = 0$ by Proposition 1. By the preceding discussion, if d is a nonzero eigenvalue of $R(p^*)$, then d is an eigenvalue of $I_m - (1+f_+)K$. Equivalently, $(1-d)/(1+f_+)$ is an eigenvalue of K. We know $d \leq 1$ because K only has nonnegative eigenvalues. On the other hand, $1-d$ is an eigenvalue of $I_m - R(p^*) = \tilde{W}\Psi$, which is a stochastic matrix. The Frobenius-Perron theorem ([10], Chapter 8) implies that $|1-d| \leq 1$, i.e., $d \geq 0$. This proves Proposition 3.

E: Convergence Rates Comparisons

This section proves Theorems 3 and 4, which study how the convergence rate of Algorithm III depends on the squeezing parameters r and f. We adopt the notation of Appendix D. For example, the matrix $R(p^*)$ is the same as in (37).

Proof of Theorem 3: Building on the argument of Appendix D, we have

$$\begin{aligned} R(r,f) &= \rho_0(R(p^*)) && (39) \\ &= \rho_0(I_m - (1+f_+)K) \\ &= (1+f_+)\rho_0(I_m - K) - f_+ && (40) \\ &= (1+f_+)\rho(I_m - K) - f_+ && (41) \\ &= (1+f_+)R(r,0) - f_+. && (42) \end{aligned}$$

Identity (39) follows from (38). Identity (40) holds because, by Proposition 3, the spectral radii involved refer to the largest eigenvalues. Because $(I_m - K)1_m = 0$, we have (41). Identity (42) holds because $I_m - K$ is precisely the matrix rate of Algorithm III that uses $(r,0)$ in place of (r,f). ■

Proof of Theorem 4: By (28) and Theorem 3, we have

$$\begin{aligned} 1 - \tilde{R}(r,g) &= 1 - R(r,\ g - (1+g_+)rW) \\ &= (1+g_+)(1-r_+)(1-R(r,0)). \end{aligned}$$

Thus, to prove $\tilde{R}(r,g) \leq \tilde{R}(\tilde{r},g)$, we only need

$$(1-r_+)(1-R(r,0)) \geq (1-\tilde{r}_+)(1-R(\tilde{r},0)).$$

Let us consider $\tilde{r} \equiv 0$, i.e.,

$$(1-r_+)(1-R(r,0)) \geq 1 - R(0,0). \tag{43}$$

The general case is obtained from (43) (details omitted) if we replace W by
$$\frac{I_m - 1_m\tilde{r}}{1-\tilde{r}_+}W,$$
and r by $r - (1-r_+)\tilde{r}/(1-\tilde{r}_+)$.

By (37), we have
$$R(r,0) = \rho(I_m - UFU^\top \Delta_{p^*})$$
where
$$U = \frac{I_m - 1_m r}{1-r_+}, \quad F = W\Delta_s^{-1}W^\top,$$
$$s = p^*UW = \hat{p}W, \quad p^* = (1-r_+)\hat{p} + r,$$
and $\hat{p}$ denotes the (same) final output of Algorithm III using $(r,0)$ or $(0,0)$ for (r,f). Define
$$A = FU^\top \Delta_{p^*}. \tag{44}$$
Noting $A = (1-r_+)UFU^\top\Delta_{p^*} + 1_m rA$, and following the proof of Proposition 3, we can show that all eigenvalues of A are in the interval $[0,1]$. A calculation similar to that of Theorem 3 gives
$$\rho(I_m - A) = (1-r_+)R(r,0) + r_+. \tag{45}$$
Define
$$\begin{aligned} C &\equiv F^{1/2}r^\top \hat{p}F^{1/2}; \\ \tilde{A} &\equiv F^{1/2}U^\top\Delta_{p^*}F^{1/2} \\ &= F^{1/2}\left(\Delta_{\hat{p}} + \frac{\Delta_r - r^\top r}{1-r_+}\right)F^{1/2} - C. \end{aligned} \tag{46}$$
Comparing (46) with (44) shows that $\tilde{A}$ and A have the same set of eigenvalues. Let a be the smallest eigenvalue of $\tilde{A}$, and let β be a corresponding right eigenvector. Then $a = 1 - \rho(I_m - \tilde{A})$, and by (45),
$$a = 1 - \rho(I_m - A) = (1-r_+)(1-R(r,0)). \tag{47}$$
By direct calculation, we have $\hat{p}F = 1_m^\top$, and hence
$$aC\beta = C\tilde{A}\beta = [(1-r_+)C + F^{1/2}r^\top rF^{1/2}]\beta. \tag{48}$$
If $a = 1 - r_+$, then (48) gives $F^{1/2}r^\top rF^{1/2}\beta = 0$, which implies
$$\beta^\top F^{1/2}r^\top rF^{1/2}\beta = 0; \quad rF^{1/2}\beta = 0; \quad \beta^\top C = 0.$$
Thus,
$$\begin{aligned} a\beta^\top\beta &= \beta^\top\tilde{A}\beta \qquad (49) \\ &= \beta^\top F^{1/2}\left(\Delta_{\hat{p}} + \frac{\Delta_r}{1-r_+}\right)F^{1/2}\beta \\ &\geq \beta^\top F^{1/2}\Delta_{\hat{p}}F^{1/2}\beta \\ &\geq (1-R(0,0))\beta^\top\beta, \qquad (50) \end{aligned}$$
where (50) follows from
$$R(0,0) = \rho(I_m - F\Delta_{\hat{p}}) = \rho(I_m - F^{1/2}\Delta_{\hat{p}}F^{1/2}).$$
We deduce $a \geq 1 - R(0,0)$ and conclude the proof of (43). If $a \neq 1 - r_+$, then (47) implies $a + r_+ - 1 < 0$, and (48) leads to
$$\beta^\top C\beta = \frac{\beta^\top F^{1/2}r^\top rF^{1/2}\beta}{a + r_+ - 1}.$$
Calculations similar to (49)–(50) yield the same conclusion, i.e., $a \geq 1 - R(0,0)$. ■

References

[1] S. Arimoto, "An algorithm for computing the capacity of arbitrary discrete memoryless channels," *IEEE Trans. Inform. Theory*, vol. 18, pp. 14–20, 1972.

[2] R. E. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Inform. Theory*, vol. 18, pp. 460–473, 1972.

[3] C. Chang and L. D. Davisson, "On calculating the capacity of an infinite-input finite (infinite)-output channel," *IEEE Trans. Information Theory*, vol. 34, pp. 1004-1010, 1988.

[4] T. Cover and J. Thomas, *Elements of Information Theory*, 2nd ed., New York: Wiley, 2006.

[5] I. Csiszár and P. Shields, "Information theory and statistics: a tutorial," *Foundations and Trends in Communications and Information Theory*, vol. 1, pp. 417–528, 2004.

[6] I. Csiszár and G. Tusnády, "Information geometry and alternating minimization procedures," *Statistics & Decisions* Supplement Issue 1, pp. 205–237, 1984.

[7] A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum likelihood estimation from incomplete data via the EM algorithm" (with discussion), *J. Roy. Statist. Soc. B*, vol. 39, pp. 1–38, 1977.

[8] F. Dupuis, W. Yu and F.M.J. Willems, "Blahut-Arimoto algorithms for computing channel capacity and rate-distortion with side information," *Proc. 2004 International Symposium on Information Theory*, Chicago, IL, June/July 2004.

[9] R. G. Gallager, *Information Theory and Reliable Communication*, Wiley: New York, 1968.

[10] R.A. Horn and C.R. Johnson, *Matrix Analysis*, Cambridge University Press, 1990.

[11] A. Lapidoth and S. M. Moser, "On the capacity of the discrete-time Poisson channel," *IEEE Trans. Inf. Theory*, vol. 55, no. 1, pp. 303–322, 2009.

[12] G. Matz and P. Duhamel, "Information geometric formulation and interpretation of accelerated Blahut-Arimoto-type algorithms," *Proc. 2004 Information Theory Workshop*, Oct. 2004.

[13] X. L. Meng, "On the rate of convergence of the ECM algorithm," *Ann. Statist.*, vol. 22, no. 1, pp. 326–339, 1994.

[14] H. Nagaoka, "Algorithms of Arimoto-Blahut type for computing quantum channel capacity," *Proc. 1998 International Symposium on Information Theory*, Cambridge, MA, Aug. 1998.

[15] Z. Naja, F. Alberge and P. Duhamel, "Geometrical interpretation and improvements of the Blahut-Arimoto algorithm," *Proc. 2009 IEEE International Conference on Acoustics, Speech, and Signal Processing*, April, 2009.

[16] J. B. Pedersen and F. Topsøe, "Block symmetry in discrete memoryless channels," *Proc. 2002 IEEE Information Theory Workshop*, Bangalore, India, pp. 131–134, 2002.

[17] M. Rezaeian and A. Grant, "A generalization of the Arimoto-Blahut algorithm," *Proc. 2004 International Symposium on Information Theory*, Chicago, IL, June/July 2004.

[18] J. Sayir, "Iterating the Arimoto-Blahut algorithm for faster convergence," *Proc. 2000 International Symposium on Information Theory*, Sorrento, Italy, 2000.

[19] S. Shamai (Shitz), "Capacity of a pulse amplitude modulated direct detection photon channel," *Proc. Inst. Elec. Eng.*, vol. 137, no. 6, pp. 424-430, Dec. 1990, part I (Communications, Speech and Vision).

[20] P. O. Vontobel, "A generalized Blahut-Arimoto algorithm," *Proc. 2003 International Symposium on Information Theory*, Yokohama, Japan, June/July 2003.

Yaming Yu (M'08) received the B.S. degree in mathematics from Beijing University, P.R. China, in 1999, and the Ph.D. degree in statistics from Harvard University, in 2005. Since 2005 he has been an Assistant Professor in the Department of Statistics at the University of California, Irvine.