A Faster Arimoto-Blahut Algorithm via Squeezing

Yaming Yu

Department of Statistics, UC Irvine
Outline

- Discrete memoryless channel: notations
- Arimoto-Blahut
- Squeezing strategy and new algorithms
- Empirical performance: a small simulation
- Monotonic convergence
- Convergence rate comparisons

http://www.ics.uci.edu/~yamingy/
Discrete memoryless channel: notations

- $W = (W_{ij})$: the $m \times n$ stochastic matrix (channel matrix).
  - $W_{ij}$: the probability of receiving the output $j$ if the input is $i$.
  - $W_{ij} \geq 0$ and $\sum_j W_{ij} = 1$ for all $i$. 
Discrete memoryless channel: notations

- $W = (W_{ij})$: the $m \times n$ stochastic matrix (channel matrix).
  - $W_{ij}$: the probability of receiving the output $j$ if the input is $i$.
  - $W_{ij} \geq 0$ and $\sum_j W_{ij} = 1$ for all $i$.

- **Information capacity**:
  \[ \sup_{p \in \Omega} I(p), \quad I(p) = \sum_i p_i D(W_i || pW). \]

- $\Omega = \{ p = (p_1, \ldots, p_m) : p_i \geq 0, \sum p_i = 1 \}$: the probability simplex.

- $W_i$: the $i$th row of $W$.

- $D(f || g) = \sum_i f_i \log(f_i / g_i)$ for nonnegative vectors $f$ and $g$. 
Information capacity:

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

- the maximum mutual information between input and output distributions
Information capacity:

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

- the maximum mutual information between input and output distributions
- the highest rate per channel use at which information can be sent with arbitrarily low probability of error
Information capacity

Information capacity:

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

- the maximum mutual information between input and output distributions
- the highest rate per channel use at which information can be sent with arbitrarily low probability of error
- the optimal prior in a certain “objective” sense in Bayesian statistics
Information capacity:

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

- the maximum mutual information between input and output distributions
- the highest rate per channel use at which information can be sent with arbitrarily low probability of error
- the optimal prior in a certain “objective” sense in Bayesian statistics
- How to calculate this fundamental quantity?
The Arimoto-Blahut Algorithm

- proposed independently by Arimoto (1972) and Blahut (1972)
The Arimoto-Blahut Algorithm

- proposed independently by Arimoto (1972) and Blahut (1972)

- Advantages:
  - simplicity
  - ease of implementation
  - monotonic convergence
  - works for all discrete memoryless channels
The Arimoto-Blahut Algorithm

- proposed independently by Arimoto (1972) and Blahut (1972)

- Advantages:
  - simplicity
  - ease of implementation
  - monotonic convergence
  - works for all discrete memoryless channels

- Disadvantages:
  - can be slow (takes many iterations to converge)
Algorithm 0 (Arimoto-Blahut):

- **Starting value:** $p^{(0)} \in \Omega$ such that $p_i^{(0)} > 0$ for all $i$.

- **Updating rule:**

$$
p_i^{(t+1)} = \frac{p_i^{(t)} \exp\left(z_i^{(t)}\right)}{\sum_l p_l^{(t)} \exp\left(z_l^{(t)}\right)}; \\
z_i^{(t)} = D(W_i \| p^{(t)} W).$$
The Arimoto-Blahut Algorithm

Algorithm 0 (Arimoto-Blahut):

- **Starting value:** $p^{(0)} \in \Omega$ such that $p_i^{(0)} > 0$ for all $i$.
- **Updating rule:**

\[
p_i^{(t+1)} = \frac{p_i^{(t)} \exp(z_i^{(t)})}{\sum_l p_l^{(t)} \exp(z_l^{(t)})}; \quad z_i^{(t)} = D(W_i || p^{(t)} W).
\]

- This is a multiplicative algorithm.
- **Geometric interpretation:** Csiszár and Tusnady (1984).
- **Extensions:** Nagaoka (1998); Vontobel (2003); Dupuis et al. (2004); Rezaeian and Grant (2004).
Squeezing Strategies and New Algorithms

- strategies are based on reparameterization/algebraic manipulation;
- simplicity and monotonic convergence are preserved;
- speed is improved.
Squeezing Strategies and New Algorithms

- strategies are based on reparameterization/algebraic manipulation;
- simplicity and monotonic convergence are preserved;
- speed is improved.

**Algorithm I (Singly Squeezed Arimoto-Blahut):** Choose $\lambda$ such that

$$1 \leq \lambda \leq \frac{1}{1 - \sum_{j} \min_{i} W_{ij}}.$$
Squeezing Strategies and New Algorithms

• strategies are based on reparameterization/algebraic manipulation;

• simplicity and monotonic convergence are preserved;

• speed is improved.

Algorithm I (Singly Squeezed Arimoto-Blahut): Choose $\lambda$ such that

\[ 1 \leq \lambda \leq \frac{1}{1 - \sum_j \min_i W_{ij}}. \]

• Starting value: $p^{(0)} \in \Omega$ such that $p_i^{(0)} > 0$ for all $i$.

• Updating rule:

\[ p_i^{(t+1)} = \frac{p_i^{(t)} \exp\left(\lambda z_i^{(t)}\right)}{\sum_l p_l^{(t)} \exp\left(\lambda z_l^{(t)}\right)}; \quad z_i^{(t)} = D\left(W_i \parallel p^{(t)} W\right). \]
Squeezing Strategies and New Algorithms

- Arimoto-Blahut corresponds to $\lambda = 1$.

- **Theorem (Monotonic Convergence):** For a sequence $p^{(t)}$ generated by Algorithm I, $I(p^{(t)}) \uparrow \sup_{p \in \Omega} I(p)$ as $t \uparrow \infty$.

- **Proposition:** The rate of convergence of Algorithm I improves as $\lambda$ increases.
Squeezing Strategies and New Algorithms

• Arimoto-Blahut corresponds to $\lambda = 1$.

• **Theorem (Monotonic Convergence):** For a sequence $p^{(t)}$ generated by Algorithm I, $I(p^{(t)}) \uparrow \sup_{p \in \Omega} I(p)$ as $t \to \infty$.

• **Proposition:** The rate of convergence of Algorithm I improves as $\lambda$ increases.

*Algorithm I is just as simple, has nice properties, but converges faster.*
Squeezing Strategies and New Algorithms

Example 1.

• Channel matrix

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

(also used by Matz and Duhamel (2004) as an illustration).

• Arimoto-Blahut vs. Algorithm I with \( \lambda = 5/3 \) (which attains the upper bound).
Figure 1: Iterations of $p_1^{(t)}$ for Arimoto-Blahut (ABA) and Algorithm I with $\lambda = 5/3$. 
New Algorithms

Let $r$ be a nonnegative $1 \times m$ vector such that $W_i \geq rW$. (entrywise)

Let $\lambda$ satisfy ($r_+ = \sum r_i$)

\[
\frac{1}{1 - r_+} \leq \lambda \leq \frac{1}{1 - \sum_j \min_i W_{ij}}.
\]
New Algorithms

Let $r$ be a nonnegative $1 \times m$ vector such that $W_i \geq rW$. (entrywise)

Let $\lambda$ satisfy ($r_+ = \sum r_i$)

$$\frac{1}{1 - r_+} \leq \lambda \leq \frac{1}{1 - \sum_{j} \min_i W_{ij}}.$$

Algorithm II (Doubly Squeezed Arimoto-Blahut)

- **Starting value:** $p^{(0)}$ such that $p_{i}^{(0)} > 0$ and $p_{i}^{(0)} \geq r_{i}$ for all $i$. 

Dec. 2009, Paris
New Algorithms

Let $r$ be a nonnegative $1 \times m$ vector such that $W_i \geq rW$. (entrywise)

Let $\lambda$ satisfy ($r_+ = \sum r_i$)

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

Algorithm II (Doubly Squeezed Arimoto-Blahut)

- **Starting value:** $p^{(0)}$ such that $p_i^{(0)} > 0$ and $p_i^{(0)} \geq r_i$ for all $i$.

- **Updating rule:** $p_i^{(t+1)} = \max \left\{ r_i, \delta^{(t)} p_i^{(t)} \exp \left( \lambda z_i^{(t)} \right) \right\}$

  where

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

  and $\delta^{(t)}$ is such that $\sum_i p_i^{(t+1)} = 1$. 

Dec. 2009, Paris
New Algorithms

Let $r$ be a nonnegative $1 \times m$ vector such that $W_i \geq rW$. (entrywise)

Let $\lambda$ satisfy ($r_+ = \sum r_i$)

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

Algorithm II (Doubly Squeezed Arimoto-Blahut)

- **Starting value:** $p^{(0)}$ such that $p_{i}^{(0)} > 0$ and $p_{i}^{(0)} \geq r_i$ for all $i$.

- **Updating rule:**
  
  $$p_i^{(t+1)} = \max \left\{ r_i, \delta^{(t)} p_i^{(t)} \exp \left( \lambda z_i^{(t)} \right) \right\}$$

  where

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

  and $\delta^{(t)}$ is such that $\sum_i p_i^{(t+1)} = 1$.

- **Upon convergence,** output $\hat{p} = (p^{(\infty)} - r)/(1 - r_+).$
Algorithm II

- Convergence Criterion:

\[
\max_i z_i(t) - \sum_i q_i(t) z_i(t) \leq \epsilon.
\]
Algorithm II

• Convergence Criterion:

\[ \max_i z_i(t) - \sum_i q_i(t) z_i(t) \leq \epsilon. \]

• Key requirement: \( W_i \geq rW \).
  - Example: \( m = 2 \)

\[ \frac{r_1}{1 - r_1 - r_2} \leq \min_{j: w_{1j} > w_{2j}} \frac{W_{2j}}{W_{1j} - W_{2j}}, \]

\[ \frac{r_2}{1 - r_1 - r_2} \leq \min_{j: w_{2j} > w_{1j}} \frac{W_{1j}}{W_{2j} - W_{1j}}. \]

  - less clear if \( m > 2 \).
Algorithm II

- **Convergence Criterion:**
  \[ \max_i z_i(t) - \sum_i q_i(t) z_i(t) \leq \epsilon. \]

- **Key requirement:** \( W_i \geq rW. \)
  - Example: \( m = 2 \)
    \[ \frac{r_1}{1 - r_1 - r_2} \leq \min_{j: W_{1j} > W_{2j}} \frac{W_{2j}}{W_{1j} - W_{2j}}, \]
    \[ \frac{r_2}{1 - r_1 - r_2} \leq \min_{j: W_{2j} > W_{1j}} \frac{W_{1j}}{W_{2j} - W_{1j}}. \]
    - less clear if \( m > 2. \)

- **Algorithm I corresponds to** \( r \equiv 0. \)
Algorithm II

- Algorithm I corresponds to $r \equiv 0$.
- Slightly more complicated than Algorithm I.
Algorithm II

- Algorithm I corresponds to $r \equiv 0$.
- Slightly more complicated than Algorithm I.

Example 1 (continued)

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

Consider Algorithm II with

- $\lambda = 5/3$ (largest allowable)
- $r = (1/8, 1/8)$ (largest allowable)
Algorithm II

• Algorithm I corresponds to $r \equiv 0$.

• Slightly more complicated than Algorithm I.

Example 1 (continued)

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

Consider Algorithm II with

• $\lambda = 5/3$ (largest allowable)

• $r = (1/8, 1/8)$ (largest allowable)

Algorithm II converges in one iteration regardless of the starting value!
Algorithm II: properties

- **Theorem (Monotonic Convergence):** For a sequence $p^{(t)}$ generated by Algorithm II, $I((p^{(t)} - r)/(1 - r_+)) \uparrow \sup_{p \in \Omega} I(p)$ as $t \uparrow \infty$. 
Algorithm II: properties

- **Theorem (Monotonic Convergence):** For a sequence $p^{(t)}$ generated by Algorithm II, $I((p^{(t)} - r)/(1 - r_+)) \rightarrow \sup_{p \in \Omega} I(p)$ as $t \rightarrow \infty$.

- **Rate comparisons:** Algorithm II is faster for larger $\lambda$ and $r/(1 - r_+)$. 

Dec. 2009, Paris
Algorithm II: properties

• **Theorem (Monotonic Convergence):** For a sequence $p^{(t)}$ generated by Algorithm II, $I((p^{(t)} - r)/(1 - r_+)) \xrightarrow{t \to \infty} \sup_{p \in \Omega} I(p)$ as $t \to \infty$.

• **Rate comparisons:** Algorithm II is faster for larger $\lambda$ and $r/(1 - r_+)$.  
  – With the same $\lambda$, Algorithm II is no slower than Algorithm I.
Algorithm II: properties

- **Theorem (Monotonic Convergence):** For a sequence $p^{(t)}$ generated by Algorithm II, $I((p^{(t)} - r)/(1 - r_+)) \uparrow \sup_{p \in \Omega} I(p)$ as $t \uparrow \infty$.

- **Rate comparisons:** Algorithm II is faster for larger $\lambda$ and $r/(1 - r_+)$.
  - With the same $\lambda$, Algorithm II is no slower than Algorithm I.

- **Practical Guideline:**
  - set $\lambda$ at its upper bound, and
  - let $r/(1 - r_+)$ be as large as possible, subject to restriction $W_i \geq rW$. 

Dec. 2009, Paris
Example 2. Matrix $W$ with $m = 2$ and $n = 8$ is generated according to
$W_{ij} = \frac{u_{ij}}{\sum_k u_{ik}}$ where $u_{ij}$ are independent uniform$(0, 1)$ variates.

- Arimoto-Blahut: $\lambda = 1$ and $r = 0$.
- Algorithm I: $\lambda$ at its upper bound.
- Algorithm II: $\lambda$ at its upper bound, and $r/(1 - r_+)$ at its upper bound.

Convergence criterion:
\[
\max_i z_i^{(t)} - \sum_i q_i^{(t)} z_i^{(t)} \leq 10^{-8}.
\]

100 replications.
Figure 2: Comparing the numbers of iterations for three algorithms in Example 2.
Figure 3: $\log_2$ acceleration ratios in Example 2.

(Acceleration ratio = num. iter. Arimoto-Blahut/num. iter.)
Theory

• Why monotonic convergence?
• Why faster?
Let $r \ (1 \times m)$ and $f \ (1 \times n)$ be nonnegative vectors that satisfy

$$
\tilde{W} \equiv (1 + f_+) \frac{I_m - 1_m r}{1 - r_+} W - 1_m f \geq 0, \quad r_+ \equiv r1_m < 1,
$$

and $f_+ \equiv f1_n$. Set

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

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

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

$$\tilde{W} \equiv (1 + f_+) \frac{I_m - 1_m r}{1 - r_+} W - 1_m f \geq 0, \quad r_+ \equiv r1_m < 1,$$

and $f_+ \equiv f1_n$. Set

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

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

- **Key observation**: maximizing $I(p|W, 0, 0) \equiv I(p)$ is the same as maximizing $I(p|\tilde{W}, f, c)$. 
Let $r$ (1 $\times$ $m$) and $f$ (1 $\times$ $n$) be nonnegative vectors that satisfy

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

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

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

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

- **Key observation**: maximizing $I(p|W, 0, 0) \equiv I(p)$ is the same as maximizing $I(p|\tilde{W}, f, c)$.

- **Key observation**: Arimoto-Blahut applies to maximizing $I(p|\tilde{W}, f, c)$. 
Let $r$ $(1 \times m)$ and $f$ $(1 \times n)$ be nonnegative vectors that satisfy

$$\tilde{W} \equiv (1 + f_+) \frac{I_m - 1}{1-r_+} W - 1_m f \geq 0, \quad r_+ \equiv r_1^m < 1,$$

and $f_+ \equiv f_1^n$. Set

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

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

- **Key observation**: maximizing $I(p|W, 0, 0) \equiv I(p)$ is the same as maximizing $I(p|\tilde{W}, f, c)$.

- **Key observation**: Arimoto-Blahut applies to maximizing $I(p|\tilde{W}, f, c)$.

- **Key observation**: Arimoto-Blahut converges faster for $\tilde{W}$ since its rows have less overlap.
Equivalent Form of Algorithm II

Arimoto-Blahut applies to maximizing \( I(p|\tilde{W}, f, c) \).

Algorithm III

- Starting value: \( p^{(0)} \) such that \( p_{i}^{(0)} > 0 \) and \( p_{i}^{(0)} \geq r_{i} \) for all \( i \).
- Updating rule:

  \[
  \Phi_{ji}^{(t)} = \frac{p_{i}^{(t)} \tilde{W}_{ij}}{f_{j} + \sum_{l} p_{l}^{(t)} \tilde{W}_{lj}}; \quad p_{i}^{(t+1)} = \max \left\{ r_{i}, \alpha^{(t)} e^{c_{i} + \sum_{j} \tilde{W}_{ij} \log \Phi_{ji}^{(t)}} \right\},
  \]

  where \( \alpha^{(t)} \) is such that \( \sum_{i} p_{i}^{(t+1)} = 1 \).
- Upon convergence, output \( \hat{p} = (p^{(\infty)} - r)/(1 - r_{+}) \).

Algorithm III is equivalent to Algorithm II upon setting

\[
\lambda = \frac{1 + f_{+}}{1 - r_{+}}.
\]
Arimoto-Blahut applies to maximizing $I(p|\tilde{W}, f, c)$.

**Algorithm III**

- **Starting value**: $p^{(0)}$ such that $p^{(0)}_i > 0$ and $p^{(0)}_i \geq r_i$ for all $i$.
- **Updating rule**:

$$\Phi^{(t)}_{ji} = \frac{p^{(t)}_i \tilde{W}_{ij}}{f_j + \sum_l p^{(t)}_l \tilde{W}_{lj}}; \quad p^{(t+1)}_i = \max \left\{ r_i, \alpha^{(t)} e^{c_i + \sum_j \tilde{W}_{ij} \log \Phi^{(t)}_{ji}} \right\},$$

where $\alpha^{(t)}$ is such that $\sum_i p^{(t+1)}_i = 1$.
- **Upon convergence**, output $\hat{p} = (p^{(\infty)} - r)/(1 - r_+)$.

It fits the alternating minimization scheme of Csiszár and Tusnady (1984) – **Monotonic convergence!**
Why faster?

• Fixed point algorithm: $p^{(t+1)} = M(p^{(t)})$

• Matrix rate of convergence: $R(p^*) = \partial M(p^*)/\partial p$ for a fixed point $p^*$

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

• Global rate of convergence: the spectral radius of $R(p^*)$. 
Why faster?

- Fixed point algorithm: \( p^{(t+1)} = M(p^{(t)}) \)
- Matrix rate of convergence: \( R(p^*) = \partial M(p^*)/\partial p \) for a fixed point \( p^* \)
- \( p^{(t+1)} - p^* \approx (p^{(t)} - p^*)R(p^*) \)
- Global rate of convergence: the spectral radius of \( R(p^*) \).

**Theorem (Convergence rate of Algorithm II/III):**

\[
R(p^*) = I_m - \tilde{W}\Psi,
\]

where \( \Psi = (\Psi_{ji}) \) is given by

\[
\Psi_{ji} = \Phi_{ji}(p^*) + p_i^*\Phi_{j0}(p^*), \quad 1 \leq j \leq n, \ 1 \leq i \leq m,
\]

and

\[
\Phi_{ji}(p) = \frac{p_i\tilde{W}_{ij}}{f_j + \sum_l p_l\tilde{W}_{lj}}, \quad \Phi_{j0}(p) = \frac{f_j}{f_j + \sum_l p_l\tilde{W}_{lj}}.
\]
Why faster?

- For Arimoto-Blahut

\[ R(p^*) = I_m - W\Phi(p^*) \]
Why faster?

- For Arimoto-Blahut

\[ R(p^*) = I_m - W\Phi(p^*) \]

- This can be interpreted as measuring how noisy the channel is.
  - If \( m = n \) and \( W \) approaches \( I_m \), then so does \( \Phi(p^*) \), and \( R(p^*) \) approaches zero (fast convergence).
  - If rows of \( W \) overlap almost entirely, then \( W\Phi(p^*) \) is nearly singular, leading to a large spectral radius of \( R(p^*) \) (slow convergence).
Rate comparisons

• **Theorem:** If $d$ is an eigenvalue of $R(p^*)$, then $d$ is real and $0 \leq d \leq 1$.

• Global rates for Algorithm II/III with different “squeezing parameters”: Larger $f$ and larger $r/(1 - r_+)$ are better.

• Proof is algebraic – a more intuitive explanation?
Summary

• Simple improvements of Arimoto-Blahut on its own terms.

• Formula for the convergence rate
Summary

• Simple improvements of Arimoto-Blahut on its own terms.
• Formula for the convergence rate
• Extensions?
• Optimal squeezing parameters?
• Some channel matrices are not so squeezable ... What then?