Estimating the Capacity of the 2-D Hard Square Constraint Using Generalized Belief Propagation

Navin Kashyap

(joint work with Eric Chan, Mahdi Jafari Siavoshani,
Sidharth Jaggi and Pascal Vontobel,
Chinese University of Hong Kong)

Conference on Applied Mathematics
University of Hong Kong
August 23, 2016

1-D RLL Constraints

Let $0 \leq d < k \leq \infty$ be fixed integers (k is allowed to be ∞).

Definition

A binary sequence $x_1 x_2 \ldots x_n \in \{0, 1\}^n$ satisfies the (one-dimensional) (d, k)-runlength-limited (RLL) constraint if any pair of successive 1s in the sequence is separated by at least d and at most k 0s.

1-D RLL Constraints

Let $0 \leq d < k \leq \infty$ be fixed integers (k is allowed to be ∞).

Definition

A binary sequence $x_1 x_2 \ldots x_n \in \{0, 1\}^n$ satisfies the (one-dimensional) (d, k)-runlength-limited (RLL) constraint if any pair of successive 1s in the sequence is separated by at least d and at most k 0s.

Codes consisting of sequences satisfying a (d, k)-RLL constraint are used for writing information on magnetic and optical recording devices such as hard drives and CDs/DVDs.

The maximum rate of such a code is given by the capacity of the (d, k)-RLL constraint, defined as

$$C_{d,k} = \lim_{n \to \infty} \frac{1}{n} \log_2 Z_n^{(d,k)}$$

where $Z_n^{(d,k)}$ denotes the number of binary length-n sequences satisfying the constraint.

The 1-D $(1, \infty)$-RLL Constraint

A binary sequence satisfies the $(1, \infty)$-RLL constraint if it does not contain 1s in adjacent (i.e., consecutive) positions.

Some easy facts about $Z_n := Z_n^{(1,\infty)}$:

- Z_n, $n = 1, 2, 3, \ldots$, forms a Fibonacci sequence

$$Z_1 = 2, \quad Z_2 = 3, \quad \text{and} \quad Z_n = Z_{n-1} + Z_{n-2} \text{ for all } n \geq 3$$

- $C_{1,\infty} := \lim_{n \to \infty} \frac{1}{n} \log_2 Z_n = \log_2 \frac{1+\sqrt{5}}{2} = 0.6942\ldots$

The 2-D Hard Square Constraint

Definition

A binary $m \times n$ array satisfies the 2-D hard square constraint (also called the 2-D $(1, \infty)$-RLL constraint) if no row or column of the array contains 1s in adjacent positions.

Each such array can also be viewed as an independent set in the $m \times n$ grid graph.

The Hard-Square Entropy Constant

Let $Z_{m,n}$ denote the number of such $m \times n$ arrays.

It can be shown (for example, using subadditivity arguments) that the limit

$$\eta = \lim_{m,n\to\infty} \frac{1}{mn} \log_2(Z_{m,n})$$

exists. This limit is called the hard-square entropy constant.

Open Problem: Determine the hard-square entropy constant η.

The Hard-Square Entropy Constant

Let $Z_{m,n}$ denote the number of such $m \times n$ arrays.

It can be shown (for example, using subadditivity arguments) that the limit

$$\eta = \lim_{m,n\to\infty} \frac{1}{mn} \log_2(Z_{m,n})$$

exists. This limit is called the hard-square entropy constant.

Open Problem: Determine the hard-square entropy constant η.

What is known:

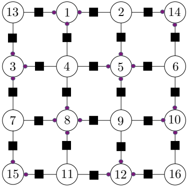
- Various upper and lower bounds, resulting in the numerical estimate

  $$\eta = 0.58789116\ldots$$

  [Justesen & Forchhammer, 1999]

- η is computable to within an accuracy of $\frac{1}{N}$ in time polynomial in N [Pavlov, 2010]

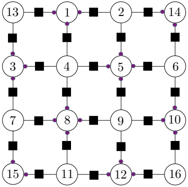
Binary Ising Model on the 2-D Lattice



- $m \times n$ grid with mn vertices and $2mn - (m + n)$ edges
- Variable $x_i \in \{0, 1\}$ at each vertex i
- Pairwise function $f : \{0, 1\} \times \{0, 1\} \to \mathbb{R}_+$
- Defines a joint distribution on $\{0, 1\}^{m \times n}$:

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{i \sim j} f(x_i, x_j)$$

Binary Ising Model on the 2-D Lattice



- $m \times n$ grid with mn vertices and $2mn - (m+n)$ edges
- Variable $x_i \in \{0,1\}$ at each vertex i
- Pairwise function $f : \{0,1\} \times \{0,1\} \to \mathbb{R}_+$
- Defines a joint distribution on $\{0,1\}^{m \times n}$:

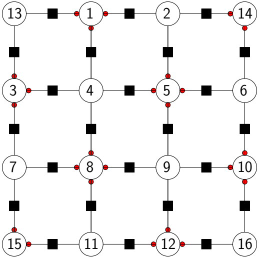
$$p(\mathbf{x}) = \frac{1}{Z} \prod_{i \sim j} f(x_i, x_j)$$

- Quantity of interest:

$$Z = \sum_{\mathbf{x}} \prod_{i \sim j} f(x_i, x_j)$$

Called the partition function.

Special Case: Hard-Square Model



$$f(a,b) = \mathbf{1}_{(a,b)\neq(1,1)}$$

The partition function here is precisely $Z_{m,n}$.

Gibbs Free Energy

Define the energy of a configuration $\mathbf{x} \in \{0,1\}^{m \times n}$ to be

$$E(\mathbf{x}) = -\sum_{i \sim j} \log f(x_i, x_j)$$

so that $p(\mathbf{x}) = \frac{1}{Z} \exp(-E(\mathbf{x}))$.

Gibbs Free Energy

Define the energy of a configuration $\mathbf{x} \in \{0,1\}^{m \times n}$ to be

$$E(\mathbf{x}) = -\sum_{i \sim j} \log f(x_i, x_j)$$

so that $p(\mathbf{x}) = \frac{1}{Z} \exp(-E(\mathbf{x}))$.

For an arbitrary probability distribution $b(\mathbf{x})$ on $\{0,1\}^{m \times n}$, define

- the average energy

$$U(b) = \sum_{\mathbf{x}} b(\mathbf{x}) E(\mathbf{x})$$

- the entropy

$$H(b) = -\sum_{\mathbf{x}} b(\mathbf{x}) \log b(\mathbf{x})$$

- the Gibbs free energy

$$F(b) = U(b) - H(b)$$

A Variational Principle

$$-\log Z = \min_b F(b)$$

Proof: Write

$$\begin{aligned} F(b) &= -\log Z + \sum_{\mathbf{x}} b(\mathbf{x}) \log \frac{b(\mathbf{x})}{p(\mathbf{x})} \\ &= -\log Z + D(b \parallel p) \end{aligned}$$

The Bethe Free Energy

Let $(b_{i,j}(x_i, x_j))$ and $(b_i(x_i))$ be "beliefs" defined for all edges $i \sim j$ and vertices i, respectively. These must satisfy the following:

- the $b_{i,j}$s and b_is are probability mass functions on $\{0,1\}^2$ and $\{0,1\}$, respectively
- $\sum_{x_j} b_{i,j}(x_i, x_j) = b_i(x_i)$ (consistency)

The Bethe Free Energy

Let $(b_{i,j}(x_i, x_j))$ and $(b_i(x_i))$ be "beliefs" defined for all edges $i \sim j$ and vertices i, respectively. These must satisfy the following:

- the $b_{i,j}$s and b_is are probability mass functions on $\{0,1\}^2$ and $\{0,1\}$, respectively
- $\sum_{x_j} b_{i,j}(x_i, x_j) = b_i(x_i)$ (consistency)

We then define

- the Bethe average energy

$$U_{\mathrm{B}}(\{b_{i,j}\}, \{b_i\}) = \sum_{i \sim j} \sum_{(a,b) \in \{0,1\}^2} b_{i,j}(a,b) \log f(a,b)$$

- the Bethe entropy

$$H_{\mathrm{B}}(\{b_{i,j}\}, \{b_i\}) = \sum_{i \sim j} H(b_{i,j}) - \sum_i (d_i - 1) H(b_i)$$

  where d_i denotes the degree of the vertex i.

- the Bethe free energy

$$F_{\mathrm{B}}(\{b_{i,j}\}, \{b_i\}) = U_{\mathrm{B}}(\{b_{i,j}\}, \{b_i\}) - H_{\mathrm{B}}(\{b_{i,j}\}, \{b_i\})$$

The Bethe Approximation

$$-\log Z_{\mathrm{B}} := \min_{\{b_{i,j}\},\{b_i\}} F_{\mathrm{B}}(\{b_{i,j}\},\{b_i\})$$

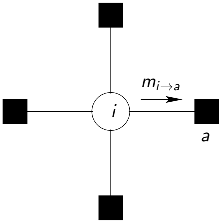
The Bethe Approximation

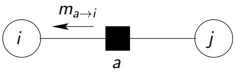
$$-\log Z_{\mathrm{B}} := \min_{\{b_{i,j}\},\{b_i\}} F_{\mathrm{B}}(\{b_{i,j}\},\{b_i\})$$

Theorem (Yedidia, Freeman and Weiss, 2001)

Stationary points of the Bethe free energy functional correspond to the beliefs at fixed points of the belief propagation algorithm.

Belief Propagation (The Sum-Product Algorithm)





Message update rules:

- $m_{i \to a}(x_i) = \prod_{c \in N(i) \setminus a} m_{c \to i}(x_i)$
- $m_{a \to i}(x_i) = \sum_{x_j} f(x_i, x_j) m_{j \to a}(x_j)$

Beliefs:

- $b_i(x_i) \propto \prod_{a \in N(i)} m_{a \to i}(x_i)$
- $b_a(x_i, x_j) \propto f(x_i, x_j) m_{i \to a}(x_i) m_{j \to a}(x_j)$

(the use of $\propto$ indicates that the beliefs must be normalized to sum to 1)

How Good is the Bethe Approximation?

For beliefs $\{b_{i,j}\}$ and $\{b_i\}$ at a fixed point of the BP algorithm, define

$$Z_{\mathrm{BP}}(\{b_{i,j}\},\{b_i\}) := \exp\big(-F_{\mathrm{B}}(\{b_{i,j}\},\{b_i\})\big)$$

Theorem (Wainwright, Jaakkola and Willsky, 2003)

At any fixed point of the BP algorithm,

$$\frac{Z}{Z_{\mathrm{BP}}(\{b_{i,j}\},\{b_i\})} = \sum_{\mathbf{x}} \frac{\prod_{i\sim j} b_{i,j}(x_i,x_j)}{\prod_i b_i(x_i)^{d_i-1}}$$

How Good is the Bethe Approximation?

For beliefs $\{b_{i,j}\}$ and $\{b_i\}$ at a fixed point of the BP algorithm, define

$$Z_{\mathrm{BP}}(\{b_{i,j}\},\{b_i\}) := \exp\big(-F_{\mathrm{B}}(\{b_{i,j}\},\{b_i\})\big)$$

Theorem (Wainwright, Jaakkola and Willsky, 2003)

At any fixed point of the BP algorithm,

$$\frac{Z}{Z_{\mathrm{BP}}(\{b_{i,j}\},\{b_i\})} = \sum_{\mathbf{x}} \frac{\prod_{i\sim j} b_{i,j}(x_i,x_j)}{\prod_i b_i(x_i)^{d_i-1}}$$

Theorem (Chertkov and Chernyak, 2006)

At any fixed point of the BP algorithm,

$$\frac{Z}{Z_{\mathrm{BP}}(\{b_{i,j}\},\{b_i\})} = 1 + \textit{ a finite series of correction terms}$$

How Good is the Bethe Approximation?

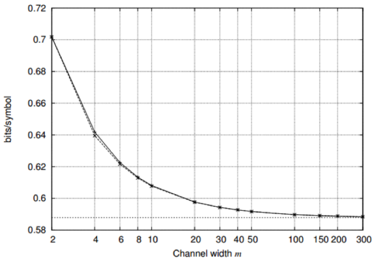
Theorem (Ruozzi, 2012)

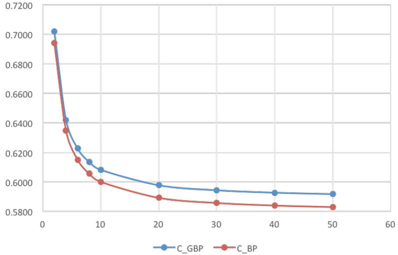
For the binary Ising model considered here,

$$Z \geq Z_{\mathrm{B}}.$$

Numerical Results for the Hard-Square Model

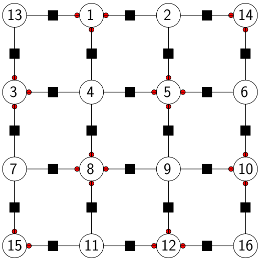
Recall: $\eta = 0.58789116\ldots$.

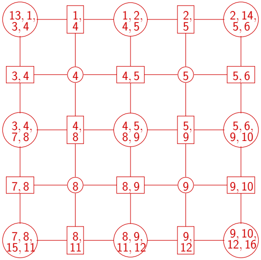




[Plot above is from Sabato and Molkaraie, 2012]

Regions and Region Graphs





$\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1 \cup \mathcal{R}_2$, where

- $\mathcal{R}_0$: all the 2x2 subgrids
- $\mathcal{R}_1$: all the non-boundary edges (intersections of regions in $\mathcal{R}_0$)
- $\mathcal{R}_2$: all the non-boundary vertices (intersections of regions in $\mathcal{R}_1$)

Region-Based Beliefs

Beliefs $b_R(\mathbf{x}_R)$ are defined for all regions $R \in \mathcal{R}$. These must satisfy the following:

- each b_R is a probability mass function on $\{0,1\}^{|R|}$, where $|R|$ denotes the size (number of vertices) of the region R;
- for regions $P \subset R$, we have $\sum_{\mathbf{x}_{R \setminus P}} b_R(\mathbf{x}_R) = b_P(\mathbf{x}_P)$ (consistency)

Region-Based Free Energy

Given a set of beliefs $\{b_R : R \in \mathcal{R}\}$, we define for each region $R \in \mathcal{R}$:

- the average energy of R

$$U_R = \sum_{\mathbf{x}_R \in \{0,1\}^{|R|}} \sum_{i,j \in R : i \sim j} b_R(\mathbf{x}_R) \log f(x_i, x_j)$$

- the entropy of b_R

$$H_R = -\sum_{\mathbf{x}_R} b_R(\mathbf{x}_R) \log b_R(\mathbf{x}_R)$$

- the free energy of R

$$F_R = U_R - H_R$$

Region-Based Free Energy

Given a set of beliefs $\{b_R : R \in \mathcal{R}\}$, we define for each region $R \in \mathcal{R}$:

- the average energy of R

$$U_R = \sum_{\mathbf{x}_R \in \{0,1\}^{|R|}} \sum_{i,j \in R : i \sim j} b_R(\mathbf{x}_R) \log f(x_i, x_j)$$

- the entropy of b_R

$$H_R = -\sum_{\mathbf{x}_R} b_R(\mathbf{x}_R) \log b_R(\mathbf{x}_R)$$

- the free energy of R

$$F_R = U_R - H_R$$

The region-based free energy of the model $(\mathcal{R}, \{b_R\})$ is defined as

$$F_{\mathcal{R}}(\{b_R\}) = \sum_{R \in \mathcal{R}_0} F_R - \sum_{R \in \mathcal{R}_1} F_R + \sum_{R \in \mathcal{R}_2} F_R$$

The Kikuchi Approximation

$$-\log Z_{\mathcal{R}} = \min_{\{b_R\}} F_{\mathcal{R}}(\{b_R\})$$

The Kikuchi Approximation

$$-\log Z_{\mathcal{R}} = \min_{\{b_R\}} F_{\mathcal{R}}(\{b_R\})$$

Theorem (Yedidia, Freeman and Weiss, 2001)

Stationary points of the region-based free energy functional correspond to the beliefs at fixed points of a generalized belief propagation (GBP) algorithm.

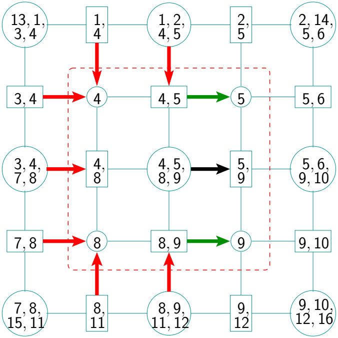
The Kikuchi Approximation

$$-\log Z_{\mathcal{R}} = \min_{\{b_R\}} F_{\mathcal{R}}(\{b_R\})$$

Theorem (Yedidia, Freeman and Weiss, 2001)

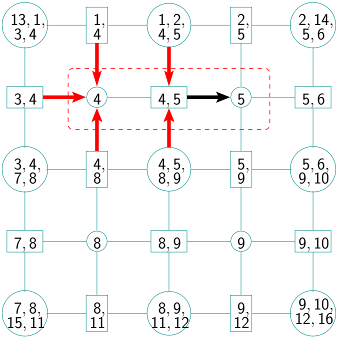
Stationary points of the region-based free energy functional correspond to the beliefs at fixed points of a generalized belief propagation (GBP) algorithm.

GBP Message Updates (The Parent-to-Child Algorithm)



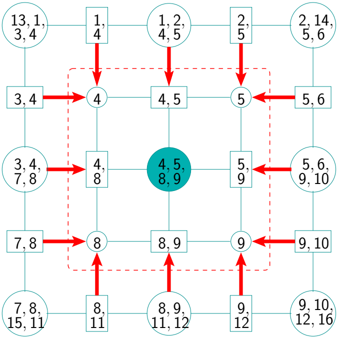
$$m_{(4,5,8,9)\to(5,9)}(x_5, x_9) = \frac{\sum_{x_4, x_8} f(x_4, x_5) f(x_4, x_8) f(x_8, x_9) \prod \text{(red messages)}}{\prod \text{(green messages)}}$$

GBP Message Updates (The Parent-to-Child Algorithm)



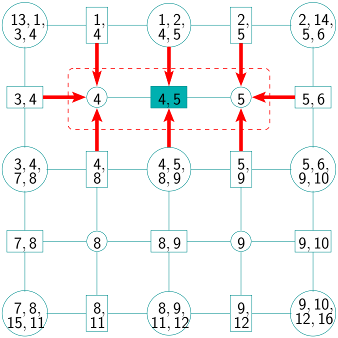
$$m_{(4,5)\to(5)}(x_5) = \sum_{x_4} f(x_4, x_5) \prod \text{(red messages)}$$

GBP Beliefs (The Parent-to-Child Algorithm)



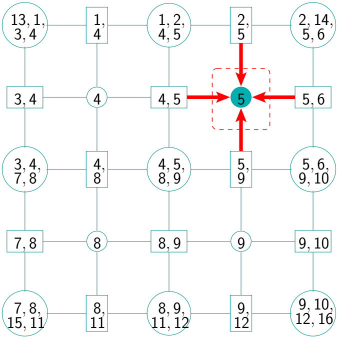
$$b_{(4,5,8,9)}(x_4, x_5, x_8, x_9) \;\; \propto \;\; f(x_4, x_5) f(x_4, x_8) f(x_5, x_9) f(x_8, x_9) \prod \text{(red messages)}$$

GBP Beliefs (The Parent-to-Child Algorithm)



$$b_{(4,5)}(x_4, x_5) \quad \propto \quad f(x_4, x_5) \prod \text{(red messages)}$$

GBP Beliefs (The Parent-to-Child Algorithm)

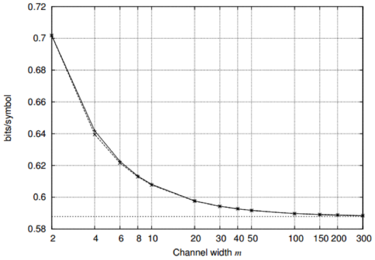


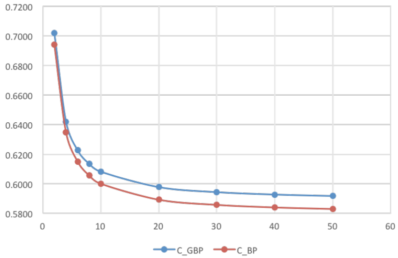
$$b_{(5)}(x_5) \propto \prod \text{(red messages)}$$

How Good is the Kikuchi Approximation?

Looking at the hard-square model again . . .

Recall: $\eta = 0.58789116\ldots$.





[Plot above is from Sabato and Molkaraie, 2012]

What We Conjecture

Conjecture (Chan et al., ISIT'14)

For the binary Ising model considered here,

$$\frac{1}{mn}\log Z - \frac{1}{mn}\log Z_{\mathcal{R}} = o(1)$$

where $o(1)$ *is a positive term that goes to* 0 *as* $m, n \to \infty$*.*

In other words, we conjecture that

$$\frac{Z}{Z_{\mathcal{R}}} = \exp\big(mn\, o(1)\big).$$

Attacking the Conjecture: The Opening Gambit

For beliefs $\{b_R\}$ at a fixed point of the GBP algorithm, define

$$Z_{\mathrm{GBP}}(\{b_R\}) := \exp\big(-F_{\mathcal{R}}(\{b_R\})\big)$$

Theorem (Chan et al., ISIT'14)

At a fixed point of the GBP algorithm,

$$\frac{Z}{Z_{\mathrm{GBP}}(\{b_R\})} = \sum_{\mathbf{x}} \frac{\prod_{R \in \mathcal{R}_0 \cup \mathcal{R}_2} b_R(\mathbf{x}_R)}{\prod_{R \in \mathcal{R}_1} b_R(\mathbf{x}_R)}$$

Attacking the Conjecture: The Opening Gambit

For beliefs $\{b_R\}$ at a fixed point of the GBP algorithm, define

$$Z_{\mathrm{GBP}}(\{b_R\}) := \exp\big(-F_{\mathcal{R}}(\{b_R\})\big)$$

Theorem (Chan et al., ISIT'14)

At a fixed point of the GBP algorithm,

$$\frac{Z}{Z_{\mathrm{GBP}}(\{b_R\})} = \sum_{\mathbf{x}} \frac{\prod_{R \in \mathcal{R}_0 \cup \mathcal{R}_2} b_R(\mathbf{x}_R)}{\prod_{R \in \mathcal{R}_1} b_R(\mathbf{x}_R)}$$

Compare this with

Theorem (Wainwright, Jaakkola and Willsky, 2003)

At any fixed point of the BP algorithm,

$$\frac{Z}{Z_{\mathrm{BP}}(\{b_{i,j}\}, \{b_i\})} = \sum_{\mathbf{x}} \frac{\prod_{i \sim j} b_{i,j}(x_i, x_j)}{\prod_i b_i(x_i)^{d_i - 1}}$$

Question to be Addressed

Question

For the binary Ising model considered here, is it true that the beliefs $\{b_R\}$ *at a fixed point of GBP satisfy*

$$\frac{1}{mn}\log\sum_{\mathbf{x}}\frac{\prod_{R\in\mathcal{R}_0\cup\mathcal{R}_2} b_R(\mathbf{x}_R)}{\prod_{R\in\mathcal{R}_1} b_R(\mathbf{x}_R)} = o(1)$$

where $o(1)$ *is a positive term that goes to* 0 *as* $m, n \to \infty$*?*

What We Can Prove . . .

Theorem (Chan et al., ISIT'14)

For a binary Ising model of size at most 5×5*, or size equal to* $3 \times n$ *(or* $n \times 3$*) we have*

$$\sum_{\mathbf{x}} \frac{\prod_{R \in \mathcal{R}_0 \cup \mathcal{R}_2} b_R(\mathbf{x}_R)}{\prod_{R \in \mathcal{R}_1} b_R(\mathbf{x}_R)} \geq 1$$

at any fixed point of GBP. Consequently,

$$Z \geq Z_{\mathrm{GBP}}(\{b_R\})$$

at any fixed point of GBP.

A Key Tool: Log-Supermodularity

A function $g : \{0,1\}^k \to \mathbb{R}_+$ is called log-supermodular if

$$g(\mathbf{x})g(\mathbf{y}) \leq g(\mathbf{x} \vee \mathbf{y})g(\mathbf{x} \wedge \mathbf{y})$$

for all $\mathbf{x}, \mathbf{y} \in \{0,1\}^k$.

A Key Tool: Log-Supermodularity

A function $g : \{0,1\}^k \to \mathbb{R}_+$ is called log-supermodular if

$$g(\mathbf{x})g(\mathbf{y}) \leq g(\mathbf{x} \vee \mathbf{y})g(\mathbf{x} \wedge \mathbf{y})$$

for all $\mathbf{x}, \mathbf{y} \in \{0,1\}^k$.

Log-supermodularity is preserved under

- multiplication: g_1, g_2 log-supermodular $\implies$ $g_1 g_2$ log-supermodular
- marginalization: g log-supermodular $\implies$ $\sum_{x_1} g(\mathbf{x})$ log-supermod

  (this follows from the Ahlswede-Daykin four functions theorem)

Log-Supermodularity of Functions with Binary Inputs

- A function $f : \{0,1\}^2 \to \mathbb{R}_+$ is log-supermodular iff

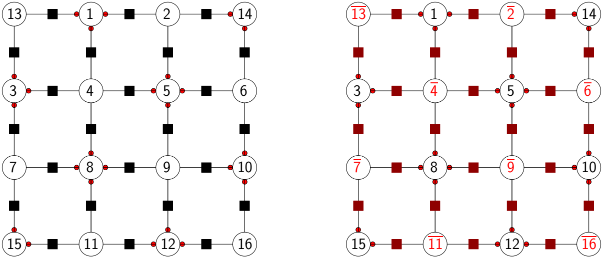
$$f(01)f(10) \leq f(00)f(11)$$

- If $f : \{0,1\}^2 \to \mathbb{R}_+$ is **not** log-supermodular, then the function

$$\bar{f}(a,b) = f(a, 1-b)$$

  is log-supermodular.

A Local Transformation



A binary Ising model defined by local functions f is equivalent to a binary Ising model defined by $\bar{f}$:

- The partition functions are equal: $Z(f) = Z(\bar{f})$
- For each set of beliefs $\{b_R\}$, there exists a corresponding $\{\bar{b}_R\}$ such that

$$\sum_{\mathbf{x}} \frac{\prod_{R \in \mathcal{R}_0 \cup \mathcal{R}_2} b_R(\mathbf{x}_R)}{\prod_{R \in \mathcal{R}_1} b_R(\mathbf{x}_R)} = \sum_{\mathbf{x}} \frac{\prod_{R \in \mathcal{R}_0 \cup \mathcal{R}_2} \bar{b}_R(\mathbf{x}_R)}{\prod_{R \in \mathcal{R}_1} \bar{b}_R(\mathbf{x}_R)}$$

Ising Models with Log-Supermodular Local Functions

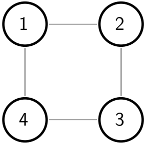
Lemma

In a binary Ising model with log-supermodular local functions, the BP and GBP message update rules preserve log-supermodularity of messages.

Thus, if BP and GBP are initialized with log-supermodular messages, then

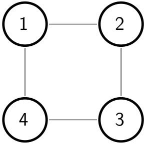
- the messages in subsequent iterations of BP and GBP remain log-supermodular, and
- the BP-based beliefs $b_{i,j}, b_i$ and the GBP-based beliefs b_R are all log-supermodular.

The 2×2 Grid



The Kikuchi approximation is exact: $Z_{\mathcal{R}} = Z$.

The 2×2 Grid



The Kikuchi approximation is exact: $Z_{\mathcal{R}} = Z$.

Theorem

For the 2×2 grid, at any fixed point of BP, we have

$$\frac{Z}{Z_{\mathrm{BP}}(\{b_{i,j}\},\{b_i\})} = 1 + \frac{\Delta_{1,2}\Delta_{2,3}\Delta_{3,4}\Delta_{4,1}}{\prod_i b_i(0)b_i(1)},$$

where $\Delta_{i,j} = b_{i,j}(00)b_{i,j}(11) - b_{i,j}(01)b_{i,j}(10)$.

Proof of 2×2 Theorem

- Start with Wainwright-Jaakkola-Willsky:

$$\frac{Z}{Z_{\mathrm{BP}}(\{b_{i,j}\}, \{b_i\})} = \sum_{x_1,\ldots,x_4} \frac{\prod_{i\sim j} b_{i,j}(x_i, x_j)}{\prod_i b_i(x_i)}$$
$$= \sum_{x_1,\ldots,x_4} \prod_i b_i(x_i) \prod_{i\sim j} \frac{b_{i,j}(x_i, x_j)}{b_i(x_i) b_j(x_j)}$$

- Verify that

$$\frac{b_{i,j}(x_i, x_j)}{b_i(x_i) b_j(x_j)} = 1 + \frac{s(x_i) s(x_j) \Delta_{ij}}{b_i(x_i) b_j(x_j)},$$

where $s(0) = -1$ and $s(1) = +1$.

- Hence,

$$\frac{Z}{Z_{\mathrm{BP}}(\{b_{i,j}\}, \{b_i\})} = \sum_{x_1,\ldots,x_4} \prod_i b_i(x_i) \prod_{i\sim j} \left(1 + \frac{s(x_i) s(x_j) \Delta_{ij}}{b_i(x_i) b_j(x_j)}\right)$$

Proof (cont'd)

- Expand out the product $\prod_{i\sim j}\left(1+\frac{s(x_i)s(x_j)\Delta_{ij}}{b_i(x_i)b_j(x_j)}\right)$:

$$\frac{Z}{Z_{\mathrm{BP}}} = \sum_{x_1,\dots,x_4} \prod_i b_i(x_i)\left(1+\cdots+\frac{\prod_{i\sim j}\Delta_{i,j}}{\prod_i b_i(x_i)^2}\right)$$

- Note that

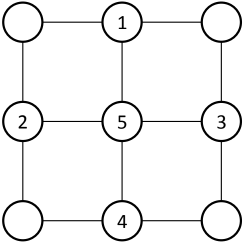
$$\sum_{x_1,\dots,x_4} \prod_i b_i(x_i) \;=\; \prod_i \sum_{x_i} b_i(x_i) \;=\; 1.$$

and

$$\begin{aligned}\sum_{x_1,\dots,x_4} \prod_i b_i(x_i)\frac{\prod_{i\sim j}\Delta_{i,j}}{\prod_i b_i(x_i)^2} &= \prod_{i\sim j}\Delta_{i,j}\sum_{x_1,\dots,x_4}\frac{1}{\prod_i b_i(x_i)}\\ &= \prod_{i\sim j}\Delta_{i,j}\prod_i\sum_{x_i}\frac{1}{b_i(x_i)}\\ &= \prod_{i\sim j}\Delta_{i,j}\prod_i\frac{1}{b_i(0)b_i(1)}\end{aligned}$$

All other terms $\sum_{x_1,\dots,x_4} \prod_i b_i(x_i)(\cdots)$ vanish.

The 3×3 Grid



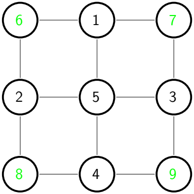
Theorem

For the 3×3 *grid, at any fixed point of GBP, the ratio* $Z/Z_{\mathrm{GBP}}(\{b_R\})$ *is given by*

$$1 + b_5(0)\left(\frac{\Delta(0)}{b_{51}(00)b_{51}(01)}\right)^4 + b_5(1)\left(\frac{\Delta(1)}{b_{51}(10)b_{51}(11)}\right)^4,$$

where $\Delta(x) = b_{512}(x00)b_{512}(x11) - b_{512}(x01)b_{512}(x10)$.

Proof of 3×3 Theorem



- Start with

$$\frac{Z}{Z_{\mathrm{GBP}}(\{b_R\})} = \sum_{x_1,\ldots,x_9} \frac{b(x_{1526})b(x_{1537})b(x_{2548})b(x_{3549})b(x_5)}{b(x_{15})b(x_{25})b(x_{35})b(x_{45})}$$

- Marginalize out x_6, x_7, x_8, x_9:

$$\frac{Z}{Z_{\mathrm{GBP}}(\{b_R\})} = \sum_{x_1,\ldots,x_5} \frac{b(x_{152})b(x_{153})b(x_{254})b(x_{354})b(x_5)}{b(x_{15})b(x_{25})b(x_{35})b(x_{45})}$$

Proof (cont'd)

- Now, define

$$B(\mathbf{x}) = \frac{b(x_{15})b(x_{25})b(x_{35})b(x_{45})}{b(x_5)^3}$$

and write $\frac{Z}{Z_{\mathrm{GBP}}(\{b_R\})}$ as

$$\sum_{x_1,\ldots,x_5} B(\mathbf{x}) \cdot \frac{b(x_{152})b(x_5)}{b(x_{15})b(x_{25})} \cdot \frac{b(x_{153})b(x_5)}{b(x_{15})b(x_{35})} \cdot \frac{b(x_{254})b(x_5)}{b(x_{25})b(x_{45})} \cdot \frac{b(x_{354})b(x_5)}{b(x_{35})b(x_{45})}$$

- Next, verify that

$$\frac{b(x_{i5j})b(x_5)}{b(x_{i5})b(x_{j5})} = 1 + \frac{s(x_i)s(x_j)\Delta_{5ij}(x_5)}{b(x_{i5})b(x_{j5})}$$

where $s(0) = -1$ and $s(1) = +1$.

Plug this back into the expression for $\frac{Z}{Z_{\mathrm{GBP}}(\{b_R\})}$ and simplify.

Proof (cont'd)

- Upon simplification, we obtain

$$\frac{Z}{Z_{\mathrm{GBP}}(\{b_R\})} = 1 + \sum_{x_5} \frac{\Delta(x_5)^4}{b(x_5)^3} \sum_{x_1,\dots,x_4} \prod_{i=1}^{4} \frac{1}{b(x_{i5})}$$

- This further simplifies to

$$\begin{aligned}\frac{Z}{Z_{\mathrm{GBP}}(\{b_R\})} &= 1 + \sum_{x_5} \frac{\Delta(x_5)^4}{b(x_5)^3} \left(\frac{1}{b(x_5 0)} + \frac{1}{b(x_5 1)} \right)^4 \\ &= 1 + \sum_{x_5} b(x_5) \left(\frac{\Delta(x_5)}{b(x_5 0) b(x_5 1)} \right)^4\end{aligned}$$

□

References

[1] Chun Lam Chan, *Estimating the Partition Function of Binary Pairwise Graphical Models Using Generalized Belief Propagation*, M. Phil. Thesis, Dept. Information Engg., Chinese University of Hong Kong, Aug. 2015.

[2] C.L. Chan, M. Jafari Siavoshani, S. Jaggi, N. Kashyap, and P.O. Vontobel, "Generalized Belief Propagation for Estimating the Partition Function of the 2D Ising Model," in *Proc. 2015 IEEE Int. Symp. Inf. Theory (ISIT 2015)*, Hong Kong, China, June 2015.