An FPTAS for #Knapsack and Related Counting Problems

Parikshit Gopalan
Adam Klivans
Raghu Meka
Daniel Štefankovič
Santosh Vempala
Eric Vigoda
An FPTAS for \#Knapsack and Related Counting Problems

Parikshit Gopalan, Adam Klivans, Raghu Meka
Daniel Štefankovič, Santosh Vempala, Eric Vigoda
What can be counted?
(in polynomial-time)

exactly? very little...

number of spanning trees (using determinant), Kirchoff’1847.

perfect matchings in planar graphs (using Pfaffians), Kasteleyn’1960.

(rest: usually #P-hard)
What can be counted?
(in polynomial-time)

approximately? a little more...


Independent sets ($\Delta \leq 5$), Weitz’2004.

$k$-colorings ($k \geq (11/6)\Delta$), Vigoda’1999.

....

(approximate counting $\approx$ random sampling, Jerrum, Valiant, Vazirani’1986)
Approximate counting
(in polynomial-time)

deterministic:

\[
\begin{align*}
\text{INPUT} & \quad \varepsilon \\
& \quad \rightarrow \\
\quad & \quad \rightarrow \text{OUT} \\
\end{align*}
\]

\[
1 - \varepsilon \leq \frac{\text{OUT}}{Q} \leq 1 + \varepsilon
\]

randomized:

\[
\begin{align*}
\text{INPUT} & \quad \varepsilon \\
& \quad \rightarrow \\
\quad & \quad \rightarrow \text{OUT} \\
P\left(1 - \varepsilon \leq \frac{\text{OUT}}{Q} \leq 1 + \varepsilon\right) & \geq 1 - \delta
\end{align*}
\]
not too many examples: independent sets in degree $\leq 5$ graphs (Weitz’2004), matchings in bounded degree graphs (Bayati, Gamarnik, Katz, Nair, Tetali’2007), satisfying assignments of DNF formulas with terms of size $\leq C$ (Ajtai, Wigderson’1985)

more examples; Monte Carlo, usually using a Markov chain (dependence $1/\varepsilon^2$)
1) is randomness necessary?

Is \( P = \text{BPP} \) ?

Primes \( \in P \) (Agarwal, Kayal, Saxena 2001)

2) dependence on \( \varepsilon \) ?

Monte Carlo \( \rightarrow \Theta(1/\varepsilon^2) \)
Knapsack (optimization)

INPUT:
\((w_1, v_1), \ldots (w_n, v_n), L\) (integers)

OUTPUT:
\(S \subseteq [n]\)
\[
\max \sum_{i \in S} v_i \\
\sum_{i \in S} w_i \leq L
\]
Dynamic program #1 (L is small)

$T[i, M]$

(optimal solution with items 1,...,i and limit M)

$T[i, M] = \max \begin{cases} T[i-1, M] \\ T[i-1, M-w_i] + v_i \end{cases}$
Dynamic program #2 (\(v_i\)’s are small)

\[
T[i, V] = \min \left\{ T[i-1, V], T[i-1, V-v_i] + w_i \right\}
\]

(smallest weight of a subset of 1,...,i, with value \(\geq V\))

Knapsack (optimization)

INPUT:
\((w_1, v_1), \ldots, (w_n, v_n), L\) (integers)

OUTPUT:
\[
\max_{S \subseteq [n]} \sum_{i \in S} v_i
\]
\[
\sum_{i \in S} w_i \leq L
\]
Counting knapsack

INPUT:

\[ w_1, \ldots, w_n, \ L \]

OUTPUT:

How many \( S \subseteq [n] \)

with

\[ \sum_{i \in S} w_i \leq L \]

are there?

#P-hard
Counting knapsack

Dyer, Frieze, Kannan, Kapoor, Perkovic, Vazirani’1993
exp(O*(n^{1/2})) / \varepsilon^2
randomized approximation algorithm

Morris, Sinclair’1999
O(n^c / \varepsilon^2)
randomized approximation algorithm
(MCMC, canonical paths)

Dyer’2003
O(n^{2.5} + n^2/\varepsilon^2)
randomized approximation algorithm
(dynamic programming)

OURS: O^*(n^3/\varepsilon)
Dyer’2003:  

\[ T[i,M] \]

(number of solutions with items 1,...,i and limit M)

\[ T[i,M] = T[i-1,M] + T[i-1,M-w_i] \]

+ rejection sampling → approximate counter

Counting knapsack

INPUT:

\( w_1, \ldots, w_n, L \)

OUTPUT:

How many \( S \subseteq [n] \)

with \( \sum_{i \in S} w_i \leq L \)

are there?

#P-hard
+ rejection sampling

approximate counter

\[ w_i' = \frac{n^2 w_i}{L} \quad L' = n^2 \]

rounding:

\[ w_i'' = \left\lfloor w_i' \right\rfloor \]

1) get more solutions, \( \Omega'' \supseteq \Omega' \)

2) not too many more, \( |\Omega''| \leq (n+1)|\Omega'| \)

Proof:

\( S'' \in \Omega'' - \Omega' \), \( X \) heaviest in \( S'' \), then \( S'' - \{X\} \in \Omega' \)

Dyer’2003:

\[ T[i,M] \]

(number of solutions with items 1,...,i and limit M)

\[ T[i,M] = T[i-1,M] + T[i-1,M-w_i] \]

+ rejection sampling  

approximate counter

### Counting knapsack

INPUT:

\( w_1, \ldots, w_n, L \)

OUTPUT:

How many \( \sum_{i \in S} w_i \leq L \) are there?

\#P-hard
Our dynamic program

deterministic approximation algorithm

\[ \tau(i,A) = \text{smallest } M \text{ such that knapsack with } w_1, \ldots, w_i, M \text{ has } \geq A \text{ solutions} \]

\[ \tau(i,A) = \min_{\alpha \in [0,1]} \max \{ \tau(i-1, \alpha A), \tau(i-1, (1-\alpha) A) + w_i \} \]
Q = 1 + \varepsilon/(n+1) \\
\text{s} = \left\lfloor n \log_Q 2 \right\rfloor \\
T[0..n,0..s] \\
T(i,j) = \min_{\alpha \in [0,1]} \max \left\{ T(i-1, j + \ln_Q \alpha) , T(i-1, j + \ln_Q (1-\alpha)) + w_i \right\} \\
\text{Lemma 1:} \quad \tau(i,Q^{j-i}) \leq T[i,j] \leq \tau(i,Q^j)
\[
T(i,j) = \min_{\alpha \in [0,1]} \max \left\{ T(i-1, j + \ln_Q \alpha), T(i-1, j + \ln_Q (1 - \alpha)) + w_i \right\}
\]

Lemma 2: can compute recursion efficiently

only few values of \( \alpha \) matter

\( Q^{-j}, \ldots, Q^0, 1-Q^0, \ldots, 1-Q^j \)

can use binary search

TOTAL RUN TIME = \( O\left( \frac{n^3}{\varepsilon} \log(n/\varepsilon) \right) \)
How to deal with more constraints?
(e.g., contingency tables, multi-dimensional knapsack, ...)

multi-dimensional knapsack:
How many $S \subseteq [n]$ with
\[ \sum_{i \in S} w_{j,i} \leq L_j \]
are there?

$O\left( \frac{n}{\varepsilon} O(k^2) \log W \right)$ algorithm
Read once branching programs

\((S, n)\text{-}ROBP\)

- Layered directed graph
- \(S\) vertices per layer
- Edges between consecutive layers
- Edges labeled \(\{0, 1\}\)
- Input: \((x_1, \ldots, x_n)\)
- Output: Label of final vertex reached

Counting the number of accepting paths?
Dynamic programming, time = \(O(nS)\)
ROBP for knapsack

Problem: width too large
Solution: reduce width by approximating
Monotone ROBPs

monotone: \( u \leq v \iff A(u) \subseteq A(v) \)

given implicitly

- ordering: given \( u,v \), is \( u \leq v \)?

- midpoint: given \( u,v \), get \( w \) s.t.
  \[ |\{x;u \leq x \leq w\}| = |\{x;w \leq x \leq v\}| \pm 1 \]

- transitions: given \( u \), get the outneighbors of \( u \)
group the vertices in the layers according to the rough number of accepting paths processing right-left

Problem: width too large
Solution: reduce width by approximating

Monotone ROBPs

monotone: \( u \leq v \iff A(u) \subseteq A(v) \)

given implicitly

- ordering: given \( u,v \), is \( u \leq v \) ?
- midpoint: given \( u,v \), get \( w \) s.t. \( |\{x; u \leq x \leq w\}| = |\{x; w \leq x \leq v\}| \pm 1 \)
- transitions: given \( u \), get the outneighbors of \( u \)
More constraints?

can be generalized to distributions given by small space sources.

small space sources = ROBP + probability distributions on outgoing edges
More constraints?

\((S,n)\)-ROBP

\[
\begin{array}{c}
\vdots \\
S \\
\vdots \\
\{0,1\} \\
\vdots \\
n \text{ layers} \\
\vdots \\
\end{array}
\]

\((S,n)\)-ROBP

\[
\begin{array}{c}
\vdots \\
S \\
\vdots \\
\{0,1\} \\
\vdots \\
n \text{ layers} \\
\vdots \\
\end{array}
\]

can be combined to get \((S^2,n)\)-ROBP for intersection

additive approximation preserved
1) uniform distribution given by $\Omega''$ can be given by small space source

2) additive approximation $\Rightarrow$ multiplicative approximation
More constraints? can be generalized to distributions given by **small space sources**.

**small space sources = ROBP + probability distributions on outgoing edges**

How to deal with more constraints?
(e.g., contingency tables, multi-dimensional knapsack, ...)

**multi-dimensional knapsack:**
How many \( S \subseteq [n] \)
with \( \sum_{i \in S} w_{j,i} \leq L_j \)
are there?

\( O(\frac{n}{\varepsilon})^{O(k^2)} \log W) \) algorithm

More constraints?

\( (s, n) \)-ROBP

can be combined to get \((S^2, n)\)-ROBP for intersection

**additive approximation preserved**

1) uniform distribution given by \( \Omega'' \) can be given by small space source

2) additive approximation \( \Rightarrow \) multiplicative approximation
Other problems:
contingency tables with constant number of rows

What other problems are solvable using the technique?

Thanks!
Spin models: connections between complexity, phase transition, belief propagation, and induced matrix norms

Andreas Galanis
Daniel Štefankovič
Eric Vigoda
Which problems are easy/hard?

**EASY**
- polynomial-time algorithm
- find the largest matching in a graph $G$
- find min-cut of a graph $G$
- find a 4-coloring of a planar graph $G$
- find a 2-coloring of a graph $G$
- find 3-coloring of a graph $G$ with max-degree $\leq 3$

**HARD**
- NP-complete
- find the largest independent set in a graph $G$
- find max-cut of a graph $G$
- find 3-coloring of a planar graph $G$
- find 3-coloring of a graph $G$ with max-degree $\leq 4$
**Which counting problems are easy/hard?**

**exact counting**

<table>
<thead>
<tr>
<th>EASY</th>
<th>HARD</th>
</tr>
</thead>
<tbody>
<tr>
<td>polynomial-time algorithm</td>
<td>#P-complete</td>
</tr>
<tr>
<td>count the number of spanning trees of a graph G</td>
<td>count the number of spanning forests of a graph G</td>
</tr>
<tr>
<td>count the number of perfect matchings of a planar graph G</td>
<td>count the number of perfect matchings of a bipartite graph G</td>
</tr>
<tr>
<td>count the number of euler tours in an directed graph</td>
<td>count the number of euler tours in an undirected graph</td>
</tr>
</tbody>
</table>
Which **counting** problems are easy/hard?

**approximate counting**

**EASY**
- polynomial-time approximation algorithm
- count the number of spanning trees of a graph $G$
- count the number of perfect matchings of a planar graph $G$
- count the number of independent sets of a graph with maximum degree $\leq 5$

**HARD**
- no approximation, unless RP=NP
- count the number of perfect matchings of a bipartite graph $G$
- count the number of independent sets of a graph with maximum degree $\leq 6$
Which **counting** problems are easy/hard?

<table>
<thead>
<tr>
<th>EASY</th>
<th>HARD</th>
</tr>
</thead>
<tbody>
<tr>
<td>polynomial-time approximation algorithm</td>
<td>no approximation, unless RP=NP</td>
</tr>
<tr>
<td>count the number of spanning trees of a graph ( G )</td>
<td>count the number of independent sets of a graph with maximum degree ( \leq 5 )</td>
</tr>
<tr>
<td>count the number of perfect matchings of a planar graph ( G )</td>
<td>count the number of independent sets of a graph with maximum degree ( \leq 6 )</td>
</tr>
<tr>
<td>count the number of perfect matchings of a bipartite graph ( G )</td>
<td></td>
</tr>
</tbody>
</table>

**approximate counting**
Spin models - examples
Ising model:

valid configurations = 2-labelings of vertices
weight of a configuration C is $B^{\text{number of monochromatic edges}}$

$B > 1$ ferromagnetic
$B < 1$ anti-ferromagnetic

Partition function:

$$Z(B) = 2B^2 + 4B + 2$$
Ising model:
valid configurations = 2-labelings of vertices
weight of a configuration C is $B^{\text{number of monochromatic edges}}$

$B > 1$ ferromagnetic
$B < 1$ anti-ferromagnetic

Partition function:

$$Z(B) = 2B^2 + 4B + 2$$
Potts model:

valid configurations = q-labelings of vertices
weight of a configuration C is $B^{\text{number of monochromatic edges}}$

$B > 1$ ferromagnetic
$B < 1$ anti-ferromagnetic

Partition function:

$$Z(B) = 3(B^2 + 4B + 4)$$
q-colorings:
valid configurations = q-colorings of vertices

Partition function:
Z = 12
Hard-core model:
valid configurations = independent sets
weight of a configuration $I$ is $\lambda^{|I|}$

Partition function:
$$Z(\lambda) = 1 + 4\lambda + 2\lambda^2$$
Hard-core model:
valid configurations = independent sets
weight of a configuration $i$ is $\lambda^{|i|}$

Partition function:
$Z(\lambda) = 1 + 4\lambda + 2\lambda^2$

$\lambda \rightarrow \infty$ ?

Are these easy/hard to approximate?

Ising model:
valid configurations = 2-labelings of vertices
weight of a configuration $C = B\text{number of monochromatic edges}$

$B > 1 =$ ferromagnetic
$B < 1 =$ anti-ferromagnetic

Partition function:
$Z(B) = 2B^2 + 4B + 2$

$B \rightarrow 0$ ?

Which problems are easy/hard?

**EASY**
- polynomial-time algorithm
- find the largest matching in a graph $G$
- find min-cut of a graph $G$
- find a 4-coloring of a planar graph $G$
- find a 2-coloring of a graph $G$
- find 3-coloring of a graph $G$ with max-degree $\leq 3$

**HARD**
- NP-complete
- find the largest independent set in a graph $G$
- find max-cut of a graph $G$
- find 3-coloring of a planar graph $G$
- find 3-coloring of a graph $G$ with max-degree $\leq 4$
Spin models – general setting
General spin model

$q = \text{number of spins}$

$B = q \times q \text{ interaction matrix}$

$G = (V, E) \text{ an undirected graph}$

Configuration $\sigma: V \rightarrow \{1,2,\ldots,q\}$

$P(\sigma) = \prod_{\{u,v\} \in E} B_{\sigma(u), \sigma(v)}$

$(\text{sometimes also an external field})$
### Ising model:
- valid configurations = 2-labelings of vertices
- weight of a configuration $C$ is $B^n$ number of monochromatic edges
- $B > 1 = $ ferromagnetic
- $B < 1 = $ anti-ferromagnetic

Partition function:
$$Z(B) = 2B^2 + 4B + 2$$

### General spin model
- $q = $ number of spins
- $B = q \times q$ interaction matrix
- $G=(V,E)$ an undirected graph

Configuration $\sigma$: $V \rightarrow \{1, 2, \ldots, q\}$

Partition function:
$$P(\sigma) = \prod_{\{u, v\} \in E} B_{\sigma(u), \sigma(v)}$$

(sometimes also an external field)
Hard-core model:
valid configurations = independent sets
weight of a configuration \( l \) is \( \lambda^l \)

Partition function:
\[
Z(\lambda) = 1 + 4\lambda + 2\lambda^2
\]
forsmall-D-regular graphs

General spin model

- \( q = \) number of spins
- \( B = q \times q \) interaction matrix
- \( G = (V,E) \) an undirected graph

Configuration \( \sigma: V \to \{1,2,\ldots,q\} \)

\[
P(\sigma) = \frac{\prod_{\{u,v\} \in E} B_{\sigma(u),\sigma(v)}}{Z}
\]
(sometimes also an external field)
General spin model

- $q = \text{number of spins}$
- $B = q \times q$ interaction matrix
- $G = (V, E)$ an undirected graph

Configuration $\sigma: V \to \{1, 2, \ldots, q\}$

$$P(\sigma) = \frac{\prod_{\{u, v\} \in E} B_{\sigma(u), \sigma(v)}}{Z}$$

(sometimes also an external field)

$$\begin{pmatrix} B & 1 & 1 \\ 1 & B & 1 \\ 1 & 1 & B \end{pmatrix}$$

Potts model (arbitrary $q$)

H-colorings

$B = \text{adjacency matrix of a graph}$

$$\begin{pmatrix} 1 & \lambda^{1/\Delta} \\ \lambda^{1/\Delta} & 0 \end{pmatrix}$$

hard-core model ($q=2$)

$$\begin{pmatrix} B & 1 \\ 1 & B \end{pmatrix}$$

Ising model ($q=2$)

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Widom-Rowlinson model ($q=3$)
For which spin models is there an FPRAS for computing the partition function on graphs with max-degree $\leq \Delta$?

FPRAS (fully-polynomial randomized approximation scheme): given $\varepsilon, G$ in time poly($|G|, 1/\varepsilon$) outputs an answer in $[ (1 - \varepsilon) Z, (1 + \varepsilon) Z ]$ with probability $\geq 3/4$. 

General spin model

$q = \text{number of spins}$

$B = q \times q \text{ interaction matrix}$

$G = (V, E)$ an undirected graph

Configuration $\sigma: V \rightarrow \{1, 2, \ldots, q\}$

$$P(\sigma) = \frac{\prod_{\{u,v\} \in E} B_{\sigma(u), \sigma(v)}}{Z}$$

(sometimes also an external field)
Tree recursions, part 1
Long range influence (aka non-uniqueness)?

What is the probability that the root is \( \bullet \)?

\[
q = 2
\]

\[
\begin{pmatrix}
B & 1 \\
1 & B
\end{pmatrix}
\]
Long range influence (aka non-uniqueness)?

What is the probability that the root is \( \bullet \)?

\[
q = 2
\]

\[
\begin{pmatrix}
B & 1 \\
1 & B
\end{pmatrix}
\]

\( B^{14} \)

\( B^7 \)

\( B^{12} \)

\( B^6 \)
Long range influence (aka non-uniqueness)?

Sample on d-ary tree of height $h$, conditioned on leaves being red

Sample on d-ary tree of height $h$, conditioned on leaves being blue

Distribution of root color as $h \to \infty$

Same on left and right = no long range influence

Different = long range influence

(in general spin systems all leaf labelings need to be considered)
Long range influence (aka non-uniqueness)?

What is the probability that the root is $q = \frac{2}{g_{1828}}$?

What is the probability that the root is red?
Message passing

\[ m(\bigcirc) = B \ m_1(\bigcirc)m_2(\bigcirc)m_3(\bigcirc) + m_1(\bigcirc)m_2(\bigcirc)m_3(\bigcirc) \]

\[ m(\bigotimes) = m_1(\bigotimes)m_2(\bigotimes)m_3(\bigotimes) + B \ m_1(\bigotimes)m_2(\bigotimes)m_3(\bigotimes) \]
Message passing

\[ m(\bullet) = B \ m_1(\bullet)m_2(\bullet) + m_1(\circ) m_2(\circ) \]

\[ m(\circ) = m_1(\bullet)m_2(\bullet) + B \ m_1(\circ)m_2(\circ) \]

\[ m_1(\bullet) = 1 + B^3 \]
Message passing

\[ m(\bullet) = B \ m_1(\bullet)m_2(\bullet)m_3(\bullet) + m_1(\bullet)m_2(\bullet)m_3(\bullet) \]

\[ m(\circ) = m_1(\bullet)m_2(\bullet)m_3(\bullet) + B \ m_1(\bullet)m_2(\bullet)m_3(\bullet) \]

\[ m_1(\bullet) = B + B^2 \]
Message passing

\[ m(\bullet) = B \ m_1(\bullet)m_2(\bullet)m_3(\bullet) + m_1(\bullet)m_2(\bullet)m_3(\bullet) \]

\[ m(\bullet) = m_1(\bullet)m_2(\bullet)m_3(\bullet) + B \ m_1(\bullet)m_2(\bullet)m_3(\bullet) \]

\[ m_1(\bullet) = B + B^2 \]

\[ m_2(\bullet) = B + B^2 \]

\[ m_1(\bullet) = 1 + B^3 \]

\[ m_2(\bullet) = 1 + B^3 \]

\[ m(\bullet) = B(1+B^3)^2 + (B+B^2)^2 \]
Long range influence (aka non-uniqueness)?

Sample on d-ary tree of height $h$, conditioned on leaves being red

Sample on d-ary tree of height $h$, conditioned on leaves being blue

Distribution of root color as $h \to \infty$

Same on left and right $= \text{no long range influence}$

Different $= \text{long range influence}$

(in general spin systems all leaf labelings need to be considered)
Period 2?

\[ r = \frac{B \ c^d + 1}{c^d + B} \]

\[ c = \frac{B \ r^d + 1}{r^d + B} \]

uniqueness = one solution, \( r = c \)

non-uniqueness = multiple solutions
(long-range influence)

\[ r = \frac{m_{\text{even}}(\bullet)}{m_{\text{even}}(\bigcirc)} \quad c = \frac{m_{\text{odd}}(\bullet)}{m_{\text{odd}}(\bigcirc)} \]
Uniqueness thresholds
d=Δ-1

Hard-core model:
valid configurations = independent sets
weight of a configuration I is \( \lambda^{|I|} \)

Partition function:
\[ Z(\lambda) = 1 + 4\lambda + 2\lambda^2 \]

Ising model:
valid configurations = 2-labelings of vertices
weight of a configuration C = \( B \) number of monochromatic edges

- \( B > 1 \) = ferromagnetic
- \( B < 1 \) = anti-ferromagnetic

Partition function:
\[ Z(B) = 2B^2 + 4B + 2 \]

\[ \lambda_c(\Delta) = \frac{(\Delta-1)^{\Delta-1}}{(\Delta-2)^\Delta} \]

\[ B_c(\Delta) \]
For which spin models is there an FPRAS for computing the partition function on graphs with max-degree $\leq \Delta$?

FPRAS (fully-polynomial randomized approximation scheme):
given $\varepsilon, G$ in time $\text{poly}(|G|, 1/\varepsilon)$ outputs an answer in $[ (1 - \varepsilon) Z, (1 + \varepsilon) Z ]$ with probability $\geq 3/4$. 

General spin model

$q = \text{number of spins}$

$B = q \times q$ interaction matrix

$G = (V, E) \text{ an undirected graph}$

Configuration $\sigma: V \rightarrow \{1, 2, \ldots, q\}$

$$P(\sigma) = \frac{\prod_{\{u, v\} \in E} B_{\sigma(u), \sigma(v)}}{Z}$$

(sometimes also an external field)
For which spin models is there an FPRAS for computing the partition function on graphs with max-degree $\leq \Delta$?

FPRAS (fully-polynomial randomized approximation scheme): given $\epsilon, G$ in time poly($|G|, 1/\epsilon$) outputs an answer in $[ (1 - \epsilon) Z, (1 + \epsilon) Z ]$ with probability $\geq 3/4$. 

Previous results
polynomial-time approximation algorithm (for constant $\Delta$)

(Weitz’06)

hard to approximate (unless NP=RP)

(Sly’10)

HARD-CORE MODEL:

polynomial-time approximation algorithm (for constant $\Delta$)

(Weitz’06)

HARD-CORE MODEL:
<table>
<thead>
<tr>
<th>Easy</th>
<th>Hard</th>
</tr>
</thead>
<tbody>
<tr>
<td>count the number of spanning trees of a graph ( G )</td>
<td>no approximation, unless BPP=NP</td>
</tr>
<tr>
<td>count the number of perfect matchings of a planar graph ( G )</td>
<td></td>
</tr>
<tr>
<td>count the number of perfect matchings of a bipartite graph ( G )</td>
<td></td>
</tr>
<tr>
<td>count the number of independent sets of a graph with maximum degree ( \leq 5 )</td>
<td>count the number of independent sets of a graph with maximum degree ( \leq 6 )</td>
</tr>
</tbody>
</table>

**HARD-CORE MODEL:**

- polynomial-time approximation algorithm (for constant \( \Delta \)) *(Weitz’06)*
- hard to approximate (unless NP=RP) *(Sly’10)*

Galanis-Ge-Stefankovic-Vigoda-Yang’11 (all \( \Delta \neq 4,5 \)),
Galanis-Stefankovic-Vigoda’12 (\( \Delta=4,5 \)), Sly-Sun’12 (all \( \Delta \))
Other models?

ANTIFERROMAGNETIC ISING MODEL:

- polynomial-time approximation algorithm (for constant $\Delta$) 
  (Weitz’06, Sinclair-Srivastava-Thurley’11, Li-Lu-Yin’11)
- hard to approximate (unless NP=RP) 
  (Sly’10, Sly-Sun’12, Galanis-Stefankovic-Vigoda’12)
Spin models with two spins (q=2)

**Antiferromagnetic (det B < 0)**
long-range influence ⇒ hard to approximate
(unless NP=RP)
Uniqueness ⇒ poly-time approx. algorithm

**Ferromagnetic models (det B > 0)**
⇒ poly-time algorithm

(Goldberg-Jerrum-Patterson’03 using Jerrum-Sinclair’93)

(no external field)
Spin models with two spins (q=2)

**Antiferromagnetic (det B < 0)**
long-range influence ⇒ hard to approximate
(unless NP=RP)
Uniqueness ⇒ poly-time approx. algorithm

**Ferromagnetic models (det B > 0)**
⇒ poly-time algorithm
(Goldberg-Jerrum-Patterson’03 using Jerrum-Sinclair’93)

(no external field)

Explain the connection between non-uniqueness and hardness

Extend the connection to q>2
CS: which counting problems are easy/hard? Interpreting EASY/HARD

- polynomial-time approximation algorithm
- no approximation unless $\text{BPP}=\text{NP}$

- count the number of spanning trees of a graph $G$
- count the number of perfect matchings of a planar graph $G$
- count the number of perfect matchings of a bipartite graph $G$
- count the number of independent sets of a graph with maximum degree $\leq 5$

- count the number of independent sets of a graph with maximum degree $\leq 6$

**HARD-CORE MODEL:**

- polynomial-time approximation algorithm (for constant $\Delta$)

(Weitz’06) 

Sly’10

**Uniqueness thresholds**

- $d=\Delta-1$

- $\lambda_c(\Delta) = \frac{(\Delta-1)^{\Delta-1}}{(\Delta-2)^\Delta}$

- $B_c(\Delta)$

- long-range influence
Behavior of models on random $\Delta$-regular bipartite graphs
(Mossel, Weitz, Wormald’09)

Why? Gadget in Sly’s reduction:

max-cut?
1) sample a random $\Delta$-regular (bipartite) graph $G$ (union of $\Delta$ random matchings)

2) sample a random configuration from the model

How does the result “look” as $n \to \infty$?
1) sample a random $\Delta$-regular (bipartite) graph $G$
(union of $\Delta$ random matchings)

2) sample a random configuration from the model

How does the result “look” as $n \to \infty$?
1) sample a random $\Delta$-regular (bipartite) graph $G$ (union of $\Delta$ random matchings)

2) sample a random configuration from the model

How does the result “look” as $n \to \infty$?

$$P(\sigma) = \frac{\prod_{\{u,v\} \in E} B_{\sigma(u), \sigma(v)}}{Z}$$
1) sample a random $\Delta$-regular (bipartite) graph $G$ (union of $\Delta$ random matchings)

2) sample a random configuration from the model

How does the result “look” as $n \to \infty$?

\[
P(\sigma) = \frac{\prod_{\{u,v\} \in E} B_{\sigma(u),\sigma(v)}}{Z}
\]
1) sample a random $\Delta$-regular (bipartite) graph $G$ (union of $\Delta$ random matchings)

2) sample a random configuration from the model

How does the result “look” as $n \to \infty$?

\[
P(\sigma) = \frac{\prod_{\{u,v\} \in E} B_{\sigma(u),\sigma(v)}}{Z}
\]

$\alpha = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}$  $\beta = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}$

signature of the configuration
1) sample a random $\Delta$-regular (bipartite) graph $G$
(union of $\Delta$ random matchings)

2) sample a random configuration from the model

How does the result “look” as $n \to \infty$?

\[
P(\alpha, \beta) = \frac{Z_{\alpha,\beta}}{Z}
\]

\[
Z_{\alpha,\beta} = \sum \prod_{\{u,v\} \in E} B_{\sigma(u),\sigma(v)}
\]

$\alpha = \frac{1}{2} \quad \frac{1}{2} \quad \beta = \frac{1}{2} \quad \frac{1}{2}$
1) sample a random $\Delta$-regular (bipartite) graph $G$
(union of $\Delta$ random matchings)

2) sample a random configuration from the model

How does the result “look” as $n \to \infty$?

\[ \mathbb{P}(\alpha, \beta) = \frac{Z_{\alpha, \beta}}{Z} \]

Goal:
1) understand $E[Z_{\alpha, \beta}]$
2) understand $E[Z_{\alpha, \beta}^2]$
1) understand $E[Z_{\alpha, \beta}]$

$$
\Phi = (\Delta - 1) \left( \sum_{i=1}^{q} \alpha_i \ln \alpha_i + \sum_{j=1}^{q} \beta_j \ln \beta_j \right) + \Delta g(\alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_q),
$$

where the maximum is taken over $x_{ij}$'s such that

$$
\alpha_i = \sum_{j} x_{ij} \quad \text{and} \quad \beta_j = \sum_{i} x_{ij}.
$$

[Morel-Weitz-Wormald’09] for hardcore model

$E[Z_{\alpha, \beta}]$ for $\lambda \leq \lambda_c$

$E[Z_{\alpha, \beta}]$ for $\lambda > \lambda_c$
Our results
1) understand $E[Z_{\alpha,\beta}]$

$$\Phi(\alpha, \beta) = \log E[Z_{\alpha,\beta}]$$

**Theorem I:**

The local maxima of $\Phi(\alpha, \beta)$ are in 1-1 correspondence with the local maxima of

$$\frac{||Bx||_\Delta}{||x||_{\Delta/(\Delta-1)}}$$

useful: allows us to do second moment
1) understand $E[Z_{\alpha,\beta}]$

$$\Phi(\alpha, \beta) = \log E[Z_{\alpha,\beta}]$$

**Theorem II:**

The local maxima of $\Phi(\alpha, \beta)$ are in 1-1 correspondence with the attractive fixpoints of the tree recursions.

\[
m(\bullet) = B m_1(\bullet)m_2(\bullet)m_3(\bullet) + m_1(\bullet)m_2(\bullet)m_3(\bullet)
\]

\[
m(\circ) = m_1(\bullet)m_2(\bullet)m_3(\bullet) + B m_1(\bullet)m_2(\bullet)m_3(\bullet)
\]

useful: allows us to do understand specific models
1) understand $E[Z_{\alpha,\beta}]$

$$\Phi(\alpha, \beta) = \log E[Z_{\alpha,\beta}]$$

**Theorem II:**

The **local maxima** of $\Phi(\alpha, \beta)$ are in 1-1 correspondence with the **attractive fixpoints** of the tree recursions.

$$m(\bullet) = B \ m_1(\bullet)m_2(\bullet)m_3(\bullet) + m_1(\bullet)m_2(\bullet)m_3(\bullet)$$

$$m(\circ) = m_1(\bullet)m_2(\bullet)m_3(\bullet) + B \ m_1(\bullet)m_2(\bullet)m_3(\bullet)$$

useful: allows us to do understand specific models
1) understand $E[Z_{\alpha, \beta}]$

$$\Phi(\alpha, \beta) = \log E[Z_{\alpha, \beta}]$$

Theorem I:
The local maxima of $\Phi(\alpha, \beta)$ are in 1-1 correspondence with the local maxima of

$$\frac{||Bx||_\Delta}{||x||_{\Delta/(\Delta-1)}}$$

useful: allows us to do second moment

Matrix norms
Induced matrix norms

$$||A||_{p \rightarrow q}, = \max \ ||Ax||_{q},$$

$$||x||_{p} = 1$$

For example:

spectral norm:  $$||A||_{2 \rightarrow 2} = \sqrt{\lambda_{\max}(A A^{T})}$$
Induced matrix norms

\[ \|A\|_{p\rightarrow q}, = \max \|Ax\|_{q}, \quad \|x\|_{p} = 1 \]

very unwieldy, even for concrete simple inputs:

\[ \left\| \begin{pmatrix} B & 1 & 1 \\ 1 & B & 1 \\ 1 & 1 & B \end{pmatrix} \right\|_{p \rightarrow q} = ? \]
Who cares about induced matrix norms?

\[ \| A \|_{p \rightarrow q'} = \max \| A x \|_{q'}, \quad \| x \|_p = 1 \]

E.g., Bonami-Beckner inequality

\[ 1 \leq p \leq q' \quad \text{and} \quad 0 \leq \rho \leq \sqrt{\frac{p - 1}{q' - 1}} \]

\[ \begin{bmatrix} \frac{1}{2} + \rho & \frac{1}{2} - \rho \\ \frac{1}{2} - \rho & \frac{1}{2} + \rho \end{bmatrix} \leq 1 \quad p \rightarrow q' \]
Computing induced matrix norms?

\[ \| A \|_{p \rightarrow q}, = \max \|Ax\|_q, \quad \|x\|_p = 1 \]

The Power Method for \( \ell^p \) Norms

David W. Boyd*

only finds local maxima (for \( p < q' \))

applies tree recursions (in disguise)
This paper presents a generalization of the classical power method for computing matrix norms of the form

$$\|A\|_{r,p} = \max\{\|Ax\|_p : \|x\|_r = 1\},$$

where $1 < r, p < \infty$ and $\|x\|_p$ is the usual $p$ vector norm. The method converges for any matrix $A$, but the limit may be only a local maximum, or even just a critical point, of the functional $Fx = \|Ax\|_p/\|x\|_r$. If $A$ is a nonnegative matrix, it is shown that the limit is indeed the desired norm. Also, in certain cases it is shown that the sequence of vectors that is produced by the method converges to the maximizing point. The claim that the method "rapidly computes" the norm is substantiated only by applying it to a number of examples. The convergence proof uses a monotonicity argument which gives no information about how fast the iteration converges.

Reviewed by J. Vandergraft
Tensorisation of induced matrix norms

\[ \| A \|_{p \to q'}, = \max \| A x \|_{q'}, \]
\[ \| x \|_p = 1 \]

A useful fact (Bennett'77):

For \( p < q' \)

\[ \| A \otimes B \|_{p \to q'} = \| A \|_{p \to q'} \| B \|_{p \to q'} \]

\[
\begin{bmatrix}
a_{11}B & \cdots & a_{1n}B \\
\vdots & \ddots & \vdots \\
a_{n1}B & \cdots & a_{nn}B
\end{bmatrix}
\]
Theorem I:

The local maxima of $\Phi(\alpha, \beta)$ are in 1-1 correspondence with the local maxima of

$$\frac{||Bx||_\Delta}{||x||_{\Delta/(\Delta-1)}}$$

To understand behavior of a spin model with matrix B on random $\Delta$-regular bipartite random graph just compute $||B||_{\Delta/(\Delta-1) \to \Delta}$
Tree recursions, part 2

1) understand \( E[Z_{\alpha,\beta}] \)

\[ \Phi(\alpha, \beta) = \log E[Z_{\alpha,\beta}] \]

Theorem II:
The local maxima of \( \Phi(\alpha, \beta) \) are in 1-1 correspondence with the attractive fixpoints of the tree recursions.

useful: allows us to do understand specific models

\[
m(\bigcirc) = B \, m_1(\bigcirc)m_2(\bigcirc)m_3(\bigcirc) + m_1(\bigcirc)m_2(\bigcirc)m_3(\bigcirc)
\]

\[
m(\bullet) = m_1(\bullet)m_2(\bullet)m_3(\bullet) + B \, m_1(\bullet)m_2(\bullet)m_3(\bullet)
\]
General spin model

- q = number of spins
- B = q×q interaction matrix
- G=(V,E) an undirected graph

Configuration $\sigma: V \to \{1,2,\ldots,q\}$

$$P(\sigma) = \frac{\prod_{(u,v)\in E} B_{\sigma(u),\sigma(v)}}{Z}$$

(sometimes also an external field)

“Period 2?” (semi-translation invariant Gibbs measure) for general models:

$$r = \frac{B \cdot c^d + 1}{c^d + B}$$

$$c = \frac{B \cdot r^d + 1}{r^d + B}$$

uniqueness = one solution, $r=c=1$

non-uniqueness = multiple solutions
(long-range influence)

$$\hat{R}_i \propto \left( \sum_{j=1}^{q} B_{i,j} C_j \right)^{\Delta-1}$$

and

$$\hat{C}_j \propto \left( \sum_{i=1}^{q} B_{i,j} R_i \right)^{\Delta-1}$$
LOCAL MAXIMA* OF:

\[ \Phi = (\Delta - 1) \left( \sum_{i=1}^{q} \alpha_i \ln \alpha_i + \sum_{i=1}^{q} \beta_j \ln \beta_j \right) + \Delta g(\alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_q), \]

\[ g(\alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_q) = \max \sum_{i=1}^{q} \sum_{j=1}^{q} x_{ij} (\ln(B_{ij}) - \ln x_{ij}), \]

where the maximum is taken over \( x_{ij} \)'s such that

\[ \alpha_i = \sum_j x_{ij} \quad \text{and} \quad \beta_j = \sum_i x_{ij}. \]

STABLE* (=ATTRACTION) SOLUTIONS OF:

\[ \hat{R}_i \propto \left( \sum_{j=1}^{q} B_{ij} C_j \right)^{\Delta-1} \]

and

\[ \hat{C}_j \propto \left( \sum_{i=1}^{q} B_{ij} R_j \right)^{\Delta-1} \]
LOCAL MAXIMA* OF:

$$\Phi = \begin{pmatrix} q & q \\ q & \end{pmatrix}$$

- Jacobian = 0
- Hessian < 0

$\Rightarrow$ local maximum

where the maximum is taken over $x_{ij}$'s such that

$$\alpha_i = \sum_j x_{ij} \quad \text{and} \quad \beta_j = \sum_i x_{ij}.$$
Why useful?

Potts model:

\[ R_i = (C + (B - 1)C_i)^{\Delta^{-1}} \]
\[ C_i = (R + (B - 1)R_i)^{\Delta^{-1}} \]

\[ R = R_1 + \ldots + R_q \]
\[ C = C_1 + \ldots + C_q \]

at most 3 different values of \( R_i \) and \( C_i \)

value occurring > 1 times \( \Rightarrow \) eigenvalue of jacobian,

(less <1 iff “+” root)
Theorem II:

The **local maxima** of $\Phi(\alpha, \beta)$ are in 1-1 correspondence with the **attractive fixpoints** of the tree recursions.

To understand behavior of a spin model with matrix $B$ on random $\Delta$-regular bipartite random graph just iterate tree recursions.
Second moment

1) sample a random $\Delta$-regular (bipartite) graph $G$
   (union of $\Delta$ random matchings)

2) sample a random configuration from the model

How does the result “look” as $n \to \infty$?

$$P(\alpha, \beta) = \frac{Z_{\alpha,\beta}}{Z}$$

goal:
1) understand $E[Z_{\alpha,\beta}]$
2) understand $E[Z_{\alpha,\beta}^2]$
First moment "$E[Z_{\alpha, \beta}]$"

$$\Phi = (\Delta - 1) \left( \sum_{i=1}^{q} \alpha_i \ln \alpha_i + \sum_{i=1}^{q} \beta_j \ln \beta_j \right) + \Delta g(\alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_q),$$

$$g(\alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_q) = \max \sum_{i=1}^{q} \sum_{j=1}^{q} x_{ij} (\ln (B_{ij}) - \ln x_{ij}),$$

where the maximum is taken over $x_{ij}$’s such that

$$\alpha_i = \sum_{j} x_{ij} \quad \text{and} \quad \beta_j = \sum_{i} x_{ij}.$$ 

Second moment "$E[Z_{\alpha, \beta}^2]$"

\[= \text{maximum of the first moment of spin system with interaction matrix } B \otimes B \text{ with additional constraints:} \]

$$\sum_{j} \gamma_{ij} = \alpha_i , \sum_{i} \gamma_{ij} = \alpha_j , \sum_{j} \delta_{ij} = \beta_i , \sum_{i} \gamma_{ij} = \beta_j$$
Induced matrix norms characterize the first moment

**Theorem:**
The local maxima of \( \Phi_1(\alpha, \beta) \) are in 1-1 correspondence with the maxima of

\[
\frac{\|Bx\|_\Delta}{\|x\|_p}
\]

where

\[
p = \frac{\Delta}{\Delta - 1}
\]

Induced matrix norms

\[
\|A\|_{p \to q'} = \max \|A\|_{q'}
\]

\[
\|x\|_p = 1
\]

A useful fact (Bennett'77):

For \( p < q' \)

\[
\|A \otimes B\|_{p \to q'} = \|A\|_{p \to q'} \|B\|_{p \to q'}
\]

*Second moment \( E[Z_{\alpha, \beta}^2] \)*

= maximum of the first moment of spin system with interaction matrix \( B \otimes B \) with additional constraints:

\[
\sum_j \gamma_{ij} = \alpha_i, \quad \sum_i \gamma_{ij} = \alpha_j, \quad \sum_j \delta_{ij} = \beta_i, \quad \sum_i \gamma_{ij} = \beta_j
\]
Induced matrix norms characterize the first moment

\[ \frac{\|Bx\|_\Delta}{\|x\|_p} \]

Theorem:

The local maxima of \( \Phi_1(\alpha, \beta) \) are in 1-1 correspondence with the maxima of

\[ p = \frac{\Delta}{\Delta - 1} \]

\[ \Rightarrow \text{Concentration at the global maxima of } \Phi \]

Second moment "\( E[Z_{\alpha,\beta}^2] \)"

= maximum of the first moment of spin system with interaction matrix \( B \otimes B \) with additional constraints:

\[
\sum_j \gamma_{ij} = \alpha_i, \quad \sum_i \gamma_{ij} = \alpha_j, \quad \sum_j \delta_{ij} = \beta_i, \quad \sum_i \gamma_{ij} = \beta_j
\]
Applications
For which spin models is there an FPRAS for computing the partition function on graphs with max-degree $\leq \Delta$?

FPRAS (fully-polynomial randomized approximation scheme): given $\varepsilon, G$ in time $\text{poly}(|G|, 1/\varepsilon)$ outputs an answer in $[ (1 - \varepsilon) Z, (1 + \varepsilon) Z ]$ with probability $\geq 3/4$. 

General spin model

$q =$ number of spins
$B = q \times q$ interaction matrix
$G = (V, E)$ an undirected graph

Configuration $\sigma$: $V \rightarrow \{1, 2, \ldots, q\}$

$$P(\sigma) = \frac{\prod_{\{u,v\} \in E} B_{\sigma(u), \sigma(v)}}{Z}$$

(sometimes also an external field)
q-colorings (q even)

\[ \frac{\Delta}{\log \Delta} \quad \Delta - \sqrt{\Delta} \quad \Delta - 1 \quad \Delta \quad \frac{11}{6} \Delta \]

Sampling hard (even for triangle-free graphs)

Sampling easy (Vigoda’99)

Generic hardness for anti-ferromagnetic models

Semitranslational non-uniqueness + unique hessian global maxima (up to symmetries of the model) \( \Rightarrow \) sampling hard
Proof flavor
LOCAL MAXIMA* OF:

\[ \Phi = \begin{pmatrix} q & q \\ \vdots & \vdots \end{pmatrix} \]

Jacobian = 0 \quad \Rightarrow \quad local maximum

Hessian < 0

where the maximum is taken over \( x_{ij} \)'s such that

\[ \alpha_i = \sum_j x_{ij} \quad \text{and} \quad \beta_j = \sum_i x_{ij}. \]

STABLE* (=ATTRACTIVE) SOLUTIONS OF:

Jacobian < 1 \quad \Rightarrow \quad stable
LOCAL MAXIMA* OF:

$$\Phi = (\Delta - 1) \left( \sum_{i=1}^{q} \alpha_i \ln \alpha_i + \sum_{i=1}^{q} \beta_j \ln \beta_j \right) + \Delta g(\alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_q),$$

$$g(\alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_q) = \max \sum_{i=1}^{q} \sum_{j=1}^{q} x_{ij} (\ln(B_{ij}) - \ln x_{ij}),$$

where the maximum is taken over $x_{ij}$'s such that

$$\alpha_i = \sum_j x_{ij} \quad \text{and} \quad \beta_j = \sum_i x_{ij}.$$
\[ \Phi = (\Delta - 1) \left( \sum_{i=1}^{q} \alpha_i \ln \alpha_i + \sum_{j=1}^{q} \beta_j \ln \beta_j \right) + \Delta g(\alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_q), \]

From Lagrange multipliers:

\[ g(\alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_q) = - \sum_{i=1}^{q} \sum_{j=1}^{q} B_{ij} R_i C_j \ln(R_i C_j) \]

\[ R_i \sum_{j=1}^{q} B_{ij} C_j = \alpha_i \quad \text{and} \quad C_j \sum_{i=1}^{q} B_{ij} R_i = \beta_j \]

Differentiating:

\[ \Phi' = \sum_{i=1}^{q} \left( ((\Delta - 1)(1 + \ln \alpha_i) - \Delta \ln R_i) \alpha_i' + \sum_{j=1}^{q} \left( ((\Delta - 1)(1 + \ln \beta_j) - \Delta \ln C_j) \beta_j' \right) \right) \]

Critical points now connect to tree recursions

\[ \alpha_i \propto R_i^{\Delta/(\Delta-1)} \quad \text{and} \quad \beta_j \propto C_j^{\Delta/(\Delta-1)} \]
Differentiating some more:

\[
\Phi'' = \sum_{i=1}^{q} \alpha_i \left( \frac{R'_i}{R_i} + \frac{\sum_{j=1}^{q} B_{ij} C'_j}{\sum_{j=1}^{q} B_{ij} C_j} \right) \left( (\Delta - 1) \frac{\sum_{j=1}^{q} B_{ij} C'_j}{\sum_{j=1}^{q} B_{ij} C_j} - \frac{R'_i}{R_i} \right) \\
+ \sum_{j=1}^{q} \beta_j \left( \frac{C'_j}{C_j} + \frac{\sum_{i=1}^{q} B_{ij} R'_i}{\sum_{i=1}^{q} B_{ij} R_i} \right) \left( (\Delta - 1) \frac{\sum_{i=1}^{q} B_{ij} R'_i}{\sum_{i=1}^{q} B_{ij} R_i} - \frac{C'_j}{C_j} \right) \\
= w^T (I + L)((\Delta - 1)L - I)w
\]

where \( L: (r_1, \ldots, r_q, c_1, \ldots, c_q) \mapsto (\hat{r}_1, \ldots, \hat{r}_q, \hat{c}_1, \ldots, \hat{c}_q) \)

\[
\hat{r}_i = \sum_{j=1}^{q} \frac{B_{ij} R_i C_j}{\sqrt{\alpha_i \beta_j}} c_j \quad \text{and} \quad \hat{c}_j = \sum_{i=1}^{q} \frac{B_{ij} R_i C_j}{\sqrt{\alpha_i \beta_j}} r_i
\]

\[
r_i = \sqrt{\alpha_i} R'_i / R_i \quad \text{and} \quad c_j = \sqrt{\beta_j} C'_j / C_j
\]

\[
\sum_{i=1}^{q} \sqrt{\alpha_i} r_i + \sum_{j=1}^{q} \sqrt{\beta_j} c_j = 0
\]

\[
w = (r_1, \ldots, r_q, c_1, \ldots, c_q)
\]
Differentiating the tree recursions:

\[
\hat{R}_i \propto \left( \sum_{j=1}^{q} B_{ij} C_j \right)^{\Delta - 1} \quad \text{and} \quad \hat{C}_j \propto \left( \sum_{i=1}^{q} B_{ij} R_j \right)^{\Delta - 1}
\]

\[
\frac{\hat{R}_i'}{\hat{R}_i} = (\Delta - 1) \frac{\sum_{j=1}^{q} B_{ij} C_j \frac{C_j'}{C_j}}{\sum_{j=1}^{q} B_{ij} C_j} = \quad \text{and} \quad \frac{\hat{C}_j'}{\hat{C}_j} = (\Delta - 1) \frac{\sum_{i=1}^{q} B_{ij} R_i \frac{R_i'}{R_i}}{\sum_{i=1}^{q} B_{ij} R_i}
\]

The map is \((\Delta - 1)L\)

where \(L\): \((r_1, \ldots, r_q, c_1, \ldots, c_q) \mapsto (\hat{r}_1, \ldots, \hat{r}_q, \hat{c}_1, \ldots, \hat{c}_q)\)

\[
\hat{r}_i = \sum_{j=1}^{q} \frac{B_{ij} R_i C_j}{\sqrt{\alpha_i \beta_j}} c_j \quad \text{and} \quad \hat{c}_j = \sum_{i=1}^{q} \frac{B_{ij} R_i C_j}{\sqrt{\alpha_i \beta_j}} r_i
\]

\[
r_i = \sqrt{\alpha_i R_i' / R_i}, \quad \text{and} \quad c_j = \sqrt{\beta_j C_j' / C_j}
\]
Establishing bimodality

First moment

(3G) with

\[ Z_{\text{unbalanced}} > Z_{\text{balanced}} \]

need more \( \Rightarrow \) second moment

\[
E[Z_{\lambda}] \text{ for } \lambda < \lambda_0
\]

\[
E[Z_{\lambda}] \text{ for } \lambda > \lambda_0
\]

\[
\alpha \text{ spin } + 1 \quad \Rightarrow \quad \beta \text{ spin } + 1
\]

\[
(1 - \alpha) \text{ spin } - 1 \quad \Rightarrow \quad (1 - \beta) \text{ spin } - 1
\]

Period 2?

\[
r = \frac{B \ c^d + 1}{c^d + B}
\]

\[
c = \frac{B \ r^d + 1}{r^d + B}
\]

uniqueness = one solution, \( r = c \)

non-uniqueness = multiple solutions

---

**Hessian**

\[
= w^T (I + L)((\Delta - 1)L - I)w
\]

**Jacobian**

\[
= (\Delta - 1)L
\]

Hessian < 0 \( \Leftrightarrow \) Jacobian < 1
Questions
What is the condition for hardness?

Our: antiferromagnetic + semi-translational non-uniqueness implies hard

Does antiferromagnetic + non-uniqueness imply hard?

Positive results?

colorings: \( k \geq 2\Delta \) easy (Jerrum’95)
\( k \geq 2.58\Delta+1 \) easy (Lu, Yin’13)
Proof flavor

Does non-uniqueness manifest macroscopically (on a useful gadget)?