Approximating 2-State Spin Systems

Heng Guo

Queen Mary, University of London

Based on joint work with Pinyan Lu,

and with Leslie Ann Goldberg

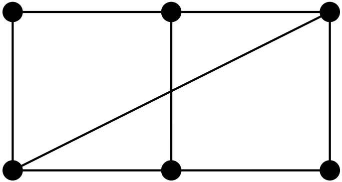
Simons Institute

Mar 28 2016

Ising Model

Edge interaction

	0	1
0	β	1
1	1	β



Configuration $\sigma : V \to \{0, 1\}$

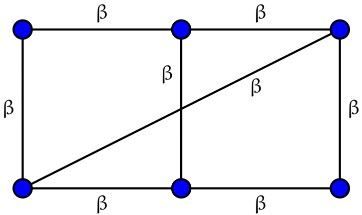
$$w(\sigma) = \beta^{mono(\sigma)}$$

$$\pi(\sigma) \sim w(\sigma)$$

Ising Model

Edge interaction

	0	1
0	β	1
1	1	β



Configuration $\sigma : V \to \{0, 1\}$

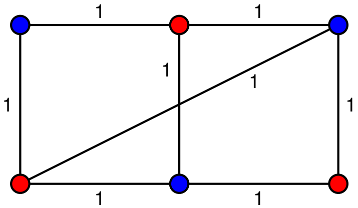
$$w(\sigma) = \beta^8$$

$$\pi(\sigma) \sim w(\sigma)$$

Ising Model

Edge interaction

	0	1
0	β	1
1	1	β



Configuration $\sigma : V \to \{0, 1\}$

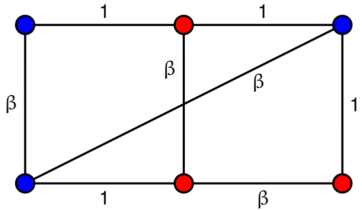
$$w(\sigma) = \beta^0 = 1$$

$$\pi(\sigma) \sim w(\sigma)$$

Ising Model

Edge interaction

	0	1
0	β	1
1	1	β



Partition function (normalizing factor):

$$Z_G(\beta) = \sum_{\sigma:V\to\{0,1\}} w(\sigma)$$

where $w(\sigma) = \beta^{mono(\sigma)}$, $mono(\sigma)$ is the number of monochromatic edges under σ.

2-State Spin System

Edge:

	0	1
0	β	1
1	1	β

Vertex:

0	1
1	1

More generally, three parameters β, γ, and λ.

$$w(\sigma) = \beta^{m_0(\sigma)}\gamma^{m_1(\sigma)}\lambda^{n_0(\sigma)}$$

$m_0(\sigma)$: # of $(0,0)$ edges;

$m_1(\sigma)$: # of $(1,1)$ edges;

$n_0(\sigma)$: # of 0 vertices.

$$Z_G(\beta,\gamma,\lambda) = \sum_{\sigma:V\to\{0,1\}} w(\sigma)$$

2-State Spin System

Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \beta \end{bmatrix}$ Vertex: $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

More generally, three parameters β, γ, and λ.

$$w(\sigma) = \beta^{m_0(\sigma)} \gamma^{m_1(\sigma)} \lambda^{n_0(\sigma)}$$

$m_0(\sigma)$: # of $(0,0)$ edges;

$m_1(\sigma)$: # of $(1,1)$ edges;

$n_0(\sigma)$: # of 0 vertices.

$$Z_G(\beta, \gamma, \lambda) = \sum_{\sigma : V \to \{0,1\}} w(\sigma)$$

Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$

More generally, three parameters β, γ, and λ.

$$w(\sigma) = \beta^{m_0(\sigma)} \gamma^{m_1(\sigma)} \lambda^{n_0(\sigma)}$$

$m_0(\sigma)$: # of $(0,0)$ edges;

$m_1(\sigma)$: # of $(1,1)$ edges;

$n_0(\sigma)$: # of 0 vertices.

$$Z_G(\beta, \gamma, \lambda) = \sum_{\sigma: V \to \{0,1\}} w(\sigma)$$

2-State Spin System

Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$

More generally, three parameters β, γ, and λ.

$$w(\sigma) = \beta^{m_0(\sigma)}\gamma^{m_1(\sigma)}\lambda^{n_0(\sigma)}$$

$m_0(\sigma)$: # of $(0,0)$ edges;

$m_1(\sigma)$: # of $(1,1)$ edges;

$n_0(\sigma)$: # of 0 vertices.

$$Z_G(\beta,\gamma,\lambda) = \sum_{\sigma:V\to\{0,1\}} w(\sigma)$$

2-State Spin System

Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$

More generally, three parameters β, γ, and λ.

$$w(\sigma) = \beta^{m_0(\sigma)}\gamma^{m_1(\sigma)}\lambda^{n_0(\sigma)}$$

$m_0(\sigma)$: # of $(0,0)$ edges;

$m_1(\sigma)$: # of $(1,1)$ edges;

$n_0(\sigma)$: # of 0 vertices.

$$Z_G(\beta,\gamma,\lambda) = \sum_{\sigma:V\to\{0,1\}} w(\sigma)$$

Examples

- Ising model: $\begin{bmatrix} \beta & 1 \\ 1 & \beta \end{bmatrix}$ (no field)

$$Z_G(\beta) = \sum_{\sigma : V \to \{0,1\}} \beta^{mono(\sigma)}$$

- Hardcore gas model: $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ and $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$ (Weighted independent set)

$$Z_G(\beta) = \sum_{\text{Independent set } I} \lambda^{|I|}$$

Examples

- Ising model: $\begin{bmatrix} \beta & 1 \\ 1 & \beta \end{bmatrix}$ (no field)

$$Z_G(\beta) = \sum_{\sigma : V \to \{0,1\}} \beta^{mono(\sigma)}$$

- Hardcore gas model: $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ and $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$ (Weighted independent set)

$$Z_G(\beta) = \sum_{\text{Independent set } I} \lambda^{|I|}$$

Approximate Counting

- Exact evaluating Z is **#P**-hard unless $\beta\gamma = 1$ or $\beta = \gamma = 0$ or $\lambda = 0$.
- Approximate the partition function Z.
  - Fully Polynomial-time Randomized Approximation Scheme (FPRAS) and FPTAS:
    polynomial time in n and $\frac{1}{\varepsilon}$ (multiplicative error ε).
- Approximating Z is equivalent to approximate marginal probabilities p_v due to self-reducibility [Jerrum, Valiant, Vazirani 86] .

Approximate Counting

- Exact evaluating Z is **#P**-hard unless $\beta\gamma = 1$ or $\beta = \gamma = 0$ or $\lambda = 0$.
- Approximate the partition function Z.
  - Fully Polynomial-time Randomized Approximation Scheme (FPRAS) and FPTAS:
    polynomial time in n and $\frac{1}{\varepsilon}$ (multiplicative error ε).
- Approximating Z is equivalent to approximate marginal probabilities p_v due to self-reducibility [Jerrum, Valiant, Vazirani 86] .

Approximate Counting

- Exact evaluating Z is **#P**-hard unless $\beta\gamma = 1$ or $\beta = \gamma = 0$ or $\lambda = 0$.
- Approximate the partition function Z.
  - Fully Polynomial-time Randomized Approximation Scheme (FPRAS) and FPTAS:
    polynomial time in n and $\frac{1}{\varepsilon}$ (multiplicative error ε).
- Approximating Z is equivalent to approximate marginal probabilities p_v due to self-reducibility [Jerrum, Valiant, Vazirani 86] .

Ferromagnetic and Anti-ferromagnetic

Edge Interaction

$$\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$$

- If $\beta\gamma = 1$, then the 2-spin system is trivial.
- Ferromagnetic Ising: $\beta\gamma > 1$.

  Neighbours tend to have the same spin.
- Anti-ferromagnetic Ising: $\beta\gamma < 1$.

  Neighbours tend to have different spins.

Edge Interaction

$$\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$$

- If $\beta\gamma = 1$, then the 2-spin system is trivial.

- Ferromagnetic Ising: $\beta\gamma > 1$.

  Neighbours tend to have the same spin.

- Anti-ferromagnetic Ising: $\beta\gamma < 1$.

  Neighbours tend to have different spins.

Ferromagnetic and Anti-ferromagnetic

Edge Interaction

$$\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$$

- If $\beta\gamma = 1$, then the 2-spin system is trivial.

- Ferromagnetic Ising: $\beta = \gamma > 1$. $\beta\gamma > 1$.

  Neighbours tend to have the same spin.

- Anti-ferromagnetic Ising: $\beta = \gamma < 1$. $\beta\gamma < 1$.

  Neighbours tend to have different spins.

Ferromagnetic and Anti-ferromagnetic

Edge Interaction

$$\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$$

- If $\beta\gamma = 1$, then the 2-spin system is trivial.
- Ferromagnetic Ising: $\beta\gamma > 1$.

  Neighbours tend to have the same spin.
- Anti-ferromagnetic Ising: $\beta\gamma < 1$.

  Neighbours tend to have different spins.

Outline

1. Anti-ferromagnetic 2-Spin Systems
2. Ferromagnetic 2-Spin Systems
3. Complex weighted Ising models

   (approximation of $|Z|$)

Anti-ferromagnetic Systems

Anti-ferromagnetic Systems

For antiferro systems,

$$\text{FPTAS for } Z \quad \Leftrightarrow \quad \text{Correlation decays}$$

Computational Transition

Approximate counting weighted independent sets (Hardcore model)

Edge: $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$

Computational Transition

Approximate counting weighted independent sets (Hardcore model)

Edge: $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$

For G with a bounded degree Δ:

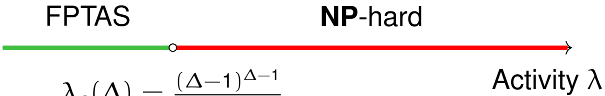
FPTAS **NP**-hard

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

Approximate counting weighted independent sets (Hardcore model)

Edge: $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$

For G with a bounded degree Δ:



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

Approximate counting weighted independent sets (Hardcore model)

Edge: $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$

For G with a bounded degree Δ:

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

- Algorithm: [Weitz 06]

Computational Transition

Approximate counting weighted independent sets (Hardcore model)

Edge: $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$

For G with a bounded degree Δ:

FPTAS **NP**-hard

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

Activity λ

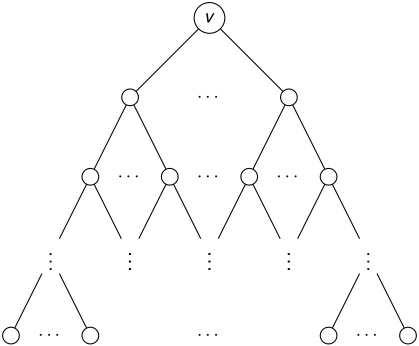
- Algorithm: [Weitz 06]
- Hardness: [Sly 10] [Galanis, Štefankovič, Vigoda 12] [Sly Sun 14]

Uniqueness Transition

- $\lambda_c(\Delta)$: uniqueness threshold of Gibbs measures in $\mathbb{T}_\Delta$.
- Two extremal cases: all leaves are 0 or 1.

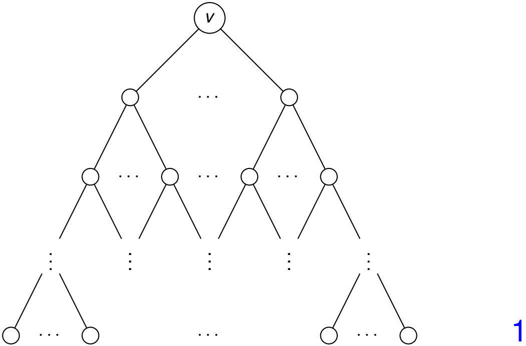
Uniqueness Transition

- $\lambda_c(\Delta)$: uniqueness threshold of Gibbs measures in $\mathbb{T}_\Delta$.
- Two extremal cases: all leaves are 0 or 1.



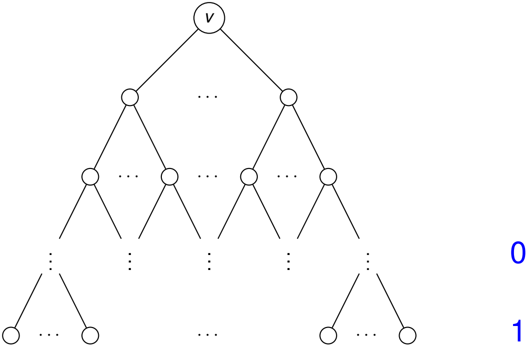
Uniqueness Transition

- $\lambda_c(\Delta)$: uniqueness threshold of Gibbs measures in $\mathbb{T}_\Delta$.
- Two extremal cases: all leaves are 0 or 1.



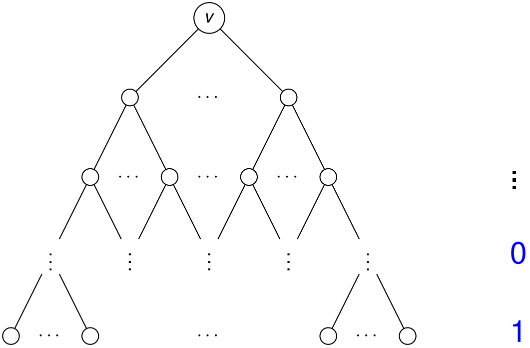
Uniqueness Transition

- $\lambda_c(\Delta)$: uniqueness threshold of Gibbs measures in $\mathbb{T}_\Delta$.
- Two extremal cases: all leaves are 0 or 1.



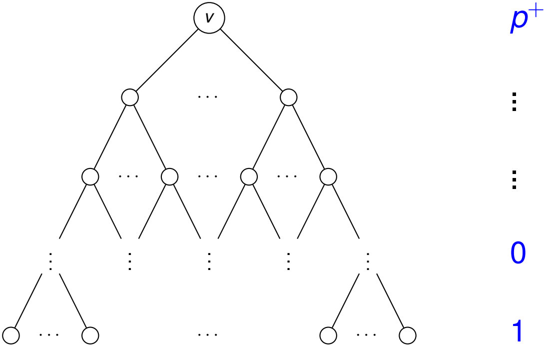
Uniqueness Transition

- $\lambda_c(\Delta)$: uniqueness threshold of Gibbs measures in $\mathbb{T}_\Delta$.
- Two extremal cases: all leaves are 0 or 1.



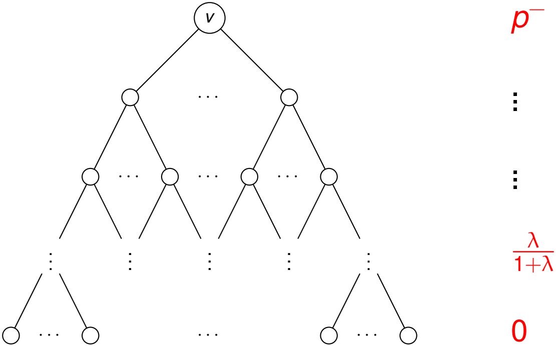
Uniqueness Transition

- $\lambda_c(\Delta)$: uniqueness threshold of Gibbs measures in $\mathbb{T}_\Delta$.
- Two extremal cases: all leaves are 0 or 1.



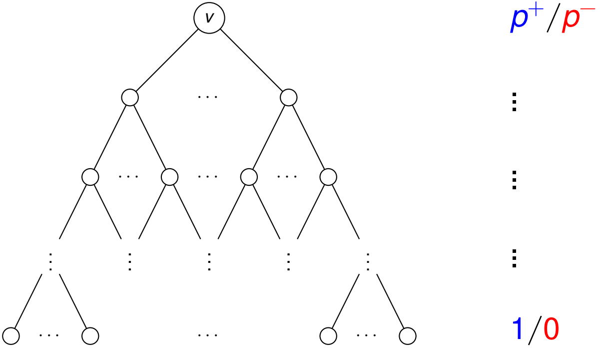
Uniqueness Transition

- $\lambda_c(\Delta)$: uniqueness threshold of Gibbs measures in $\mathbb{T}_\Delta$.
- Two extremal cases: all leaves are 0 or 1.



Uniqueness Transition

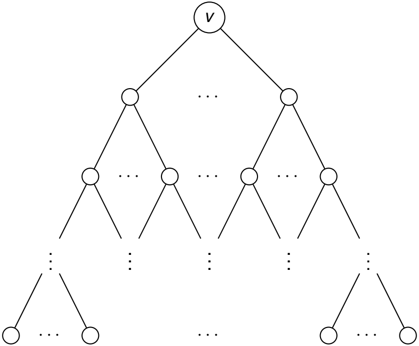
- $\lambda_c(\Delta)$: uniqueness threshold of Gibbs measures in $\mathbb{T}_\Delta$.
- Two extremal cases: all leaves are 0 or 1.



Does $|p^+ - p^-|$ go to 0 or not?

Uniqueness Transition

- $\lambda_c(\Delta)$: uniqueness threshold of Gibbs measures in $\mathbb{T}_\Delta$.
- Two extremal cases: all leaves are 0 or 1.



p^+/p^-

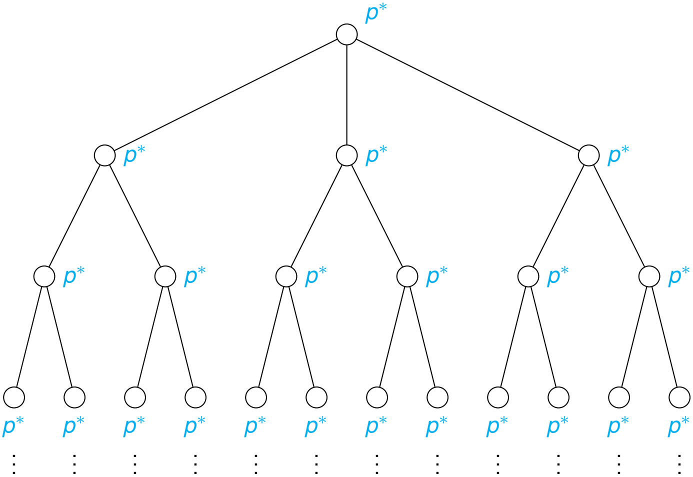
$\vdots$

$1/0$

$$|p^+ - p^-| \to 0 \quad \Leftrightarrow \quad \lambda \leqslant \lambda_c(\Delta).$$

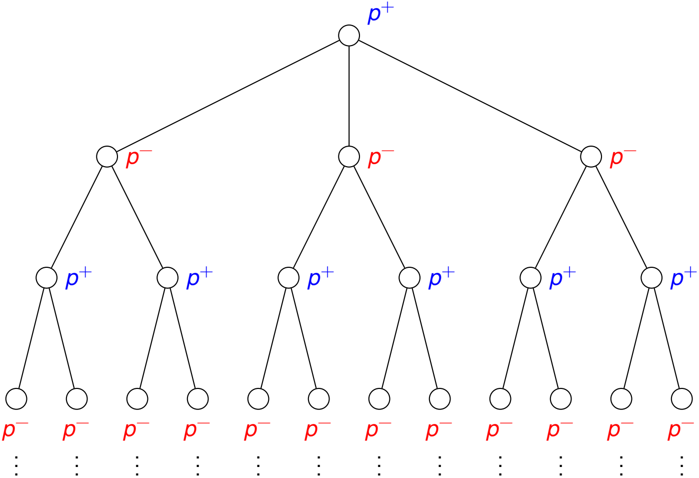
Uniqueness Transition (cont.)

$$\lambda \leqslant \lambda_c(\Delta)$$



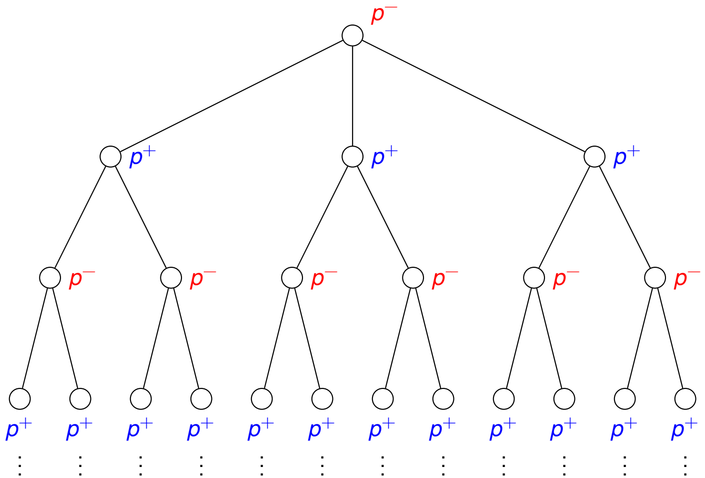
Uniqueness Transition (cont.)

$$\lambda > \lambda_c(\Delta)$$



Uniqueness Transition (cont.)

$$\lambda > \lambda_c(\Delta)$$



Weak and Strong Spatial Mixing

- WSM: Let σ_Λ and τ_Λ be two partial configurations on Λ,

$$|p_v^{\sigma_\Lambda} - p_v^{\tau_\Lambda}| \leqslant \exp(-\Omega(\mathrm{dist}(v, \Lambda)))$$

Weak and Strong Spatial Mixing

- WSM: Let σ_Λ and τ_Λ be two partial configurations on Λ,

$$|p_v^{\sigma_\Lambda} - p_v^{\tau_\Lambda}| \leqslant \exp(-\Omega(\mathrm{dist}(v, \Lambda)))$$

- SSM: Let S be the set where σ_Λ and τ_Λ differ,

$$|p_v^{\sigma_\Lambda} - p_v^{\tau_\Lambda}| \leqslant \exp(-\Omega(\mathrm{dist}(v, S)))$$

Weak and Strong Spatial Mixing

- WSM: Let σ_Λ and τ_Λ be two partial configurations on Λ,

$$|p_v^{\sigma_\Lambda} - p_v^{\tau_\Lambda}| \leqslant \exp(-\Omega(\text{dist}(v, \Lambda)))$$

- SSM: Let S be the set where σ_Λ and τ_Λ differ,

$$|p_v^{\sigma_\Lambda} - p_v^{\tau_\Lambda}| \leqslant \exp(-\Omega(\text{dist}(v, S)))$$

- SSM $\Rightarrow$ WSM $\Leftrightarrow$ Uniqueness

Weak and Strong Spatial Mixing

- WSM: Let σ_Λ and τ_Λ be two partial configurations on Λ,

$$|p_v^{\sigma_\Lambda} - p_v^{\tau_\Lambda}| \leqslant \exp(-\Omega(\mathrm{dist}(v, \Lambda)))$$

- SSM: Let S be the set where σ_Λ and τ_Λ differ,

$$|p_v^{\sigma_\Lambda} - p_v^{\tau_\Lambda}| \leqslant \exp(-\Omega(\mathrm{dist}(v, S)))$$

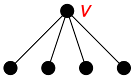
- SSM $\Rightarrow$ WSM $\Leftrightarrow$ Uniqueness
- SSM in $\mathbb{T}_\Delta \Rightarrow$ FPTAS in graphs of degree $\leqslant \Delta$ [Weitz 06]

Breaking Cycles

Goal: calculate marginal probabilities using tree recursions.

Replace a vertex of degree d with d copies.

$$R_v = \frac{\Pr(v = 0)}{\Pr(v = 1)}$$

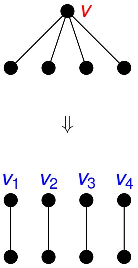


Breaking Cycles

Goal: calculate marginal probabilities using tree recursions.

Replace a vertex of degree d with d copies.

$$R_v = \frac{\Pr(v = 0)}{\Pr(v = 1)} = \frac{\Pr(v_1 = 0, \ldots, v_d = 0)}{\Pr(v_1 = 1, \ldots, v_d = 1)}$$

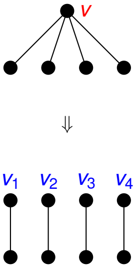


Breaking Cycles

Goal: calculate marginal probabilities using tree recursions.

Replace a vertex of degree d with d copies.

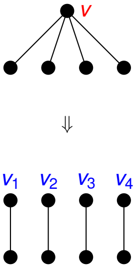
$$R_v = \frac{\Pr(v = 0)}{\Pr(v = 1)} = \frac{\Pr(v_1 = 0, \ldots, v_d = 0)}{\Pr(v_1 = 1, \ldots, v_d = 1)}$$
$$= \frac{\Pr(0000)}{\Pr(0001)} \cdot \frac{\Pr(0001)}{\Pr(0011)} \cdot \frac{\Pr(0011)}{\Pr(0111)} \cdot \frac{\Pr(0111)}{\Pr(1111)}$$



Goal: calculate marginal probabilities using tree recursions.

Replace a vertex of degree d with d copies.

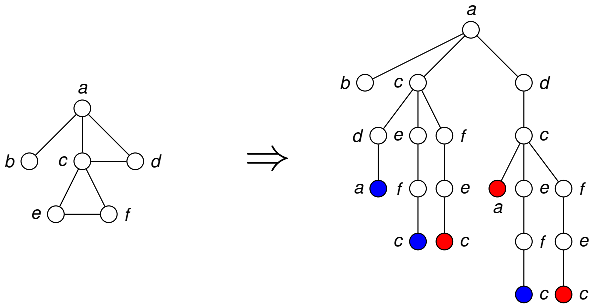
$$R_v = \frac{\Pr(v=0)}{\Pr(v=1)} = \frac{\Pr(v_1=0,\ldots,v_d=0)}{\Pr(v_1=1,\ldots,v_d=1)}$$
$$= \frac{\Pr(0000)}{\Pr(0001)} \cdot \frac{\Pr(0001)}{\Pr(0011)} \cdot \frac{\Pr(0011)}{\Pr(0111)} \cdot \frac{\Pr(0111)}{\Pr(1111)}$$



Each term $\frac{\Pr(0011)}{\Pr(0111)}$ can be viewed as the marginal ratio of v_i conditioned on a certain configuration of other v_j's.

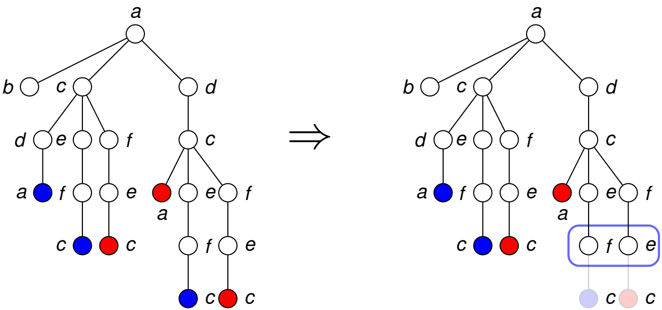
Self-Avoiding Walk (SAW) Tree

- SAW tree is essentially the tree of self-avoiding walks originating at v except that the vertices closing a cycle are also included in the tree.
  - Cycle-closing vertices are fixed according to the rule in the last slide.
- Do the tree recursion to calculate p_v.



- However, SAW tree is of exponential size in general.
  - Truncate the recursion within logarithmic depth.
  - SSM bounds the error.

    Non-uniqueness leads to constant error.



Classification of Antiferro 2-Spin Systems

The implication

$$\text{Uniqueness} \Rightarrow \text{SSM}.$$

is established for all anti-ferromagnetic 2-spin systems ($\beta\gamma < 1$).

[Sinclair, Srivastava, Thurley 12] , [Li, Lu, Yin 12,13]

Hence, for any anti-ferromagnetic 2-spin system,

$$\text{Uniqueness} \Leftrightarrow \text{SSM} \Leftrightarrow \text{FPTAS}.$$

(For general graphs, we require uniqueness to hold for all integer degrees.)

The implication

$$\text{Uniqueness} \Rightarrow \text{SSM}.$$

is established for all anti-ferromagnetic 2-spin systems ($\beta\gamma < 1$).

[Sinclair, Srivastava, Thurley 12] , [Li, Lu, Yin 12,13]

Hence, for any anti-ferromagnetic 2-spin system,

$$\text{Uniqueness} \Leftrightarrow \text{SSM} \Leftrightarrow \text{FPTAS}.$$

(For general graphs, we require uniqueness to hold for all integer degrees.)

Ferromagnetic 2-Spin Systems

Ferromagnetic Ising

Ferro ($\beta > 1$) Ising without field:

Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \beta \end{bmatrix}$ Vertex: $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

For fixed Δ:

Uniqueness in $\mathbb{T}_\Delta$ Non-uniqueness

1 $\beta_c(\Delta) = \frac{\Delta}{\Delta-2}$ β

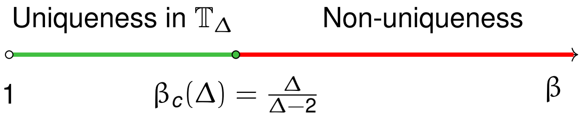
FPRAS [Jerrum, Sinclair 93]

1 (General graphs) β

Ferro ($\beta > 1$) Ising without field:

Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \beta \end{bmatrix}$ Vertex: $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

For fixed Δ:



Ferromagnetic Ising

Ferro ($\beta > 1$) Ising without field:

Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \beta \end{bmatrix}$ Vertex: $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

For fixed Δ:

Uniqueness in $\mathbb{T}_\Delta$ Non-uniqueness

1 $\beta_c(\Delta) = \frac{\Delta}{\Delta - 2}$ β

FPRAS [Jerrum, Sinclair 93]

1 (General graphs) β

Markov chain in the "subgraphs" world:

fast mixing for any $\beta = \gamma > 1$ and $\lambda_v \geqslant 1$ (or $\leqslant 1$) for all $v \in V$.

(even if uniqueness or SSM fails) [Jerrum, Sinclair 93]

- Extended to $\lambda_v \leqslant \frac{\gamma}{\beta}$ (if $\beta \leqslant \gamma$)

  [Goldberg, Jerrum, Paterson 03] , [Liu, Lu, Zhang 14] .

Markov chain in the "subgraphs" world:

fast mixing for any $\beta = \gamma > 1$ and $\lambda_v \geqslant 1$ (or $\leqslant 1$) for all $v \in V$.

(even if uniqueness or SSM fails) [Jerrum, Sinclair 93]

- Extended to $\lambda_v \leqslant \frac{\gamma}{\beta}$ (if $\beta \leqslant \gamma$)

  [Goldberg, Jerrum, Paterson 03] , [Liu, Lu, Zhang 14] .

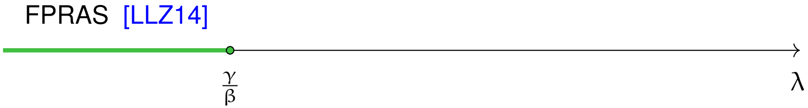
Ferro 2-Spin

Ferro 2-spin systems: Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda_v \\ 1 \end{bmatrix}$

Ferro 2-Spin

Ferro 2-spin systems: Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda_v \\ 1 \end{bmatrix}$

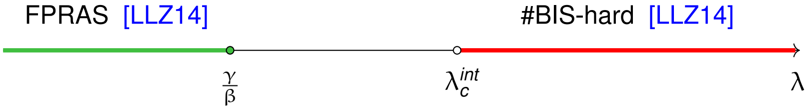
For general graph G, assuming $\beta \leqslant \gamma$:



Ferro 2-Spin

Ferro 2-spin systems: Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda_v \\ 1 \end{bmatrix}$

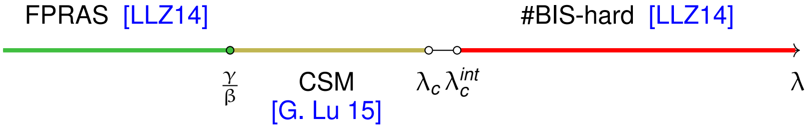
For general graph G, assuming $\beta \leqslant \gamma$:



$\lambda_c^{int} = \left(\frac{\gamma}{\beta}\right)^{(\lfloor\Delta_c\rfloor+1)/2}$, where $\Delta_c = \frac{2\sqrt{\beta\gamma}}{\sqrt{\beta\gamma}-1}$.

Ferro 2-spin systems: Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda_v \\ 1 \end{bmatrix}$

For general graph G, assuming $\beta \leqslant \gamma$:

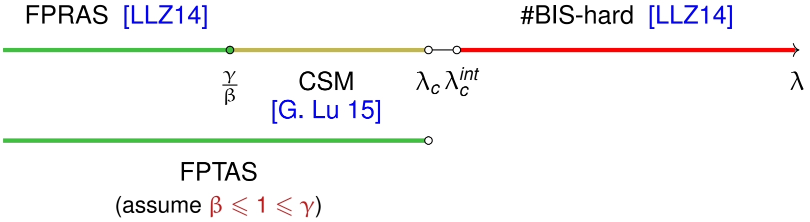


$$\lambda_c^{int} = \left(\frac{\gamma}{\beta}\right)^{(\lfloor \Delta_c \rfloor + 1)/2}, \text{ where } \Delta_c = \frac{2\sqrt{\beta\gamma}}{\sqrt{\beta\gamma}-1}. \quad \lambda_c = \left(\frac{\gamma}{\beta}\right)^{\Delta_c/2}$$

Ferro 2-Spin

Ferro 2-spin systems: Edge: $\begin{bmatrix} \beta & 1 \\ 1 & \gamma \end{bmatrix}$ Vertex: $\begin{bmatrix} \lambda_v \\ 1 \end{bmatrix}$

For general graph G, assuming $\beta \leqslant \gamma$:



$$\lambda_c^{int} = \left(\frac{\gamma}{\beta}\right)^{(\lfloor \Delta_c \rfloor + 1)/2}, \text{ where } \Delta_c = \frac{2\sqrt{\beta\gamma}}{\sqrt{\beta\gamma}-1}. \quad \lambda_c = \left(\frac{\gamma}{\beta}\right)^{\Delta_c/2}$$

Conditional Spatial Mixing

If $\lambda_v < \lambda_c$ for all v, conditional spatial mixing holds in arbitrary trees:

Instead of worst case configurations in SSM, we only allow partial configurations that are dominated by the product measure of isolated vertices ($p_v \leqslant \frac{\lambda}{1+\lambda}$).

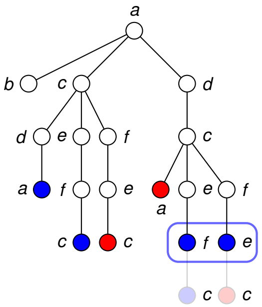
(All vertices are leaning towards the good spin.)

If $\lambda_v < \lambda_c$ for all v, conditional spatial mixing holds in arbitrary trees:

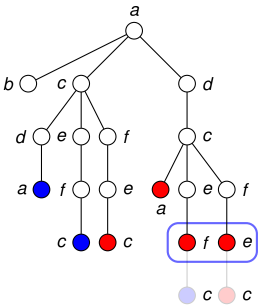
Instead of worst case configurations in SSM, we only allow partial configurations that are dominated by the product measure of isolated vertices ($p_v \leqslant \frac{\lambda}{1+\lambda}$).

(All vertices are leaning towards the good spin.)

SSM:



V.S.



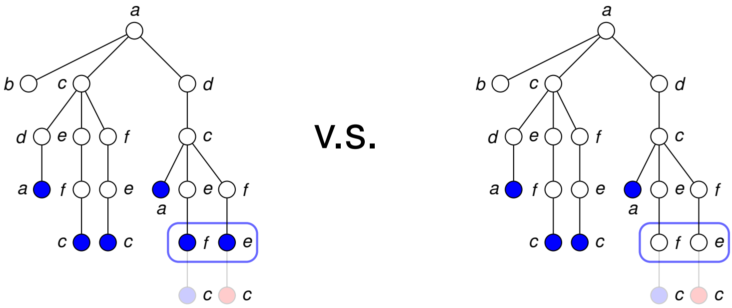
Conditional Spatial Mixing

If $\lambda_v < \lambda_c$ for all v, conditional spatial mixing holds in arbitrary trees:

Instead of worst case configurations in SSM, we only allow partial configurations that are dominated by the product measure of isolated vertices ($p_v \leqslant \frac{\lambda}{1+\lambda}$).

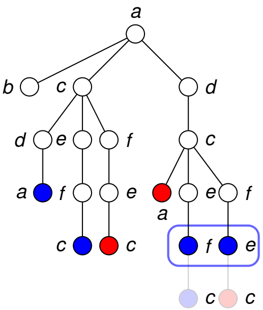
(All vertices are leaning towards the good spin.)

Conditional spatial mixing:

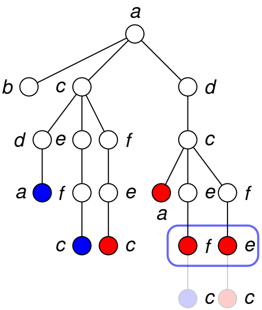


Pruning

If $\beta \leqslant 1 < \gamma$, in the SAW tree, we may first remove "bad" pinnings, the effective field is smaller (better).

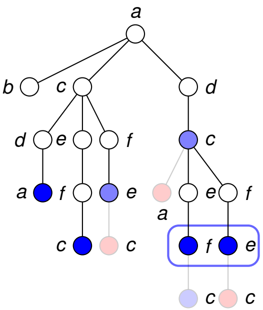


V.S.

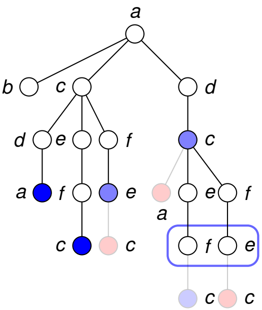


Pruning

If $\beta \leqslant 1 < \gamma$, in the SAW tree, we may first remove “bad” pinnings, the effective field is smaller (better).



V.S.

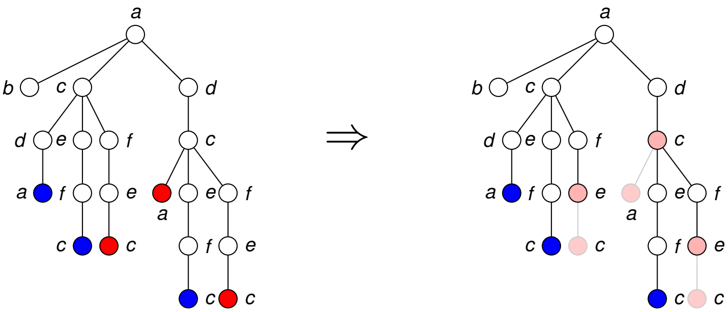


CSM $\Rightarrow$ SSM

What about $\beta > 1$?

If $\beta > 1$, then pruning fails.

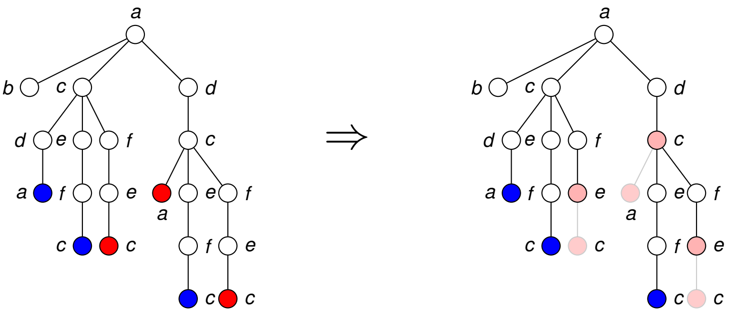
In fact, there is no λ such that SSM holds for general trees.



What about $\beta > 1$?

If $\beta > 1$, then pruning fails.

In fact, there is no λ such that SSM holds for general trees.



However, if $\lambda_v \leqslant \lambda_c$, then $p_v \leqslant \frac{\lambda}{1+\lambda}$ for any graph G.

FPTAS without SSM?

The Exact Threshold?

Our result is tight up to an integrality gap.

However, neither λ_c nor λ_c^{int} is the right bound.

- There exists a small interval beyond λ_c where FPTAS still exists.
  - Since degrees have to be integers.
- There is a $\lambda < \lambda_c^{int}$ such that SSM fails (in an irregular tree).

$$\text{Uniqueness (in } \mathbb{T}_\Delta) \not\Rightarrow \text{SSM}$$

(even if $\beta \leqslant 1 < \gamma$)

The Exact Threshold?

Our result is tight up to an integrality gap.

However, neither λ_c nor λ_c^{int} is the right bound.

- There exists a small interval beyond λ_c where FPTAS still exists.
  - Since degrees have to be integers.
- There is a $\lambda < \lambda_c^{int}$ such that SSM fails (in an irregular tree).

$$\text{Uniqueness (in } \mathbb{T}_\Delta) \not\Rightarrow \text{SSM}$$

(even if $\beta \leqslant 1 < \gamma$)

The Exact Threshold?

Our result is tight up to an integrality gap.

However, neither λ_c nor λ_c^{int} is the right bound.

- There exists a small interval beyond λ_c where FPTAS still exists.
    - Since degrees have to be integers.
- There is a $\lambda < \lambda_c^{int}$ such that SSM fails (in an irregular tree).

$$\text{Uniqueness (in } \mathbb{T}_\Delta) \not\Rightarrow \text{SSM}$$

(even if $\beta \leqslant 1 < \gamma$)

Complex Ising Model

Complex Ising Model

Complex-weighted Ising model: $\begin{bmatrix} \beta & 1 \\ 1 & \beta \end{bmatrix}$ (no field) with $\beta \in \mathbb{C}$

$$Z_G(\beta) = \sum_{\sigma:V\to\{0,1\}} \beta^{mono(\sigma)}$$

Exact evaluation of $Z_G(\beta)$:

- #**P**-hard unless $\beta = 0, \pm 1, \pm \mathtt{i}$. [Jaeger, Vertigan, Welsh 90]

Lemma (Fuiji, Morimae 13)

Given an **IQP** circuit C and an output **x**, there is a graph G such that the marginal probability of **x** equals to $|Z_G(e^{\pi \mathtt{i}/4})|$ up to an easy to compute factor.

Complex Ising Model

Complex-weighted Ising model: $\begin{bmatrix} \beta & 1 \\ 1 & \beta \end{bmatrix}$ (no field) with $\beta \in \mathbb{C}$

$$Z_G(\beta) = \sum_{\sigma : V \to \{0,1\}} \beta^{mono(\sigma)}$$

Exact evaluation of $Z_G(\beta)$:

- #**P**-hard unless $\beta = 0, \pm 1, \pm \mathtt{i}$. [Jaeger, Vertigan, Welsh 90]

Lemma (Fuiji, Morimae 13)

Given an **IQP** circuit C and an output $\mathbf{x}$, there is a graph G such that the marginal probability of $\mathbf{x}$ equals to $|Z_G(e^{\pi \mathtt{i}/4})|$ up to an easy to compute factor.

Complex Ising Model

Complex-weighted Ising model: $\begin{bmatrix} \beta & 1 \\ 1 & \beta \end{bmatrix}$ (no field) with $\beta \in \mathbb{C}$

$$Z_G(\beta) = \sum_{\sigma:V\to\{0,1\}} \beta^{mono(\sigma)}$$

Exact evaluation of $Z_G(\beta)$:

- #**P**-hard unless $\beta = 0, \pm 1, \pm \mathtt{i}$. [Jaeger, Vertigan, Welsh 90]

Lemma (Fuiji, Morimae 13)

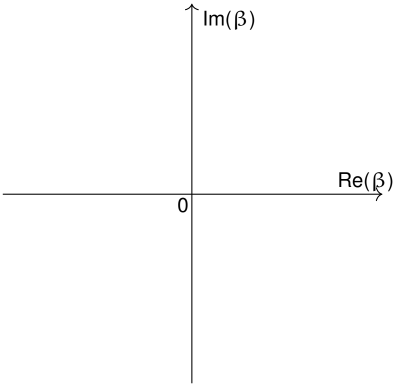
Given an **IQP** circuit C and an output $\mathbf{x}$, there is a graph G such that the marginal probability of $\mathbf{x}$ equals to $|Z_G(e^{\pi \mathtt{i}/4})|$ up to an easy to compute factor.

Approximating $|Z_G(\beta)|$

Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

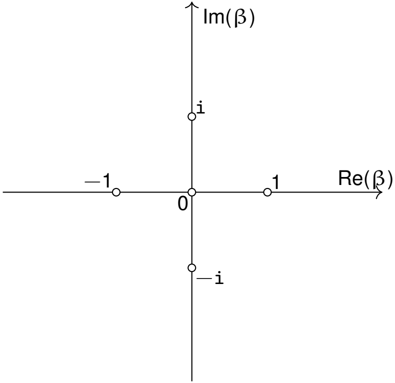
Approximating $|Z_G(\beta)|$

Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.



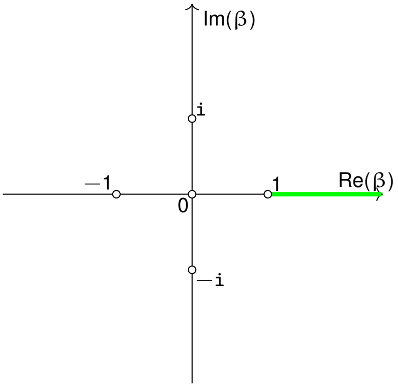
Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]



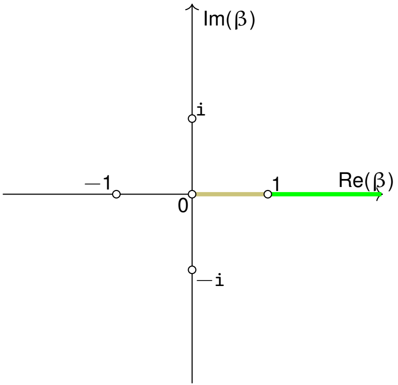
Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]
- $\beta \in (1, \infty)$, FPRAS. [JS93]



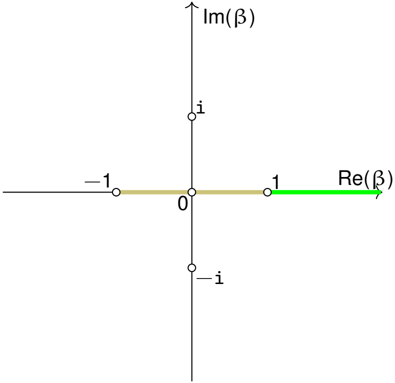
Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]
- $\beta \in (1, \infty)$, FPRAS. [JS93]
- $\beta \in (0, 1)$, **NP**-hard. [JS93]



Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

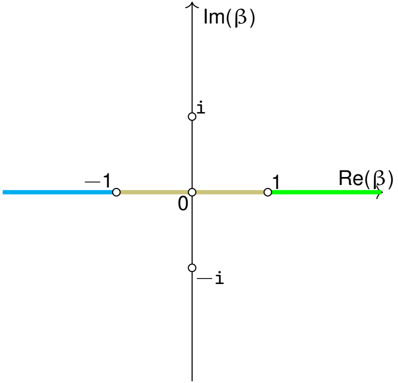
- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]
- $\beta \in (1, \infty)$, FPRAS. [JS93]
- $\beta \in (0, 1)$, **NP**-hard. [JS93]
- $\beta \in (-1, 0)$, **NP**-hard. [GJ08]



Approximating $|Z_G(\beta)|$

Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

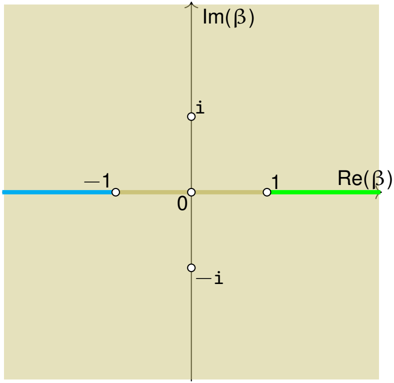
- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]
- $\beta \in (1, \infty)$, FPRAS. [JS93]
- $\beta \in (0, 1)$, **NP**-hard. [JS93]
- $\beta \in (-1, 0)$, **NP**-hard. [GJ08]
- $\beta \in (-\infty, -1)$, #PM. [GJ08]



Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]
- $\beta \in (1, \infty)$, FPRAS. [JS93]
- $\beta \in (0, 1)$, NP-hard. [JS93]
- $\beta \in (-1, 0)$, NP-hard. [GJ08]
- $\beta \in (-\infty, -1)$, #PM. [GJ08]
- $\beta \notin \mathbb{R} \cup \{\mathtt{i}, -\mathtt{i}\}$, NP-hard.

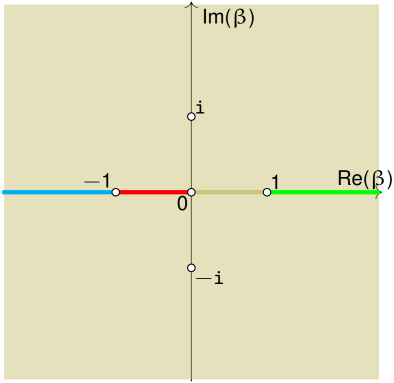
  [Goldberg, G. 14]



Approximating $|Z_G(\beta)|$

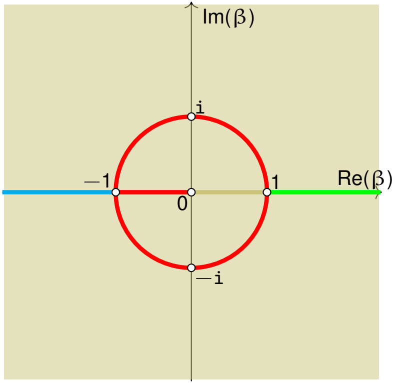
Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]
- $\beta \in (1, \infty)$, FPRAS. [JS93]
- $\beta \in (0, 1)$, **NP**-hard. [JS93]
- $\beta \in (-1, 0)$, **NP**-hard. [GJ08]
- $\beta \in (-\infty, -1)$, #PM. [GJ08]
- $\beta \notin \mathbb{R} \cup \{\mathtt{i}, -\mathtt{i}\}$, **NP**-hard. [GG14]
- $\beta \in (-1, 0)$, #**P**-hard. [GG14]



Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

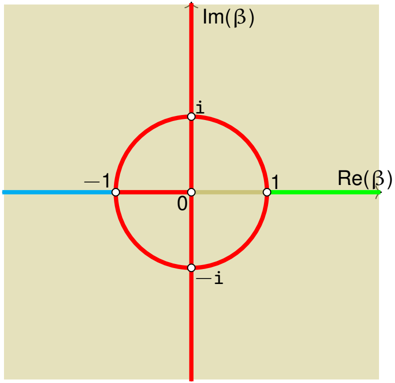
- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]
- $\beta \in (1, \infty)$, FPRAS. [JS93]
- $\beta \in (0, 1)$, NP-hard. [JS93]
- $\beta \in (-1, 0)$, NP-hard. [GJ08]
- $\beta \in (-\infty, -1)$, #PM. [GJ08]
- $\beta \notin \mathbb{R} \cup \{\mathtt{i}, -\mathtt{i}\}$, NP-hard. [GG14]
- $\beta \in (-1, 0)$, #P-hard. [GG14]
- $|\beta| = 1$, $\beta \notin \{\pm 1, \pm \mathtt{i}\}$, #P-hard. [GG14]



Approximating $|Z_G(\beta)|$

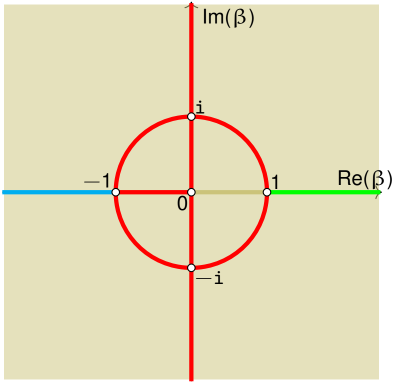
Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]
- $\beta \in (1, \infty)$, FPRAS. [JS93]
- $\beta \in (0, 1)$, **NP**-hard. [JS93]
- $\beta \in (-1, 0)$, **NP**-hard. [GJ08]
- $\beta \in (-\infty, -1)$, #PM. [GJ08]
- $\beta \notin \mathbb{R} \cup \{\mathtt{i}, -\mathtt{i}\}$, **NP**-hard. [GG14]
- $\beta \in (-1, 0)$, #**P**-hard. [GG14]
- $|\beta| = 1$, $\beta \notin \{\pm 1, \pm \mathtt{i}\}$, #**P**-hard. [GG14]
- Re(β)=0, $\beta \notin \{0, \pm \mathtt{i}\}$, #**P**-hard. [GG14]



Approximation complexity of $|Z_G(\beta)|$ for $\beta \in \mathbb{C}$.

- $\beta \in \{0, \pm 1, \pm \mathtt{i}\}$, tractable. [JVW90]
- $\beta \in (1, \infty)$, FPRAS. [JS93]
- $\beta \in (0, 1)$, **NP**-hard. [JS93]
- $\beta \in (-1, 0)$, **NP**-hard. [GJ08]
- $\beta \in (-\infty, -1)$, #PM. [GJ08]
- $\beta \notin \mathbb{R} \cup \{\mathtt{i}, -\mathtt{i}\}$, **NP**-hard. [GG14]
- $\beta \in (-1, 0)$, #**P**-hard. [GG14]
- $|\beta| = 1$, $\beta \notin \{\pm 1, \pm \mathtt{i}\}$, #**P**-hard. [GG14]
- Re(β)=0, $\beta \notin \{0, \pm \mathtt{i}\}$, #**P**-hard. [GG14]



- If $\beta = re^{\mathrm{i}\theta}$ where $\theta = \frac{p\pi}{2q}$, p and q are two co-prime positive integers and p is odd, #**P**-hard. [GG14]

#P-hardness

If $Z_G(\beta) = 0$, even the approximation requires the exact answer.

We relax our problem so that if $Z_G(\beta) = 0$, we accept any return.

Our hardness results hold for these relaxed versions.

We reduce #MINIMUM CARDINALITY (s, t)-CUT [Provan, Ball 83] to approximating $|Z_G(\beta)|$ for any $\beta \in (-1, 0)$.

The key part of the #**P**-hardness proof is a bisection argument.

This idea has been used to show hardness of determining signs of Tutte polynomials (at real points). [Goldberg, Jerrum 12]

#P-hardness

If $Z_G(\beta) = 0$, even the approximation requires the exact answer.

We relax our problem so that if $Z_G(\beta) = 0$, we accept any return.

Our hardness results hold for these relaxed versions.

We reduce #MINIMUM CARDINALITY (s, t)-CUT [Provan, Ball 83] to approximating $|Z_G(\beta)|$ for any $\beta \in (-1, 0)$.

The key part of the #**P**-hardness proof is a bisection argument.

This idea has been used to show hardness of determining signs of Tutte polynomials (at real points). [Goldberg, Jerrum 12]

#P-hardness

If $Z_G(\beta) = 0$, even the approximation requires the exact answer.

We relax our problem so that if $Z_G(\beta) = 0$, we accept any return.

Our hardness results hold for these relaxed versions.

We reduce #MINIMUM CARDINALITY (s,t)-CUT [Provan, Ball 83] to approximating $|Z_G(\beta)|$ for any $\beta \in (-1, 0)$.

The key part of the #**P**-hardness proof is a bisection argument.

This idea has been used to show hardness of determining signs of Tutte polynomials (at real points). [Goldberg, Jerrum 12]

The Reduction

- Given a graph G, suppose $C = \#\text{Min-}(s, t)\text{-Cut}$.
  We may assume (s, t) is not in G. Introduce a new edge $e = (s, t)$.
- We want to put a weight x on e and a fixed weight γ on every other edge.
  - Using edge weight β, we build gadgets to implement γ.
    We can also approximate any $x \in$ (-1,0) exponentially accurately.
- Call the graph G_x. Let $f(x) = Z_{G_x}(\gamma)$.
  Notice that $f(x)$ is a linear function in x.
  Let x_0 be the root of $f(x)$.
- Our choice of γ guarantees that $f(0) > 0$, $f(-1) < 0$.
  Moreover if we can approximate x_0 accurately enough, C can be computed exactly.

The Reduction

- Given a graph G, suppose $C = \#\text{Min-}(s,t)\text{-Cut}$.
  We may assume (s,t) is not in G. Introduce a new edge $e = (s,t)$.
- We want to put a weight x on e and a fixed weight γ on every other edge.
  - Using edge weight β, we build gadgets to implement γ.
    We can also approximate any $x \in$ (-1,0) exponentially accurately.
- Call the graph G_x. Let $f(x) = Z_{G_x}(\gamma)$.
  Notice that $f(x)$ is a linear function in x.
  Let x_0 be the root of $f(x)$.
- Our choice of γ guarantees that $f(0) > 0$, $f(\text{-}1) < 0$.
  Moreover if we can approximate x_0 accurately enough, C can be computed exactly.

The Reduction

- Given a graph G, suppose $C = \#\text{Min-}(s,t)\text{-Cut}$.
  We may assume (s,t) is not in G. Introduce a new edge $e = (s,t)$.
- We want to put a weight x on e and a fixed weight γ on every other edge.
  - Using edge weight β, we build gadgets to implement γ.
    We can also approximate any $x \in$ (-1,0) exponentially accurately.
- Call the graph G_x. Let $f(x) = Z_{G_x}(\gamma)$.
  Notice that $f(x)$ is a linear function in x.
  Let x_0 be the root of $f(x)$.
- Our choice of γ guarantees that $f(0) > 0$, $f(\text{-}1) < 0$.
  Moreover if we can approximate x_0 accurately enough, C can be computed exactly.

The Reduction

- Given a graph G, suppose $C = \#\text{Min-}(s,t)\text{-Cut}$.
  We may assume (s,t) is not in G. Introduce a new edge $e = (s,t)$.
- We want to put a weight x on e and a fixed weight γ on every other edge.
  - Using edge weight β, we build gadgets to implement γ.
    We can also approximate any $x \in$ (-1,0) exponentially accurately.
- Call the graph G_x. Let $f(x) = Z_{G_x}(\gamma)$.
  Notice that $f(x)$ is a linear function in x.
  Let x_0 be the root of $f(x)$.
- Our choice of γ guarantees that $f(0) > 0$, $f(-1) < 0$.
  Moreover if we can approximate x_0 accurately enough, C can be computed exactly.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$. We recursively shrink the interval containing x_0.

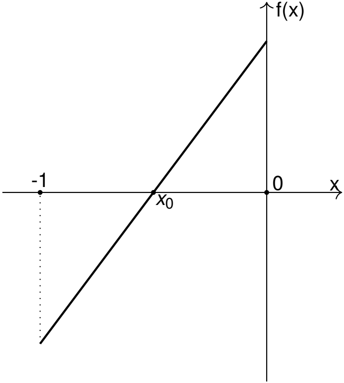
- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$. If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$. We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$. If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

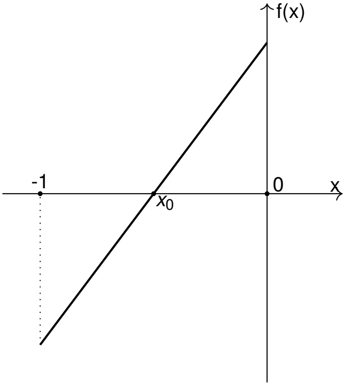


Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$. We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$. If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

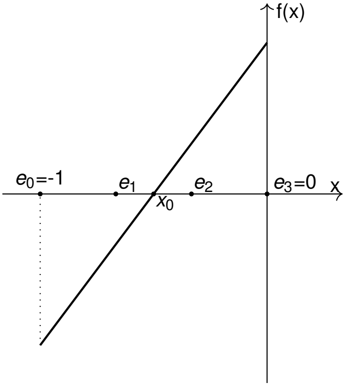


Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$.
We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$.
  If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

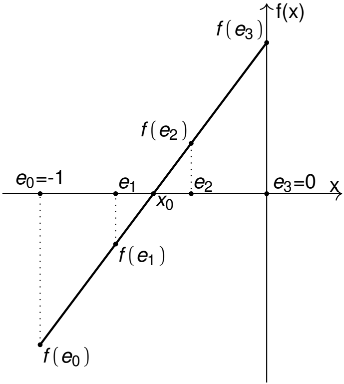


Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$.
We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$.
  If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

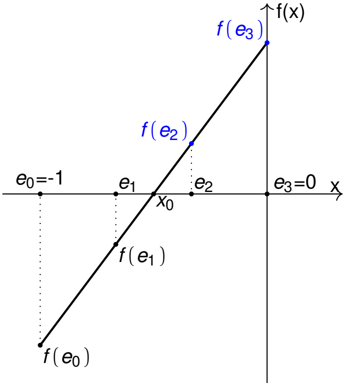


Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$. We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$. If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

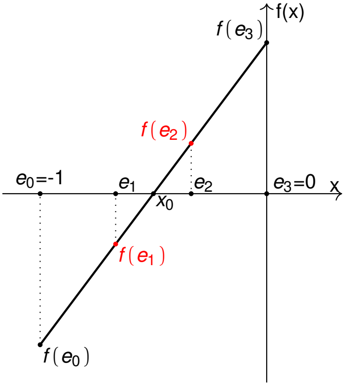


Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$. We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$. If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

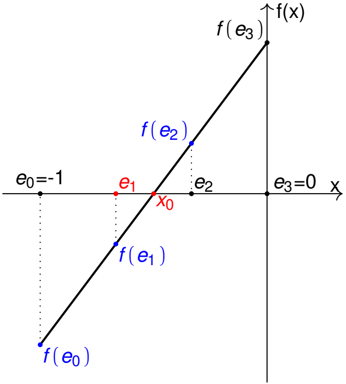


Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$. We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$. If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

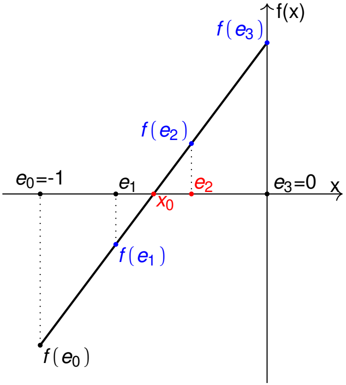


Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$. We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$.
  If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

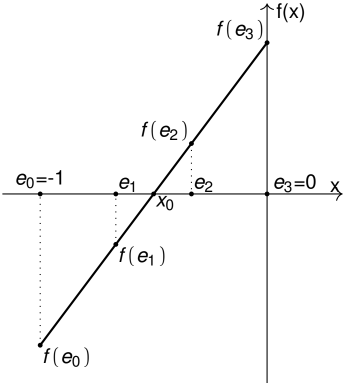


Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$. We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$.
  If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.

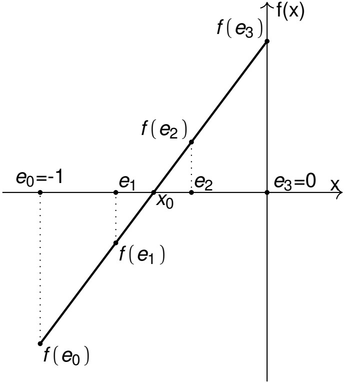


Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Bisection with an Oracle of Approximating Norms

The oracle returns $|f(x)|$ up to some constant K. Call the approximation $g(x)$. We recursively shrink the interval containing x_0.

- We begin with the interval (-1,0).
- Divide the current interval into 3 subintervals.
- Evaluate $|f(x)|$ approximately at the 4 endpoints.
- If two points x_1, x_2 are on the same side of x_0, then the accuracy K guarantees that the ordering of $g(x_1)$ and $g(x_2)$ is the same as that of $|f(x_1)|$ and $|f(x_2)|$.
- Otherwise the order may be wrong, but it happens at most once.
- If $g(e_0) > g(e_1) > g(e_2)$, then $e_1 < x_0$. If $g(e_1) < g(e_2) < g(e_3)$, then $e_2 > x_0$.
- At least one of the cases is true, so we can shrink the interval by $\frac{2}{3}$.



Divide the interval into more subintervals so that we don't need an exact evaluation of $|f(x)|$ at x_0.

Complex Ising with Fields

Edge weight β, external field λ:

$$Z_G(\beta;\lambda) \;=\; \sum_{\sigma:V\to\{0,1\}} w(\sigma)$$

where $w(\sigma) = \beta^{m(\sigma)}\lambda^{c_1(\sigma)}$, $m(\sigma)$ is the number of monochromatic edges under σ, and $c_1(\sigma)$ is the number of "blue" vertices.

Theorem

Let β and λ be two roots of unity. Then the following holds:

- If $\beta = \pm 1$, or $\beta = \pm\mathtt{i}$ and $\lambda \in \{1, -1, \mathtt{i}, -\mathtt{i}\}$, $Z_G(\beta;\lambda)$ can be computed exactly in polynomial time.
- Otherwise $|Z_G(\beta;\lambda)|$ is #**P**-hard to approximate.

Hardness results of approximating $\arg(Z_G)$:

- Given an oracle computing the sign of Tutte polynomial at $(-e^{2\pi i/5}, -e^{8\pi i/5})$ over planar graphs, all problems in **BQP** can be solved classically in polynomial time.

  [Bordewich, Freedman, Lovász, Welsh 05]

- To determine this sign is #**P**-hard over general graphs.

  [Goldberg, G. 14]

Open Questions

Antiferro 2-spin systems:

- Approximation complexity at the threshold.

Open Questions

Antiferro 2-spin systems:

- Approximation complexity at the threshold.

Ferro 2-spin systems:

- FPTAS for $1 < \beta \leqslant \gamma$, $\lambda_v < \lambda_c$?
  - Conditional spatial mixing for graphs instead of trees.
- Avoiding the gadget gap in the hardness proof.

Open Questions

Antiferro 2-spin systems:

- Approximation complexity at the threshold.

Ferro 2-spin systems:

- FPTAS for $1 < \beta \leqslant \gamma$, $\lambda_v < \lambda_c$?
  - Conditional spatial mixing for graphs instead of trees.
- Avoiding the gadget gap in the hardness proof.

Thank You!