The Classification Program I: FKT, Matchgates, and Holographic Algorithms

Jin-Yi Cai
(University of Wisconsin-Madison)

January 25, 2016

Overview

- Introduction to the Classification Program of Counting Problems.

Overview

- Introduction to the Classification Program of Counting Problems.
- When is a problem easy, when is it hard?

Overview

- Introduction to the Classification Program of Counting Problems.
- When is a problem easy, when is it hard?
- Dichotomy Theorems.

Overview

- Introduction to the Classification Program of Counting Problems.
- When is a problem easy, when is it hard?
- Dichotomy Theorems.
- Three frameworks to address this question:

Overview

- Introduction to the Classification Program of Counting Problems.
- When is a problem easy, when is it hard?
- Dichotomy Theorems.
- Three frameworks to address this question:
  1. Spin Systems/Graph Homomorphisms

Overview

- Introduction to the Classification Program of Counting Problems.
- When is a problem easy, when is it hard?
- Dichotomy Theorems.
- Three frameworks to address this question:
    1. Spin Systems/Graph Homomorphisms
    2. #CSP

Overview

- Introduction to the Classification Program of Counting Problems.
- When is a problem easy, when is it hard?
- Dichotomy Theorems.
- Three frameworks to address this question:
  1. Spin Systems/Graph Homomorphisms
  2. #CSP
  3. Holant

Overview

- Introduction to the Classification Program of Counting Problems.
- When is a problem easy, when is it hard?
- Dichotomy Theorems.
- Three frameworks to address this question:
  1. Spin Systems/Graph Homomorphisms
  2. #CSP
  3. Holant
- Holographic Reductions. Holographic Algorithms. Matchgates.

Overview

- Introduction to the Classification Program of Counting Problems.
- When is a problem easy, when is it hard?
- Dichotomy Theorems.
- Three frameworks to address this question:
  1. Spin Systems/Graph Homomorphisms
  2. #CSP
  3. Holant
- Holographic Reductions. Holographic Algorithms. Matchgates.

`http://www.cs.wisc.edu/~jyc/dichotomy-book.pdf`

Counting Problems

Valiant defined the class #P, and established the first #P-completeness results.

Counting Problems

Valiant defined the class #P, and established the first #P-completeness results.

We would like to classify broad classes of counting problems.

Counting Problems

Valiant defined the class #P, and established the first #P-completeness results.

We would like to classify broad classes of counting problems.

Some counting problems are #P-complete even though their corresponding decision problems are in P. e.g., 2SAT, Perfect Matchings.

Counting Problems

Valiant defined the class #P, and established the first #P-completeness results.

We would like to classify broad classes of counting problems.

Some counting problems are #P-complete even though their corresponding decision problems are in P. e.g., 2SAT, Perfect Matchings.

Some counting problems are #P-hard for general graphs, but in P-time for planar graphs.

Valiant defined the class #P, and established the first #P-completeness results.

We would like to classify broad classes of counting problems.

Some counting problems are #P-complete even though their corresponding decision problems are in P. e.g., 2SAT, Perfect Matchings.

Some counting problems are #P-hard for general graphs, but in P-time for planar graphs.

Counting PM over planar graphs is in P.
This is known as the FKT Algorithm (Fisher, Kasteleyn, and Temperley).

Pfaffian

The fact that Pfaffians are more fundamental than determinants, ...

—Donald Knuth: "Overlapping Pfaffians"

Pfaffian

> *The fact that Pfaffians are more fundamental than determinants, ...*
>
> —*Donald Knuth: "Overlapping Pfaffians"*

An $n \times n$ matrix A is called skew-symmetric if $A_{i,j} = -A_{j,i}$, for $1 \leq i, j \leq n$.

The fact that Pfaffians are more fundamental than determinants,
...

—Donald Knuth: "Overlapping Pfaffians"

An $n \times n$ matrix A is called skew-symmetric if $A_{i,j} = -A_{j,i}$, for $1 \leq i, j \leq n$.

E.g.,

$$A = \begin{bmatrix} 0 & x_{12} & x_{13} & x_{14} \\ -x_{12} & 0 & x_{23} & x_{24} \\ -x_{13} & -x_{23} & 0 & x_{34} \\ -x_{14} & -x_{24} & -x_{34} & 0 \end{bmatrix}.$$

> *The fact that Pfaffians are more fundamental than determinants,*
> . . .
>
> *—Donald Knuth: "Overlapping Pfaffians"*

An $n \times n$ matrix A is called skew-symmetric if $A_{i,j} = -A_{j,i}$, for $1 \le i, j \le n$.

E.g.,

$$A = \begin{bmatrix} 0 & x_{12} & x_{13} & x_{14} \\ -x_{12} & 0 & x_{23} & x_{24} \\ -x_{13} & -x_{23} & 0 & x_{34} \\ -x_{14} & -x_{24} & -x_{34} & 0 \end{bmatrix}.$$

Then there is a matrix function called the Pfaffian.

$$\mathrm{Pf}(A) = x_{12}x_{34} - x_{13}x_{24} + x_{14}x_{23}.$$

Pfaffian

The Pfaffian of an $n \times n$ skew-symmetric matrix A is defined as follows. Suppose $n = 2k \geq 2$ is even, then

$$\mathrm{Pf}(A) = \sum_{\pi} \mathrm{sign}(\pi) A_{i_1,i_2} A_{i_3,i_4} \cdots A_{i_{2k-1},i_{2k}} \tag{1}$$

where the sum is over all permutations $\pi = \begin{pmatrix} 1 & 2 & \cdots & 2k \\ i_1 & i_2 & \cdots & i_{2k} \end{pmatrix}$ such that,

$$i_1 < i_2,\ i_3 < i_4,\ \ldots,\ i_{2k-1} < i_{2k} \quad \text{and} \quad i_1 < i_3 < \ldots < i_{2k-1}. \tag{2}$$

The value $\mathrm{sign}(\pi)$ in (1) denotes the parity of π; it is $+1$ or -1 depending on whether π is an even or odd permutation, respectively.

There is a natural 1-1 correspondence between the terms in the Pfaffian expression, or equivalently, permutations π satisfying the stipulation (2) and partitions of $[n]$ into disjoint pairs

$$\tilde{\pi} = \{\{i_1, i_2\}, \{i_3, i_4\}, \ldots, \{i_{2k-1}, i_{2k}\}\}.$$

There is a natural 1-1 correspondence between the terms in the Pfaffian expression, or equivalently, permutations π satisfying the stipulation (2) and partitions of $[n]$ into disjoint pairs

$$\tilde{\pi} = \{\{i_1, i_2\}, \{i_3, i_4\}, \ldots, \{i_{2k-1}, i_{2k}\}\}.$$

For any permutation $\pi = \begin{pmatrix} 1 & 2 & \ldots & 2k \\ i_1 & i_2 & \ldots & i_{2k} \end{pmatrix}$, not necessarily satisfying the stipulation (2), define

$$a_\pi = \mathrm{sign}(\pi) A_{i_1,i_2} A_{i_3,i_4} \cdots A_{i_{2k-1},i_{2k}}.$$

Observation: If $\tilde{\pi} = \tilde{\pi}'$, then $a_\pi = a_{\pi'}$, i.e., the expression a_π has the same value if we list the partition $\tilde{\pi} = \{\{i_1, i_2\}, \{i_3, i_4\}, \ldots, \{i_{2k-1}, i_{2k}\}\}$ in any order of the pairs, as well as in any order of the two labels of each pair.

Crossover

We say two pairs of labels $i_{2j-1} < i_{2j}$ and $i_{2\ell-1} < i_{2\ell}$ form a crossover, or an overlapping pair, iff

$$i_{2j-1} < i_{2\ell-1} < i_{2j} < i_{2\ell} \quad \text{or} \quad i_{2\ell-1} < i_{2j-1} < i_{2\ell} < i_{2j}.$$

We say two pairs of labels $i_{2j-1} < i_{2j}$ and $i_{2\ell-1} < i_{2\ell}$ form a crossover, or an overlapping pair, iff

$$i_{2j-1} < i_{2\ell-1} < i_{2j} < i_{2\ell} \quad \text{or} \quad i_{2\ell-1} < i_{2j-1} < i_{2\ell} < i_{2j}.$$

Suppose π satisfies stipulation (2).

We say two pairs of labels $i_{2j-1} < i_{2j}$ and $i_{2\ell-1} < i_{2\ell}$ form a crossover, or an overlapping pair, iff

$$i_{2j-1} < i_{2\ell-1} < i_{2j} < i_{2\ell} \quad \text{or} \quad i_{2\ell-1} < i_{2j-1} < i_{2\ell} < i_{2j}.$$

Suppose π satisfies stipulation (2).

Let $c(\pi)$ be the number of crossovers among the pairs in the partition $\tilde{\pi}$. Then

$$\mathrm{sign}(\pi) = (-1)^{c(\pi)}.$$

Crossover and Parity

To see that $\mathrm{sign}(\pi) = (-1)^{c(\pi)}$, consider any permutation π and consider a sequence of adjacent transpositions which moves the sequence

$$(1, 2, \ldots, 2k-1, 2k) \longrightarrow (i_1, i_2, \ldots, i_{2k-1}, i_{2k}).$$

To see that $\mathrm{sign}(\pi) = (-1)^{c(\pi)}$, consider any permutation π and consider a sequence of adjacent transpositions which moves the sequence

$$(1, 2, \ldots, 2k-1, 2k) \longrightarrow (i_1, i_2, \ldots, i_{2k-1}, i_{2k}).$$

We have $i_1 = 1$ by (2). The number of transpositions that will bring i_2 to the position right after 1 is the number of labels strictly between $i_1 = 1$ and the number i_2, and has the same parity as the number of crossovers the pair $\{i_1, i_2\}$ forms with all other pairs $\{\{i_3, i_4\}, \ldots, \{i_{2k-1}, i_{2k}\}\}$ in $\tilde{\pi}$.

Crossover and Parity

To see that $\mathrm{sign}(\pi) = (-1)^{c(\pi)}$, consider any permutation π and consider a sequence of adjacent transpositions which moves the sequence

$$(1, 2, \ldots, 2k-1, 2k) \longrightarrow (i_1, i_2, \ldots, i_{2k-1}, i_{2k}).$$

We have $i_1 = 1$ by (2). The number of transpositions that will bring i_2 to the position right after 1 is the number of labels strictly between $i_1 = 1$ and the number i_2, and has the same parity as the number of crossovers the pair $\{i_1, i_2\}$ forms with all other pairs $\{\{i_3, i_4\}, \ldots, \{i_{2k-1}, i_{2k}\}\}$ in $\tilde{\pi}$.

After i_1, i_2 are placed in the first two positions, if $n > 2$, then i_3 is the minimum among all other labels by (2), and is currently located right after the first two elements. Then we move i_4 to the position right after i_3. The proof is completed by induction.

Pfaffian and Determinant

The Pfaffian can be computed in polynomial time. A key relation to determinant is the following theorem.

Theorem

For any $n \times n$ skew-symmetric matrix A,

$$\det(A) = [\mathrm{Pf}(A)]^2.$$

Pfaffian as a Polynomial

Let $G = (V, E)$ be a simple undirected graph without self loops and parallel edges. Suppose the vertices of G are labeled by a totally ordered set, e.g, $V = \{1, 2, \ldots, n\}$.

Pfaffian as a Polynomial

Let $G = (V, E)$ be a simple undirected graph without self loops and parallel edges. Suppose the vertices of G are labeled by a totally ordered set, e.g, $V = \{1, 2, \ldots, n\}$.

Assign an indeterminate x_e for every edge $e = \{u, v\} \in E$. Then we define the skew-symmetric adjacency matrix $A = A(G)$ of the graph G to be

$$A_{u,v} = \begin{cases} x_e & \text{if } e = \{u, v\} \in E \text{ and } u < v \\ -x_e & \text{if } e = \{u, v\} \in E \text{ and } u > v \\ 0 & \text{if } \{u, v\} \notin E \end{cases} \tag{3}$$

Pfaffian as a Polynomial

Let $G = (V, E)$ be a simple undirected graph without self loops and parallel edges. Suppose the vertices of G are labeled by a totally ordered set, e.g, $V = \{1, 2, \ldots, n\}$.

Assign an indeterminate x_e for every edge $e = \{u, v\} \in E$. Then we define the skew-symmetric adjacency matrix $A = A(G)$ of the graph G to be

$$A_{u,v} = \begin{cases} x_e & \text{if } e = \{u, v\} \in E \text{ and } u < v \\ -x_e & \text{if } e = \{u, v\} \in E \text{ and } u > v \\ 0 & \text{if } \{u, v\} \notin E \end{cases} \tag{3}$$

Denote by $\mathcal{M}(G)$ the set of all perfect matchings of G.

Pfaffian as a Polynomial

Let $G = (V, E)$ be a simple undirected graph without self loops and parallel edges. Suppose the vertices of G are labeled by a totally ordered set, e.g, $V = \{1, 2, \ldots, n\}$.

Assign an indeterminate x_e for every edge $e = \{u, v\} \in E$. Then we define the skew-symmetric adjacency matrix $A = A(G)$ of the graph G to be

$$A_{u,v} = \begin{cases} x_e & \text{if } e = \{u, v\} \in E \text{ and } u < v \\ -x_e & \text{if } e = \{u, v\} \in E \text{ and } u > v \\ 0 & \text{if } \{u, v\} \notin E \end{cases} \tag{3}$$

Denote by $\mathcal{M}(G)$ the set of all perfect matchings of G.

For any permutation π, the partition $\tilde{\pi}$ is a perfect matching iff all pairs are edges.

Weight of a Perfect Matching

There is a 1-1 correspondence between non-zero terms in $\mathrm{Pf}(A)$ and $\mathcal{M}(G)$.

Weight of a Perfect Matching

There is a 1-1 correspondence between non-zero terms in $\mathrm{Pf}(A)$ and $\mathcal{M}(G)$.

For any $M \in \mathcal{M}(G)$ there are $2^k k!$ permutations of the form $\pi' = \begin{pmatrix} 1 & 2 & \dots & 2k \\ i_1 & i_2 & \dots & i_{2k} \end{pmatrix}$ that can represent M, i.e., $\tilde{\pi}' = M$.

Weight of a Perfect Matching

There is a 1-1 correspondence between non-zero terms in $\mathrm{Pf}(A)$ and $\mathcal{M}(G)$.

For any $M \in \mathcal{M}(G)$ there are $2^k k!$ permutations of the form $\pi' = \begin{pmatrix} 1 & 2 & \dots & 2k \\ i_1 & i_2 & \dots & i_{2k} \end{pmatrix}$ that can represent M, i.e., $\tilde{\pi}' = M$.

We define the weight of a perfect matching M in G to be

$$\Gamma(M) = \Gamma_G(M) = \prod_{e \in M} x_e .$$

Orientation

An orientation of a graph G assigns one direction to each edge of G.

Orientation

An orientation of a graph G assigns one direction to each edge of G.

We denote by $u \to v$ if the edge $\{u, v\}$ is oriented from u to v.
We say u is its tail, and v is its head.

An orientation of a graph G assigns one direction to each edge of G.

We denote by $u \to v$ if the edge $\{u, v\}$ is oriented from u to v.
We say u is its tail, and v is its head.

For an oriented graph $\overrightarrow{G}$ we modify the skew-symmetric matrix A to be $B = B(\overrightarrow{G})$:

$$B_{u,v} = \begin{cases} x_e & \text{if } e = \{u, v\} \in E \text{ and } u \to v \\ -x_e & \text{if } e = \{u, v\} \in E \text{ and } v \to u \\ 0 & \text{if } \{u, v\} \notin E \end{cases} \tag{4}$$

Orientation

An orientation of a graph G assigns one direction to each edge of G.

We denote by $u \to v$ if the edge $\{u, v\}$ is oriented from u to v.
We say u is its tail, and v is its head.

For an oriented graph $\overrightarrow{G}$ we modify the skew-symmetric matrix A to be $B = B(\overrightarrow{G})$:

$$B_{u,v} = \begin{cases} x_e & \text{if } e = \{u, v\} \in E \text{ and } u \to v \\ -x_e & \text{if } e = \{u, v\} \in E \text{ and } v \to u \\ 0 & \text{if } \{u, v\} \notin E \end{cases} \tag{4}$$

In other words, we change the sign at both entries $A_{u,v}$ and $A_{v,u}$ provided

$$u < v \text{ and } \{u, v\} \text{ is oriented } v \to u.$$

Pfaffian Term

Given an orientation, we can consider the Pfaffian of B

$$\mathrm{Pf}(B) = \sum_{\pi} \mathrm{sign}(\pi) B_{i_1,i_2} B_{i_3,i_4} \cdots B_{i_{2k-1},i_{2k}}. \tag{5}$$

Pfaffian Term

Given an orientation, we can consider the Pfaffian of B

$$\mathrm{Pf}(B) = \sum_{\pi} \mathrm{sign}(\pi) B_{i_1,i_2} B_{i_3,i_4} \cdots B_{i_{2k-1},i_{2k}}. \tag{5}$$

For a perfect matching M in an oriented graph $\overrightarrow{G}$, suppose $M = \tilde{\pi}$, define the Pfaffian term

$$\mathrm{Pf}_{\overrightarrow{G}}(M) = \mathrm{sign}(\pi) B_{i_1,i_2} B_{i_3,i_4} \cdots B_{i_{2k-1},i_{2k}}. \tag{6}$$

Pfaffian Term

Given an orientation, we can consider the Pfaffian of B

$$\mathrm{Pf}(B) = \sum_{\pi} \mathrm{sign}(\pi) B_{i_1,i_2} B_{i_3,i_4} \cdots B_{i_{2k-1},i_{2k}}. \tag{5}$$

For a perfect matching M in an oriented graph $\overrightarrow{G}$, suppose $M = \tilde{\pi}$, define the Pfaffian term

$$\mathrm{Pf}_{\overrightarrow{G}}(M) = \mathrm{sign}(\pi) B_{i_1,i_2} B_{i_3,i_4} \cdots B_{i_{2k-1},i_{2k}}. \tag{6}$$

We can choose any permutation π representing M, and the value $\mathrm{Pf}_{\overrightarrow{G}}(M)$ is invariant; it only depends on M.

Pfaffian Term

Given an orientation, we can consider the Pfaffian of B

$$\mathrm{Pf}(B) = \sum_{\pi} \mathrm{sign}(\pi) B_{i_1,i_2} B_{i_3,i_4} \cdots B_{i_{2k-1},i_{2k}}. \tag{5}$$

For a perfect matching M in an oriented graph $\overrightarrow{G}$, suppose $M = \tilde{\pi}$, define the Pfaffian term

$$\mathrm{Pf}_{\overrightarrow{G}}(M) = \mathrm{sign}(\pi) B_{i_1,i_2} B_{i_3,i_4} \cdots B_{i_{2k-1},i_{2k}}. \tag{6}$$

We can choose any permutation π representing M, and the value $\mathrm{Pf}_{\overrightarrow{G}}(M)$ is invariant; it only depends on M.

It is a term in $\mathrm{Pf}(B)$ when π is the canonical expression for M and it is equal to either $\Gamma_G(M)$ or its negation $-\Gamma_G(M)$.

Sign of a Perfect Matching

Definition

For any perfect matching M in an oriented graph $\overrightarrow{G}$, the *sign* of the perfect matching M with respect to this orientation is

$$\operatorname{sgn}(M) = \frac{\mathrm{Pf}_{\overrightarrow{G}}(M)}{\Gamma_G(M)} \in \{-1, 1\}. \tag{7}$$

Sign of a Perfect Matching

Definition

For any perfect matching M in an oriented graph $\overrightarrow{G}$, the *sign* of the perfect matching M with respect to this orientation is

$$\operatorname{sgn}(M) = \frac{\operatorname{Pf}_{\overrightarrow{G}}(M)}{\Gamma_G(M)} \in \{-1, 1\}. \tag{7}$$

We note that $\operatorname{sgn}(M)$ can be computed with any permutation π with partition $\tilde{\pi} = M$, due to the invariance of $\operatorname{Pf}_{\overrightarrow{G}}(M)$.

Sign of a Perfect Matching

Definition

For any perfect matching M in an oriented graph $\overrightarrow{G}$, the *sign* of the perfect matching M with respect to this orientation is

$$\operatorname{sgn}(M) = \frac{\operatorname{Pf}_{\overrightarrow{G}}(M)}{\Gamma_G(M)} \in \{-1, 1\}. \tag{7}$$

We note that $\operatorname{sgn}(M)$ can be computed with any permutation π with partition $\tilde{\pi} = M$, due to the invariance of $\operatorname{Pf}_{\overrightarrow{G}}(M)$.

In particular, $\operatorname{sgn}(M)$ can be computed simply as the sign of the permutation $\pi = \begin{pmatrix} 1 & 2 & \dots & 2k \\ i_1 & i_2 & \dots & i_{2k} \end{pmatrix}$ where each matching edge $\{i_{2\ell-1}, i_{2\ell}\} \in M$ is listed by its orientation $i_{2\ell-1} \to i_{2\ell}$.

If We Have Two Perfect Matchings

Let M and M' be two perfect matchings.

If We Have Two Perfect Matchings

Let M and M' be two perfect matchings.

The symmetric difference $M \oplus M'$ consists of a set of cycles of even length that are alternating between edges from M and M'.

If We Have Two Perfect Matchings

Let M and M' be two perfect matchings.

The symmetric difference $M \oplus M'$ consists of a set of cycles of even length that are alternating between edges from M and M'.

For any orientation on a graph G, if C is a cycle of even length, then we say it is evenly oriented if there are an even number of edges oriented in one direction, and oddly oriented otherwise.

Let M and M' be two perfect matchings.

The symmetric difference $M \oplus M'$ consists of a set of cycles of even length that are alternating between edges from M and M'.

For any orientation on a graph G, if C is a cycle of even length, then we say it is evenly oriented if there are an even number of edges oriented in one direction, and oddly oriented otherwise.

Clearly this notion does not depend on the direction since C has an even length.

A Key Lemma

Lemma

For and perfect matchings M and M' in an oriented graph G, if k is the number of evenly oriented cycles in $M \oplus M'$, then

$$\mathrm{sgn}(M) \cdot \mathrm{sgn}(M') = (-1)^k.$$

A Key Lemma

Lemma

For and perfect matchings M and M' in an oriented graph G, if k is the number of evenly oriented cycles in $M \oplus M'$, then

$$\mathrm{sgn}(M) \cdot \mathrm{sgn}(M') = (-1)^k.$$

Proof Sketch:

A Key Lemma

Lemma

For and perfect matchings M and M' in an oriented graph G, if k is the number of evenly oriented cycles in $M \oplus M'$, then

$$\mathrm{sgn}(M) \cdot \mathrm{sgn}(M') = (-1)^k.$$

Proof Sketch:

1. If equality holds for one orientation then it holds for all orientations.

A Key Lemma

Lemma

For and perfect matchings M and M' in an oriented graph G, if k is the number of evenly oriented cycles in $M \oplus M'$, then

$$\operatorname{sgn}(M) \cdot \operatorname{sgn}(M') = (-1)^k.$$

Proof Sketch:

1. If equality holds for one orientation then it holds for all orientations.
2. Choose one orientation such that it gives a cyclic orientation on each cycle of $M \oplus M'$. Orient all other edges arbitrarily. Then k is the number of cycles in $M \oplus M'$.

A Key Lemma

Lemma

For and perfect matchings M and M' in an oriented graph G, if k is the number of evenly oriented cycles in $M \oplus M'$, then

$$\mathrm{sgn}(M) \cdot \mathrm{sgn}(M') = (-1)^k.$$

Proof Sketch:

1. If equality holds for one orientation then it holds for all orientations.
2. Choose one orientation such that it gives a cyclic orientation on each cycle of $M \oplus M'$. Orient all other edges arbitrarily. Then k is the number of cycles in $M \oplus M'$.
3. We may relabel the vertices of G.

A Key Lemma

Lemma

For and perfect matchings M and M' in an oriented graph G, if k is the number of evenly oriented cycles in $M \oplus M'$, then

$$\mathrm{sgn}(M) \cdot \mathrm{sgn}(M') = (-1)^k.$$

Proof Sketch:

1. If equality holds for one orientation then it holds for all orientations.
2. Choose one orientation such that it gives a cyclic orientation on each cycle of $M \oplus M'$. Orient all other edges arbitrarily. Then k is the number of cycles in $M \oplus M'$.
3. We may relabel the vertices of G.
4. Sequentially label each cycle of $M \oplus M'$ starting at the tail of an edge in M.

Lemma

For and perfect matchings M and M' in an oriented graph G, if k is the number of evenly oriented cycles in $M \oplus M'$, then

$$\mathrm{sgn}(M) \cdot \mathrm{sgn}(M') = (-1)^k.$$

Proof Sketch:

1. If equality holds for one orientation then it holds for all orientations.
2. Choose one orientation such that it gives a cyclic orientation on each cycle of $M \oplus M'$. Orient all other edges arbitrarily. Then k is the number of cycles in $M \oplus M'$.
3. We may relabel the vertices of G.
4. Sequentially label each cycle of $M \oplus M'$ starting at the tail of an edge in M.
5. After all cycles of $M \oplus M'$ are done, label all remaining vertices so that each edge in $M \cap M'$ is labeled consecutively with the next unused integers, and increasing from tail to head.

Key Lemma Continued

Now the Pfaffian term $\mathrm{Pf}_{\vec{G}}(M)$ for M is just the product of all edge weights $\Gamma_G(M)$. This is seen easily if we write b_π in the way where the permutation π is the identity, and we list all matched edges in the oriented order.

Key Lemma Continued

Now the Pfaffian term $\mathrm{Pf}_{\vec{G}}(M)$ for M is just the product of all edge weights $\Gamma_G(M)$. This is seen easily if we write b_π in the way where the permutation π is the identity, and we list all matched edges in the oriented order.

For the Pfaffian term $b_{\pi'}$ corresponding to M', we still list the product part $B_{i_1,i_2}B_{i_3,i_4}\cdots B_{i_{2k-1},i_{2k}}$ in the oriented order for each matched edge. The $\mathrm{sign}(\pi')$ for the permutation is as follows:

Now the Pfaffian term $\mathrm{Pf}_{\vec{G}}(M)$ for M is just the product of all edge weights $\Gamma_G(M)$. This is seen easily if we write b_π in the way where the permutation π is the identity, and we list all matched edges in the oriented order.

For the Pfaffian term $b_{\pi'}$ corresponding to M', we still list the product part $B_{i_1,i_2}B_{i_3,i_4}\cdots B_{i_{2k-1},i_{2k}}$ in the oriented order for each matched edge. The $\mathrm{sign}(\pi')$ for the permutation is as follows:

The part for the first cycle of length 2ℓ in $M \oplus M'$ has the form $\begin{pmatrix} 1 & 2 & \ldots & 2\ell-1 & 2\ell \\ 2 & 3 & \ldots & 2\ell & 1 \end{pmatrix}$, which is an even cycle as a permutation, and has an odd parity.

Now the Pfaffian term $\mathrm{Pf}_{\vec{G}}(M)$ for M is just the product of all edge weights $\Gamma_G(M)$. This is seen easily if we write b_π in the way where the permutation π is the identity, and we list all matched edges in the oriented order.

For the Pfaffian term $b_{\pi'}$ corresponding to M', we still list the product part $B_{i_1,i_2}B_{i_3,i_4}\cdots B_{i_{2k-1},i_{2k}}$ in the oriented order for each matched edge. The $\mathrm{sign}(\pi')$ for the permutation is as follows:

The part for the first cycle of length 2ℓ in $M \oplus M'$ has the form $\begin{pmatrix} 1 & 2 & \dots & 2\ell-1 & 2\ell \\ 2 & 3 & \dots & 2\ell & 1 \end{pmatrix}$, which is an even cycle as a permutation, and has an odd parity.

The permutation π' is a product of these disjoint cycles in the permutation group. Hence it has parity $\mathrm{sign}(\pi') = (-1)^k$.

Pfaffian Orientation

Definition

An orientation of a connected plane graph is called a Pfaffian orientation if along the boundary of every non-outer face, there are an odd number of clockwise oriented edges.

Definition

An orientation of a connected plane graph is called a Pfaffian orientation if along the boundary of every non-outer face, there are an odd number of clockwise oriented edges.

We have the following Pfaffian Orientation Lemma

Lemma

Any Pfaffian orientation in a connected plane graph G satisfies the following property: For every cycle C, the number of clockwise oriented edges of C is of the opposite parity to the number of vertices contained within C. (This number does not include the vertices on the cycle C).

Pfaffian Orientation Continued

Proof: Let V and E be the number of vertices and edges contained within C, and let ℓ be the number of edges on C, which is also the number of vertices on C.

Pfaffian Orientation Continued

Proof: Let V and E be the number of vertices and edges contained within C, and let ℓ be the number of edges on C, which is also the number of vertices on C.

The vertices contained within C are those in the interior of the region bounded by C; they do not include those on the cycle C. Similarly the edges within C do not include those on C.

Pfaffian Orientation Continued

Proof: Let V and E be the number of vertices and edges contained within C, and let ℓ be the number of edges on C, which is also the number of vertices on C.

The vertices contained within C are those in the interior of the region bounded by C; they do not include those on the cycle C. Similarly the edges within C do not include those on C.

Suppose there are F faces bounded by C, and let c_i be the number of clockwise oriented edges on the boundary of the i-th face $(1 \leq i \leq \mathrm{F})$. Each c_i is odd by assumption, therefore $\mathrm{F} \equiv \sum_{i=1}^{\mathrm{F}} c_i \pmod 2$.

Pfaffian Orientation Continued

Proof: Let V and E be the number of vertices and edges contained within C, and let ℓ be the number of edges on C, which is also the number of vertices on C.

The vertices contained within C are those in the interior of the region bounded by C; they do not include those on the cycle C. Similarly the edges within C do not include those on C.

Suppose there are F faces bounded by C, and let c_i be the number of clockwise oriented edges on the boundary of the i-th face $(1 \leq i \leq \mathrm{F})$. Each c_i is odd by assumption, therefore $\mathrm{F} \equiv \sum_{i=1}^{\mathrm{F}} c_i \pmod 2$.

By Euler's formula, counting the face formed by the exterior of C, we have $(\mathrm{V} + \ell) - (\mathrm{E} + \ell) + (\mathrm{F} + 1) = 2$. It follows that

$$\mathrm{E} = \mathrm{V} + \mathrm{F} - 1.$$

Pfaffian Orientation Continued

We have

$$E = V + F - 1$$

If we add up all the clockwise oriented edges among all boundary edges in F faces, each interior edge within C regardless orientation contributes one and each clockwise oriented edge on C contributes one. Hence $\sum_{i=1}^{F} c_i = E + c$, where c is the number of clockwise oriented edges on C.

We have

$$\mathsf{E} = \mathsf{V} + \mathsf{F} - 1$$

If we add up all the clockwise oriented edges among all boundary edges in F faces, each interior edge within C regardless orientation contributes one and each clockwise oriented edge on C contributes one. Hence $\sum_{i=1}^{\mathsf{F}} c_i = \mathsf{E} + c$, where c is the number of clockwise oriented edges on C.

It follows that

$$\mathsf{F} \equiv \sum_{i=1}^{\mathsf{F}} c_i = \mathsf{E} + c = \mathsf{V} + \mathsf{F} - 1 + c \pmod 2,$$

and hence $\mathsf{V} + c = 1 \pmod 2$.

For every cycle of $M \oplus M'$, any inside vertex cannot match any outside vertex. Hence the cycle contains a perfect matching (possibly empty) in its interior.

For every cycle of $M \oplus M'$, any inside vertex cannot match any outside vertex. Hence the cycle contains a perfect matching (possibly empty) in its interior.

So the number of vertices within the cycle must be even.

For every cycle of $M \oplus M'$, any inside vertex cannot match any outside vertex. Hence the cycle contains a perfect matching (possibly empty) in its interior.

So the number of vertices within the cycle must be even.

Each cycle of $M \oplus M'$ has an even length, consisting of alternatingly edges from M and M'.

For every cycle of $M \oplus M'$, any inside vertex cannot match any outside vertex. Hence the cycle contains a perfect matching (possibly empty) in its interior.

So the number of vertices within the cycle must be even.

Each cycle of $M \oplus M'$ has an even length, consisting of alternatingly edges from M and M'.

By the Pfaffian Orientation Lemma just proved, the cycle is oddly oriented.

For every cycle of $M \oplus M'$, any inside vertex cannot match any outside vertex. Hence the cycle contains a perfect matching (possibly empty) in its interior.

So the number of vertices within the cycle must be even.

Each cycle of $M \oplus M'$ has an even length, consisting of alternatingly edges from M and M'.

By the Pfaffian Orientation Lemma just proved, the cycle is oddly oriented.

By the Key Lemma proved earlier, $\mathrm{sgn}(M) \cdot \mathrm{sgn}(M') = 1$.

For every cycle of $M \oplus M'$, any inside vertex cannot match any outside vertex. Hence the cycle contains a perfect matching (possibly empty) in its interior.

So the number of vertices within the cycle must be even.

Each cycle of $M \oplus M'$ has an even length, consisting of alternatingly edges from M and M'.

By the Pfaffian Orientation Lemma just proved, the cycle is oddly oriented.

By the Key Lemma proved earlier, $\mathrm{sgn}(M) \cdot \mathrm{sgn}(M') = 1$.

Hence, with respect to a Pfaffian orientation every two perfect matchings M and M' must have the same sign: $\mathrm{sgn}(M) = \mathrm{sgn}(M')$.

Hence for a Pfaffian orientation, every Pfaffian term has the same sign.

Hence for a Pfaffian orientation, every Pfaffian term has the same sign.

Definition

The perfect matching polynomial is the following:

$$\mathrm{PerfMatch}(G) = \sum_{M \in \mathcal{M}(G)} \prod_{e \in M} x_e \tag{8}$$

PerfMatch Polynomial

Hence for a Pfaffian orientation, every Pfaffian term has the same sign.

Definition

The perfect matching polynomial is the following:

$$\mathrm{PerfMatch}(G) = \sum_{M \in \mathcal{M}(G)} \prod_{e \in M} x_e \tag{8}$$

If we assign $x_e = 1$, then $\mathrm{PerfMatch}(G)$ counts the number of perfect matchings in G.

Kasteleyn's Algorithm a.k.a. FKT

Theorem (Kasteleyn)

Every connected planar graph has a Pfaffian orientation. Such an orientation can be constructed in polynomial time, leading to a polynomial time algorithm to compute $\mathrm{PerfMatch}(G)$ *for any weighted planar graph* G.

Kasteleyn's Algorithm a.k.a. FKT

Theorem (Kasteleyn)

Every connected planar graph has a Pfaffian orientation. Such an orientation can be constructed in polynomial time, leading to a polynomial time algorithm to compute $\mathrm{PerfMatch}(G)$ *for any weighted planar graph* G.

Intuitively, one can "grow" from inside out, face by face, to extend to a Pfaffian orientation.

Theorem (Kasteleyn)

Every connected planar graph has a Pfaffian orientation. Such an orientation can be constructed in polynomial time, leading to a polynomial time algorithm to compute $\mathrm{PerfMatch}(G)$ *for any weighted planar graph* G*.*

Intuitively, one can "grow" from inside out, face by face, to extend to a Pfaffian orientation.

This is technically not quite right ...

Kasteleyn's Algorithm

Assume G is a connected planar graph, and given a planar embedding.

Kasteleyn's Algorithm

Assume G is a connected planar graph, and given a planar embedding.

If G is a tree, then any orientation is acceptable.

Kasteleyn's Algorithm

Assume G is a connected planar graph, and given a planar embedding.

If G is a tree, then any orientation is acceptable.

If G is not a tree, then choose any edge on the boundary of the outer face which belongs to a cycle.

Kasteleyn's Algorithm

Assume G is a connected planar graph, and given a planar embedding.

If G is a tree, then any orientation is acceptable.

If G is not a tree, then choose any edge on the boundary of the outer face which belongs to a cycle.

Such an edge exists by a simple induction.

Assume G is a connected planar graph, and given a planar embedding.

If G is a tree, then any orientation is acceptable.

If G is not a tree, then choose any edge on the boundary of the outer face which belongs to a cycle.

Such an edge exists by a simple induction.

Let F be the non-outer face containing this edge e. By induction we can construct a Pfaffian orientation for $G - \{e\}$, the graph with the same vertex set as G but with edge e removed.

Kasteleyn's Algorithm

Assume G is a connected planar graph, and given a planar embedding.

If G is a tree, then any orientation is acceptable.

If G is not a tree, then choose any edge on the boundary of the outer face which belongs to a cycle.

Such an edge exists by a simple induction.

Let F be the non-outer face containing this edge e. By induction we can construct a Pfaffian orientation for $G - \{e\}$, the graph with the same vertex set as G but with edge e removed.

Now add e back, and orient e appropriately we can guarantee that F also has an odd number of clockwise oriented edges.

For a Pfaffian orientation, let $B = B(\overrightarrow{G})$ be the skew-symmetric matrix. Then either $\mathrm{Pf}(B) = \mathrm{PerfMatch}(G)$ or $\mathrm{Pf}(B) = -\mathrm{PerfMatch}(G)$.

For a Pfaffian orientation, let $B = B(\overrightarrow{G})$ be the skew-symmetric matrix. Then either $\mathrm{Pf}(B) = \mathrm{PerfMatch}(G)$ or $\mathrm{Pf}(B) = -\mathrm{PerfMatch}(G)$.

The equality is a polynomial equality: for the given Pfaffian orientation, either $+$ holds for all weight values, or $-$ holds for all weight values.

Kasteleyn's Algorithm

For a Pfaffian orientation, let $B = B(\overrightarrow{G})$ be the skew-symmetric matrix. Then either $\mathrm{Pf}(B) = \mathrm{PerfMatch}(G)$ or $\mathrm{Pf}(B) = -\mathrm{PerfMatch}(G)$.

The equality is a polynomial equality: for the given Pfaffian orientation, either $+$ holds for all weight values, or $-$ holds for all weight values.

Setting all weight values to 1, we can decide which sign is valid for the particular orientation (unless there is no perfect matching and G is non-empty, in which we can safely output $\mathrm{PerfMatch}(G) = 0$).

For a Pfaffian orientation, let $B = B(\overrightarrow{G})$ be the skew-symmetric matrix. Then either $\mathrm{Pf}(B) = \mathrm{PerfMatch}(G)$ or $\mathrm{Pf}(B) = -\mathrm{PerfMatch}(G)$.

The equality is a polynomial equality: for the given Pfaffian orientation, either $+$ holds for all weight values, or $-$ holds for all weight values.

Setting all weight values to 1, we can decide which sign is valid for the particular orientation (unless there is no perfect matching and G is non-empty, in which we can safely output $\mathrm{PerfMatch}(G) = 0$).

Then we can compute $\mathrm{PerfMatch}(G)$ for the actual weight values.

Kasteleyn's Algorithm

For a Pfaffian orientation, let $B = B(\overrightarrow{G})$ be the skew-symmetric matrix. Then either $\mathrm{Pf}(B) = \mathrm{PerfMatch}(G)$ or $\mathrm{Pf}(B) = -\mathrm{PerfMatch}(G)$.

The equality is a polynomial equality: for the given Pfaffian orientation, either $+$ holds for all weight values, or $-$ holds for all weight values.

Setting all weight values to 1, we can decide which sign is valid for the particular orientation (unless there is no perfect matching and G is non-empty, in which we can safely output $\mathrm{PerfMatch}(G) = 0$).

Then we can compute $\mathrm{PerfMatch}(G)$ for the actual weight values.

If G has connected components $G_1, G_2, \ldots, G_m$, then

$$\mathrm{PerfMatch}(G) = \prod_{i=1}^{m} \mathrm{PerfMatch}(G_i).$$

Definition

A matchgate is an undirected weighted plane graph G with a subset of distinguished nodes on its outer face, called the external nodes, ordered in a clockwise order.

Definition

A matchgate is an undirected weighted plane graph G with a subset of distinguished nodes on its outer face, called the external nodes, ordered in a clockwise order.

Let G be a matchgate with k external nodes. For each $\alpha \in \{0,1\}^k$, G defines a subgraph G^α obtained from G by moving all external nodes i (and incident edges) such that $\alpha_i = 1$.

Definition

A matchgate is an undirected weighted plane graph G with a subset of distinguished nodes on its outer face, called the external nodes, ordered in a clockwise order.

Let G be a matchgate with k external nodes. For each $\alpha \in \{0,1\}^k$, G defines a subgraph G^α obtained from G by moving all external nodes i (and incident edges) such that $\alpha_i = 1$.

Definition

We define the signature of a matchgate G as the vector $\Gamma_G = (\Gamma_G^\alpha)$, indexed by $\alpha \in \{0,1\}^k$ in lexicographic order, as follows:

$$\Gamma_G^\alpha = \mathrm{PerfMatch}(G^\alpha) = \sum_{M \in \mathcal{M}(G^\alpha)} \prod_{e \in M} w(e). \tag{9}$$

Perfect Matchings as a Holant Sum

Counting the number of Perfect Matchings can be viewed as follows:

Counting the number of Perfect Matchings can be viewed as follows:

A graph G is given, where every vertex v is labeled by an EXACT-ONE function f_v of arity $\deg(v)$.

Perfect Matchings as a Holant Sum

Counting the number of Perfect Matchings can be viewed as follows:

A graph G is given, where every vertex v is labeled by an $\textsc{Exact-One}$ function f_v of arity $\deg(v)$.

We then consider

$$\mathrm{Holant}(G) = \sum_{\sigma: E \to \{0,1\}} \prod_{v \in V} f_v(\sigma \mid_{E(v)}).$$

Counting the number of Perfect Matchings can be viewed as follows:

A graph G is given, where every vertex v is labeled by an $\textsc{Exact-One}$ function f_v of arity $\deg(v)$.

We then consider

$$\mathrm{Holant}(G) = \sum_{\sigma : E \to \{0,1\}} \prod_{v \in V} f_v(\sigma \mid_{E(v)}).$$

Each product term gives a one if $\sigma^{-1}(1)$ is a Perfect Matching, and zero otherwise.

Definition

Let $\mathcal{F}$ be a set of constraint functions (signatures). A signature grid is a tuple $\Omega = (G, \pi)$ where π assigns a function $f \in \mathcal{F}$ to each vertex of G.

Holant Sum

Definition

Let $\mathcal{F}$ be a set of constraint functions (signatures). A signature grid is a tuple $\Omega = (G, \pi)$ where π assigns a function $f \in \mathcal{F}$ to each vertex of G.

Definition

For a set of signatures $\mathcal{F}$, $\mathrm{Holant}(\mathcal{F})$ is the following class of problems:
Input: A *signature grid* $\Omega = (G, \pi)$ over $\mathcal{F}$;
Output:

$$\mathrm{Holant}(\Omega; \mathcal{F}) = \sum_{\sigma: E \to \{0,1\}} \prod_{v \in V} f_v(\sigma\mid_{E(v)}),$$

where

- $E(v)$ denotes the incident edges of v and
- $\sigma\mid_{E(v)}$ denotes the restriction of σ to $E(v)$, and $f_v(\sigma\mid_{E(v)})$ is the evaluation of f_v on the ordered input tuple $\sigma\mid_{E(v)}$.

#PL-3-NAE-ICE
INPUT: A planar graph $G = (V, E)$ of maximum degree 3.
OUTPUT: The number of orientations such that no node has all incident edges directed toward it or all incident edges directed away from it.

#PL-3-NAE-ICE

INPUT: A planar graph $G = (V, E)$ of maximum degree 3.
OUTPUT: The number of orientations such that no node has all incident edges directed toward it or all incident edges directed away from it.

So #PL-3-NAE-ICE counts the number of no-sink-no-source orientations.

#PL-3-NAE-ICE
INPUT: A planar graph $G = (V, E)$ of maximum degree 3.
OUTPUT: The number of orientations such that no node has all incident edges directed toward it or all incident edges directed away from it.

So #PL-3-NAE-ICE counts the number of no-sink-no-source orientations.

For simplicity suppose G is 3-regular.

#PL-3-NAE-ICE
INPUT: A planar graph $G = (V, E)$ of maximum degree 3.
OUTPUT: The number of orientations such that no node has all incident edges directed toward it or all incident edges directed away from it.

So #PL-3-NAE-ICE counts the number of no-sink-no-source orientations.

For simplicity suppose G is 3-regular.

Let $f(x, y, z)$ be the NOT-ALL-EQUAL function. This is the constraint at every vertex.

#PL-3-NAE-ICE

INPUT: A planar graph $G = (V, E)$ of maximum degree 3.
OUTPUT: The number of orientations such that no node has all incident edges directed toward it or all incident edges directed away from it.

So #PL-3-NAE-ICE counts the number of no-sink-no-source orientations.

For simplicity suppose G is 3-regular.

Let $f(x, y, z)$ be the NOT-ALL-EQUAL function. This is the constraint at every vertex.

If f is a symmetric function on $\{x_1, x_2, \ldots, x_n\}$, we can denote it as $[f_0, f_1, \ldots, f_n]$, where f_w is the value of f on input of Hamming weight w.

#PL-3-NAE-ICE
INPUT: A planar graph $G = (V, E)$ of maximum degree 3.
OUTPUT: The number of orientations such that no node has all incident edges directed toward it or all incident edges directed away from it.

So #PL-3-NAE-ICE counts the number of no-sink-no-source orientations.

For simplicity suppose G is 3-regular.

Let $f(x, y, z)$ be the NOT-ALL-EQUAL function. This is the constraint at every vertex.

If f is a symmetric function on $\{x_1, x_2, \ldots, x_n\}$, we can denote it as $[f_0, f_1, \ldots, f_n]$, where f_w is the value of f on input of Hamming weight w.

Thus the ternary NOT-ALL-EQUAL function f is $[0, 1, 1, 0]$.

#PL-3-NAE-ICE continued

For every edge, we can replace it by a path of length 2, and assign the binary DISEQUALITY function $[0, 1, 0]$ at the new vertex.

For every edge, we can replace it by a path of length 2, and assign the binary DISEQUALITY function $[0, 1, 0]$ at the new vertex.

The Holant sum on the bipartite Edge-Vertex incidence graph of G is a sum over $2^{2|E|}$ terms.

For every edge, we can replace it by a path of length 2, and assign the binary DISEQUALITY function $[0, 1, 0]$ at the new vertex.

The Holant sum on the bipartite Edge-Vertex incidence graph of G is a sum over $2^{2|E|}$ terms.

Each edge DISEQUALITY function $[0, 1, 0]$ is 1 if the two ends are assigned a different value of $\{0, 1\}$, and is 0 otherwise.

#PL-3-NAE-ICE continued

For every edge, we can replace it by a path of length 2, and assign the binary DISEQUALITY function $[0, 1, 0]$ at the new vertex.

The Holant sum on the bipartite Edge-Vertex incidence graph of G is a sum over $2^{2|E|}$ terms.

Each edge DISEQUALITY function $[0, 1, 0]$ is 1 if the two ends are assigned a different value of $\{0, 1\}$, and is 0 otherwise.

This corresponds to an orientation.

For every edge, we can replace it by a path of length 2, and assign the binary DISEQUALITY function $[0, 1, 0]$ at the new vertex.

The Holant sum on the bipartite Edge-Vertex incidence graph of G is a sum over $2^{2|E|}$ terms.

Each edge DISEQUALITY function $[0, 1, 0]$ is 1 if the two ends are assigned a different value of $\{0, 1\}$, and is 0 otherwise.

This corresponds to an orientation.

Each vertex function $[0, 1, 1, 0]$ evaluates to 1 if the no-sink-no-source condition is satisfied, and it evaluates to 0 otherwise.

Holant Sum as a Dot Product

This Holant Sum can be viewed as a (long) dot product of the following two vectors:

Holant Sum as a Dot Product

This Holant Sum can be viewed as a (long) dot product of the following two vectors:

On LHS: we take the tensor product of all $[0, 1, 0]$, one per each edge.

Holant Sum as a Dot Product

This Holant Sum can be viewed as a (long) dot product of the following two vectors:

On LHS: we take the tensor product of all $[0, 1, 0]$, one per each edge.

On RHS: we take the tensor product of all $[0, 1, 1, 0]$, one per each vertex.

Holant Sum as a Dot Product

This Holant Sum can be viewed as a (long) dot product of the following two vectors:

On LHS: we take the tensor product of all $[0, 1, 0]$, one per each edge.

On RHS: we take the tensor product of all $[0, 1, 1, 0]$, one per each vertex.

The indices of the two (long) vectors (each of dimension $2^{2|E|}$) are matched up by the connection of the graph.

Holographic Transformation

We can perform a local transformation

Holographic Transformation

We can perform a local transformation by $H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

Holographic Transformation

We can perform a local transformation by $H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

$$[0, 1, 1, 0] \mapsto H^{\otimes 3}[0, 1, 1, 0]$$
$$[0, 1, 0](H^{-1})^{\otimes 2} \leftarrow\!\!\shortmid [0, 1, 0]$$

We can perform a local transformation by $H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

$$[0,1,1,0] \mapsto H^{\otimes 3}[0,1,1,0]$$
$$[0,1,0](H^{-1})^{\otimes 2} \leftarrow [0,1,0]$$

$$\begin{aligned} [0,1,1,0] &= \begin{bmatrix} 1 \\ 1 \end{bmatrix}^{\otimes 3} - \begin{bmatrix} 1 \\ 0 \end{bmatrix}^{\otimes 3} - \begin{bmatrix} 0 \\ 1 \end{bmatrix}^{\otimes 3} \\ \mapsto H^{\otimes 3}[0,1,1,0] &= \begin{bmatrix} 2 \\ 0 \end{bmatrix}^{\otimes 3} - \begin{bmatrix} 1 \\ 1 \end{bmatrix}^{\otimes 3} - \begin{bmatrix} 1 \\ -1 \end{bmatrix}^{\otimes 3} = [6,0,-2,0], \end{aligned}$$

We can perform a local transformation by $H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

$$\begin{aligned} [0,1,1,0] &\mapsto H^{\otimes 3}[0,1,1,0] \\ [0,1,0](H^{-1})^{\otimes 2} &\leftarrow\!\!\shortmid [0,1,0] \end{aligned}$$

$$\begin{aligned} [0,1,1,0] &= \begin{bmatrix} 1 \\ 1 \end{bmatrix}^{\otimes 3} - \begin{bmatrix} 1 \\ 0 \end{bmatrix}^{\otimes 3} - \begin{bmatrix} 0 \\ 1 \end{bmatrix}^{\otimes 3} \\ \mapsto H^{\otimes 3}[0,1,1,0] &= \begin{bmatrix} 2 \\ 0 \end{bmatrix}^{\otimes 3} - \begin{bmatrix} 1 \\ 1 \end{bmatrix}^{\otimes 3} - \begin{bmatrix} 1 \\ -1 \end{bmatrix}^{\otimes 3} = [6,0,-2,0], \end{aligned}$$

and

$$\begin{aligned} \leftarrow\!\!\shortmid \quad & [0,1,0] = \begin{bmatrix} 1 & 1 \end{bmatrix}^{\otimes 2} - \begin{bmatrix} 1 & 0 \end{bmatrix}^{\otimes 2} - \begin{bmatrix} 0 & 1 \end{bmatrix}^{\otimes 2} \\ & [0,1,0](H^{-1})^{\otimes 2} = [\frac{1}{2}, 0, \frac{-1}{2}] = \frac{1}{2}[1,0,-1]. \end{aligned}$$

Valiant's Holant Theorem

Theorem

If there is a holographic transformation mapping signature grid Ω *to* Ω'*, then* $Holant_{\Omega} = Holant_{\Omega'}$.

Valiant's Holant Theorem

Theorem

If there is a holographic transformation mapping signature grid Ω to Ω', then $\text{Holant}_{\Omega} = \text{Holant}_{\Omega'}$.

Hence the same quantity is obtained for #PL-3-NAE-ICE if we use the signature $[6, 0, -2, 0] = H^{\otimes 3}[0, 1, 1, 0]$ for each vertex,

Theorem

If there is a holographic transformation mapping signature grid Ω to Ω', then $\text{Holant}_{\Omega} = \text{Holant}_{\Omega'}$.

Hence the same quantity is obtained for #PL-3-NAE-ICE if we use the signature $[6, 0, -2, 0] = H^{\otimes 3}[0, 1, 1, 0]$ for each vertex, and the signature $\frac{1}{2}[1, 0, -1] = [0, 1, 0](H^{-1})^{\otimes 2}$ for each edge.

Holographic Algorithms by Matchgates

Both $[6, 0, -2, 0]$ and $\frac{1}{2}[1, 0, -1]$ are matchgate signatures.

Both $[6, 0, -2, 0]$ and $\frac{1}{2}[1, 0, -1]$ are matchgate signatures.

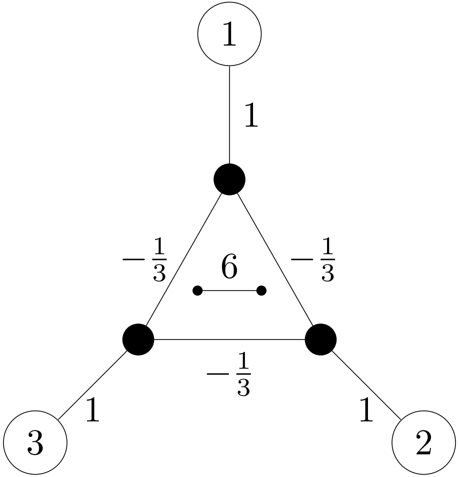

Figure: A matchgate with signature $[6, 0, -2, 0]$

Another Matchgate

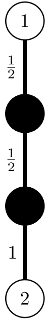

Figure: A matchgate with signature $\frac{1}{2}[1, 0, -1]$

Thus #PL-3-NAE-ICE is computable in P.

A list of symmetric matchgate signatures

Theorem

A symmetric signature is the signature of a matchgate iff it has the following form, for some $a, b \in \mathbb{C}$ and integer k (we take the convention that $0^0 = 1$):

1. $[a^k b^0, 0, a^{k-1} b, 0, a^{k-2} b^2, 0, \ldots, a^0 b^k]$ *(arity $2k \geq 2$)*
2. $[a^k b^0, 0, a^{k-1} b, 0, a^{k-2} b^2, 0, \ldots, a^0 b^k, 0]$ *(arity $2k + 1 \geq 1$)*
3. $[0, a^k b^0, 0, a^{k-1} b, 0, a^{k-2} b^2, 0, \ldots, a^0 b^k]$ *(arity $2k + 1 \geq 1$)*
4. $[0, a^k b^0, 0, a^{k-1} b, 0, a^{k-2} b^2, 0, \ldots, a^0 b^k, 0]$ *(arity $2k + 2 \geq 2$).*

1. L. G. Valiant. Holographic Algorithms, SIAM J. Comput., 37(5) (2008) 1565-1594. (FOCS 2004: 306-315.)

2. J-Y. Cai, P. Lu. Holographic algorithms: From art to science. J. Comput. Syst. Sci. 77(1): 41-61 (2011). (STOC 2007: 401-410)

3. J-Y. Cai, A. Gorenstein: Matchgates Revisited. Theory of Computing 10: 167-197 (2014)

http://www.cs.wisc.edu/~jyc/dichotomy-book.pdf http://www.cs.wisc.edu/~jyc/dichotomy-book.pdf

Some papers can be found on my web site
http://www.cs.wisc.edu/~jyc

THANK YOU!