Beyond Windability: Approximability of the Four-Vertex Model

Xiongxin Yang

Northeast Normal University ⟶ NYU Shanghai

Based on joint work with

Tianyu Liu

Stanford ⟶ Meta

Counting Complexity

Counting Complexity

Counting problem

U: ground set (with a weight function $w : U \to \mathbb{R}^+(\mathbb{C})$)

Ω: a collection of subsets of U with a certain property

Goal: compute $|\Omega|$ or $Z = \sum_{S \in \Omega} \prod_{i \in S} w(i)$

Counting Complexity

Counting problem

U: ground set (with a weight function $w : U \to \mathbb{R}^+(\mathbb{C})$)

Ω: a collection of subsets of U with a certain property

Goal: compute $|\Omega|$ or $Z = \sum_{S \in \Omega} \prod_{i \in S} w(i)$

Complexity class #P [Valiant, 1979]

a counting analogue of NP

#P-complete problem:
#SAT, #BipartitePerfectMatching, #k-coloring$\cdots$

Counting Complexity

Counting problem

U: ground set (with a weight function $w : U \to \mathbb{R}^+(\mathbb{C})$)

Ω: a collection of subsets of U with a certain property

Goal: compute $|\Omega|$ or $Z = \sum_{S \in \Omega} \prod_{i \in S} w(i)$

Complexity class #P [Valiant, 1979]

a counting analogue of NP

#P-complete problem:
#SAT, #BipartitePerfectMatching, #k-coloring$\cdots$

Other applications

- determinant/permanents of matrices
- volume of convex bodies
- partition function in statistical physics

Counting Complexity

Counting problem

U: ground set (with a weight function $w : U \to \mathbb{R}^+(\mathbb{C})$)

Ω: a collection of subsets of U with a certain property

Goal: compute $|\Omega|$ or $Z = \sum_{S\in\Omega}\prod_{i\in S} w(i)$

Complexity class #P [Valiant, 1979]

a counting analogue of NP

#P-complete problem:
#SAT, #BipartitePerfectMatching, #k-coloring$\cdots$

Other applications

- determinant/permanents of matrices
- volume of convex bodies
- partition function in statistical physics

This talk!

Holant Problem

[Cai, Lu, and Xia, 2009]

Holant Problem

[Cai, Lu, and Xia, 2009]

Signature Grid $\Omega = (G, \pi)$ over signature set $\mathscr{F}$

mapping π

- assigns to each vertex $v \in V$ an $f_v \in \mathscr{F}$
- a linear order of the incident edges at v.

Holant Problem

[Cai, Lu, and Xia, 2009]

Signature Grid $\Omega = (G, \pi)$ over signature set $\mathscr{F}$

mapping π

- assigns to each vertex $v \in V$ an $f_v \in \mathscr{F}$
- a linear order of the incident edges at v.

$\mathscr{F}$: AT-Most-One

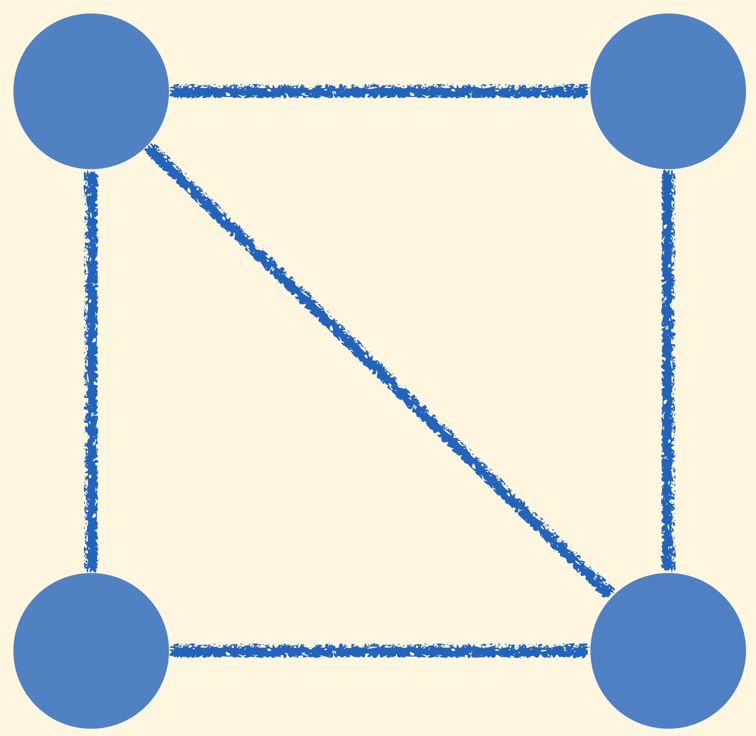
$$f^{(1)}(0) = 1,\ f^{(1)}(1) = 1$$

$$f^{(2)}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

$$f^{(3)}\left(\begin{bmatrix} 000 & 001 & 010 & 011 \\ 100 & 101 & 110 & 111 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\cdots$$

$G = (V, E)$



Holant Problem

[Cai, Lu, and Xia, 2009]

Signature Grid $\Omega = (G, \pi)$ over signature set $\mathscr{F}$

mapping π

- assigns to each vertex $v \in V$ an $f_v \in \mathscr{F}$
- a linear order of the incident edges at v.

$\mathscr{F}$: AT-Most-One

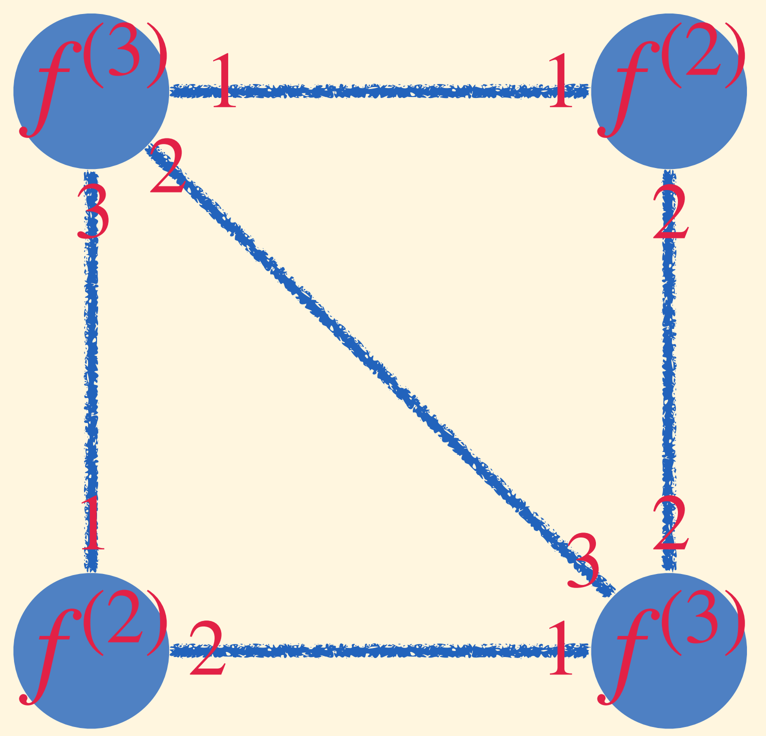
$$f^{(1)}(0) = 1,\ f^{(1)}(1) = 1$$

$$f^{(2)}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

$$f^{(3)}\left(\begin{bmatrix} 000 & 001 & 010 & 011 \\ 100 & 101 & 110 & 111 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\cdots$$

$G = (V, E)$



Holant Problem

[Cai, Lu, and Xia, 2009]

Signature Grid $\Omega = (G, \pi)$ over signature set $\mathscr{F}$

mapping π

- assigns to each vertex $v \in V$ an $f_v \in \mathscr{F}$
- a linear order of the incident edges at v.

Holant value

$$\mathrm{Holant}_{\Omega}(\mathscr{F}) = \sum_{\sigma: E \to [q]} \prod_{v \in V} f_v(\sigma|_{E(v)})$$

$\mathscr{F}$: AT-Most-One

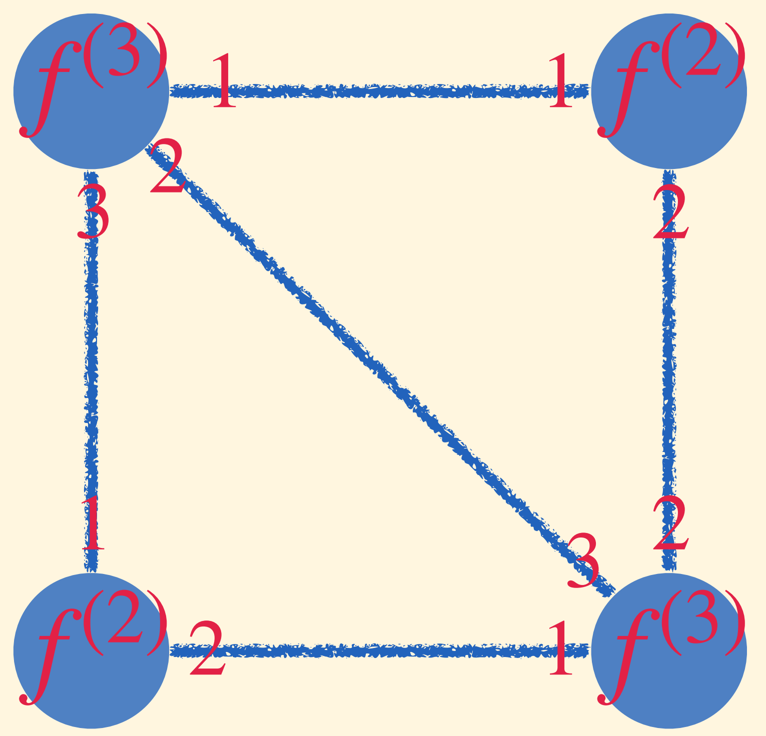
$$f^{(1)}(0) = 1,\ f^{(1)}(1) = 1$$

$$f^{(2)}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

$$f^{(3)}\left(\begin{bmatrix} 000 & 001 & 010 & 011 \\ 100 & 101 & 110 & 111 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$\cdots$

$G = (V, E)$



Holant Problem

[Cai, Lu, and Xia, 2009]

Signature Grid $\Omega = (G, \pi)$ over signature set $\mathscr{F}$

mapping π

- assigns to each vertex $v \in V$ an $f_v \in \mathscr{F}$
- a linear order of the incident edges at v.

Holant value

$$\mathrm{Holant}_\Omega(\mathscr{F}) = \sum_{\sigma: E \to [q]} \prod_{v \in V} f_v(\sigma|_{E(v)})$$

$\mathscr{F}$: AT-Most-One

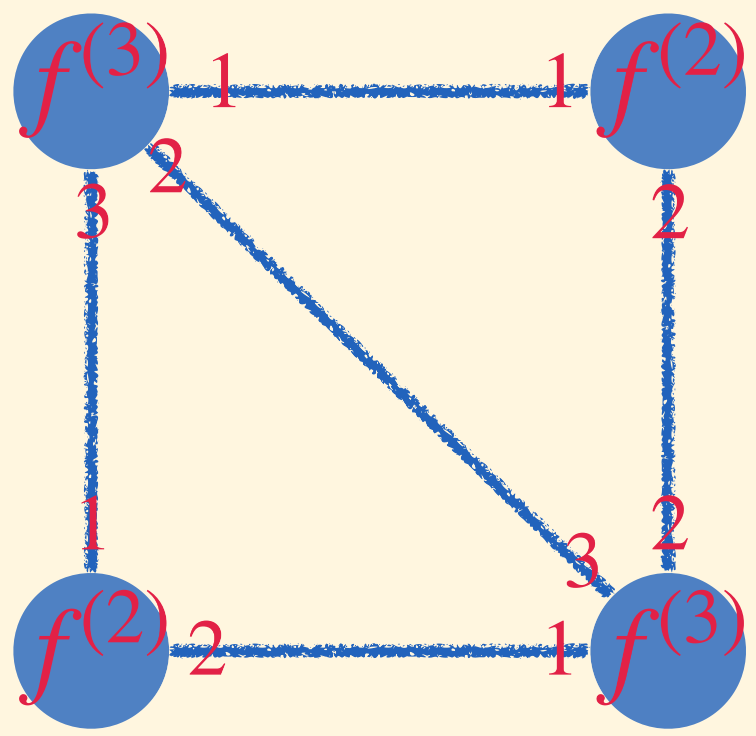
$$f^{(1)}(0) = 1,\ f^{(1)}(1) = 1$$

$$f^{(2)}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

$$f^{(3)}\left(\begin{bmatrix} 000 & 001 & 010 & 011 \\ 100 & 101 & 110 & 111 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$\cdots$

$G = (V, E)$



Holant $= 1 + 5 + 2 = 8 =$ #Matching

Holant Problem

[Cai, Lu, and Xia, 2009]

Signature Grid $\Omega = (G, \pi)$ over signature set $\mathscr{F}$

mapping π

- assigns to each vertex $v \in V$ an $f_v \in \mathscr{F}$
- a linear order of the incident edges at v.

Holant value

$$\text{Holant}_\Omega(\mathscr{F}) = \sum_{\sigma: E \to [q]} \prod_{v \in V} f_v(\sigma|_{E(v)})$$

$\mathscr{F}$: AT-Most-One

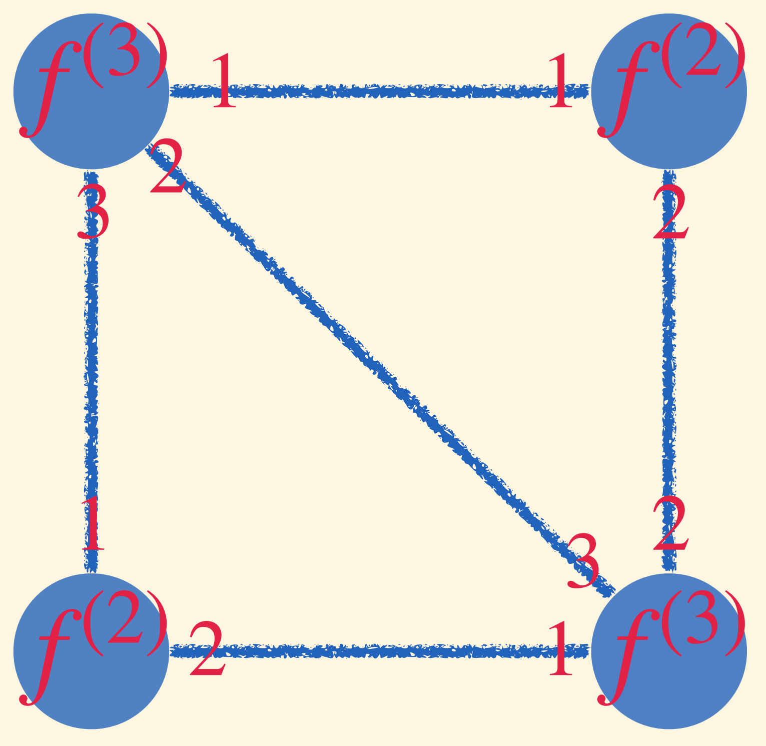
$$f^{(1)}(0) = 1,\ f^{(1)}(1) = 1$$

$$f^{(2)}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

$$f^{(3)}\left(\begin{bmatrix} 000 & 001 & 010 & 011 \\ 100 & 101 & 110 & 111 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$\cdots$

$G = (V, E)$



Holant = 1 + 5 + 2 = 8 = #Matching

Other examples:

#PerfectMatching, #CycleCover, #EdgeCover

#CSP, #GH

Holant Problem

[Cai, Lu, and Xia, 2009]

Signature Grid $\Omega = (G, \pi)$ over signature set $\mathscr{F}$

mapping π

- assigns to each vertex $v \in V$ an $f_v \in \mathscr{F}$
- a linear order of the incident edges at v.

$\mathscr{F}$: AT-Most-One

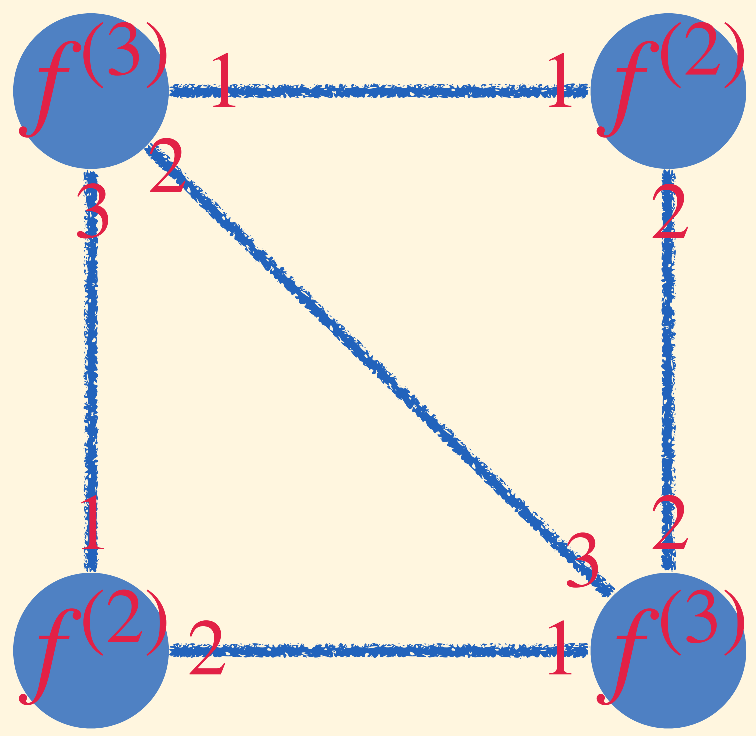
$f^{(1)}(0) = 1,\ f^{(1)}(1) = 1$

$$f^{(2)}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

$$f^{(3)}\left(\begin{bmatrix} 000 & 001 & 010 & 011 \\ 100 & 101 & 110 & 111 \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$\cdots$

$G = (V, E)$



Holant value

$$\mathrm{Holant}_\Omega(\mathscr{F}) = \sum_{\sigma: E \to [q]} \prod_{v \in V} f_v(\sigma|_{E(v)})$$

Holant $= 1 + 5 + 2 = 8 =$ #Matching

Other examples:

#PerfectMatching, #CycleCover, #EdgeCover

#CSP, #GH

Dichotomy theorem

find explicit dichotomy criterion for $\mathscr{F}$

computing the Holant value is

- solvable in polynomial time
- #P-hard

Real Boolean case was done!
[Shao and Cai, 2020]

Approximate Counting

How to approximate #P-complete problems?

Approximate Counting

How to approximate #P-complete problems?

FPTAS

(**f**ully **p**olynomial **t**ime **a**pproximation **s**cheme)

Deterministic algorithms that solves the problem in time poly$(n, \frac{1}{\varepsilon})$

Output a number $\hat{Z}$ such that

$$(1-\varepsilon)Z \le \hat{Z} \le (1+\varepsilon)Z$$

Approximate Counting

How to approximate #P-complete problems?

FPTAS

(**f**ully **p**olynomial **t**ime **a**pproximation **s**cheme)

Deterministic algorithms that solves the problem in time poly$(n, \frac{1}{\varepsilon})$

Output a number $\hat{Z}$ such that

$$(1-\varepsilon)Z \le \hat{Z} \le (1+\varepsilon)Z$$

FPRAS

(**f**ully **p**olynomial time **r**andomised **a**pproximation **s**cheme)

Randomised algorithms that solves the problem in time poly$(n, \frac{1}{\varepsilon})$

Output a *random* number $\hat{Z}$ such that

$$\Pr\left[(1-\varepsilon)Z \le \hat{Z} \le (1+\varepsilon)Z\right] \ge \frac{3}{4}$$

Approximate Counting

How to approximate #P-complete problems?

FPTAS

(**f**ully **p**olynomial **t**ime **a**pproximation **s**cheme)

Deterministic algorithms that solves the problem in time $\text{poly}(n, \frac{1}{\varepsilon})$

Output a number $\hat{Z}$ such that

$$(1-\varepsilon)Z \le \hat{Z} \le (1+\varepsilon)Z$$

FPRAS

(**f**ully **poly** … imation scheme)

Rand … solves the … $\frac{1}{\varepsilon})$

can be boosted to $1-\delta$

using $O(\log \delta)$ repeated trials

Output a *random* number $\hat{Z}$ such that

$$\Pr\left[(1-\varepsilon)Z \le \hat{Z} \le (1+\varepsilon)Z\right] \ge \boxed{\frac{3}{4}}$$

Approximate Counting

How to approximate #P-complete problems?

FPTAS

(**f**ully **p**olynomial **t**ime **a**pproximation **s**cheme)

Deterministic algorithms that solves the problem in time poly$(n, \frac{1}{\varepsilon})$

Output a number $\hat{Z}$ such that

$$(1-\varepsilon)Z \le \hat{Z} \le (1+\varepsilon)Z$$

FPRAS

(**f**ully **p**olynomial time **r**andomised **a**pproximation **s**cheme)

Randomised algorithms that solves the problem in time poly$(n, \frac{1}{\varepsilon})$

Output a *random* number $\hat{Z}$ such that

$$\Pr\left[(1-\varepsilon)Z \le \hat{Z} \le (1+\varepsilon)Z\right] \ge \frac{3}{4}$$

Approach:

Markov Chain Monte Carlo

Approximate Counting

How to approximate #P-complete problems?

FPTAS

(**f**ully **p**olynomial **t**ime **a**pproximation **s**cheme)

Deterministic algorithms that solves the problem in time $\mathrm{poly}(n, \frac{1}{\varepsilon})$

Output a number $\hat{Z}$ such that

$$(1-\varepsilon)Z \le \hat{Z} \le (1+\varepsilon)Z$$

Approaches:

- decay of correlation [Weitz, 2006]
- zero-freeness [Barvinok, 2016]
- liner programming [Moitra, 2019]
- cluster expansion [Helmuth, Perkins and Regts, 2019]

FPRAS

(**f**ully **p**olynomial time **r**andomised **a**pproximation **s**cheme)

Randomised algorithms that solves the problem in time $\mathrm{poly}(n, \frac{1}{\varepsilon})$

Output a *random* number $\hat{Z}$ such that

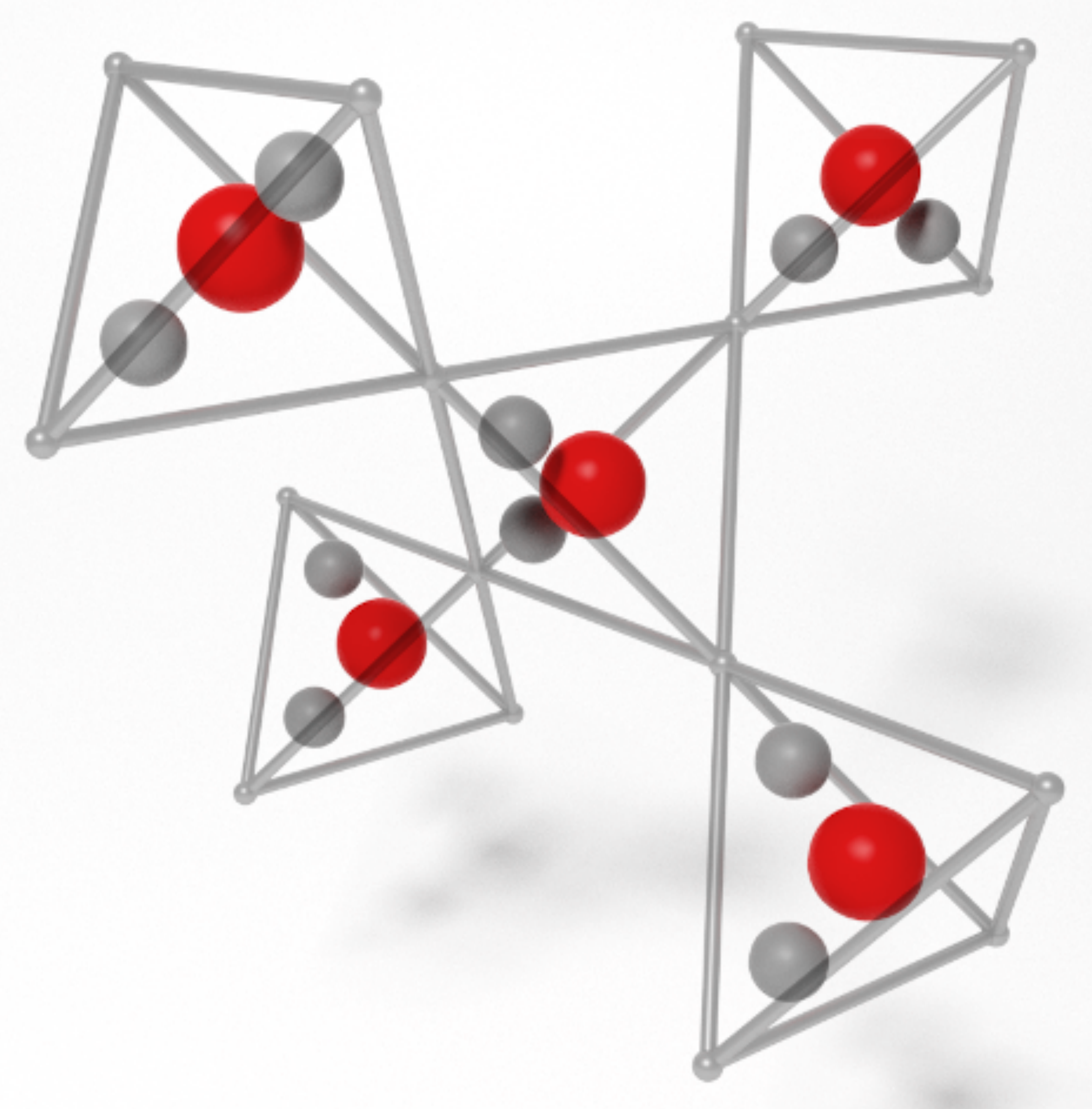
$$\Pr\left[(1-\varepsilon)Z \le \hat{Z} \le (1+\varepsilon)Z\right] \ge \frac{3}{4}$$

Approach:

Markov Chain Monte Carlo

Six-Vertex Model

[Pauling, 1935]



(Figure by Jonas Greitemann)

A model to describe the properties of ice H_2O

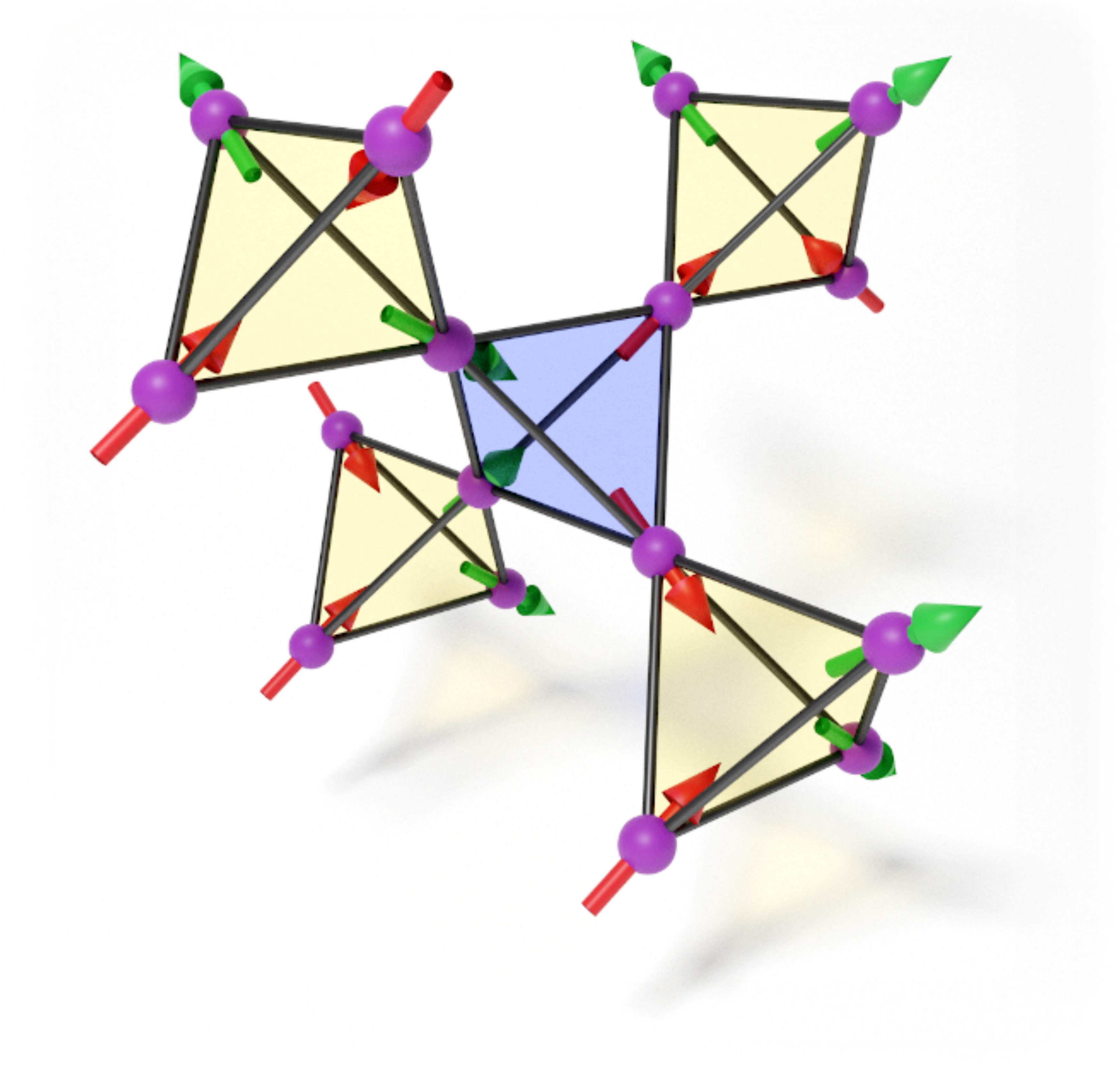
ice condition

- Each O has 4 nearest neighbors with O-H-O bonds
- Each H is in 2 possible positions (closer to one O or another)
- Each O must be surrounded by two H's near to it and two on the far side

Six-Vertex Model

[Pauling, 1935]

A model to describe the properties of ice H_2O



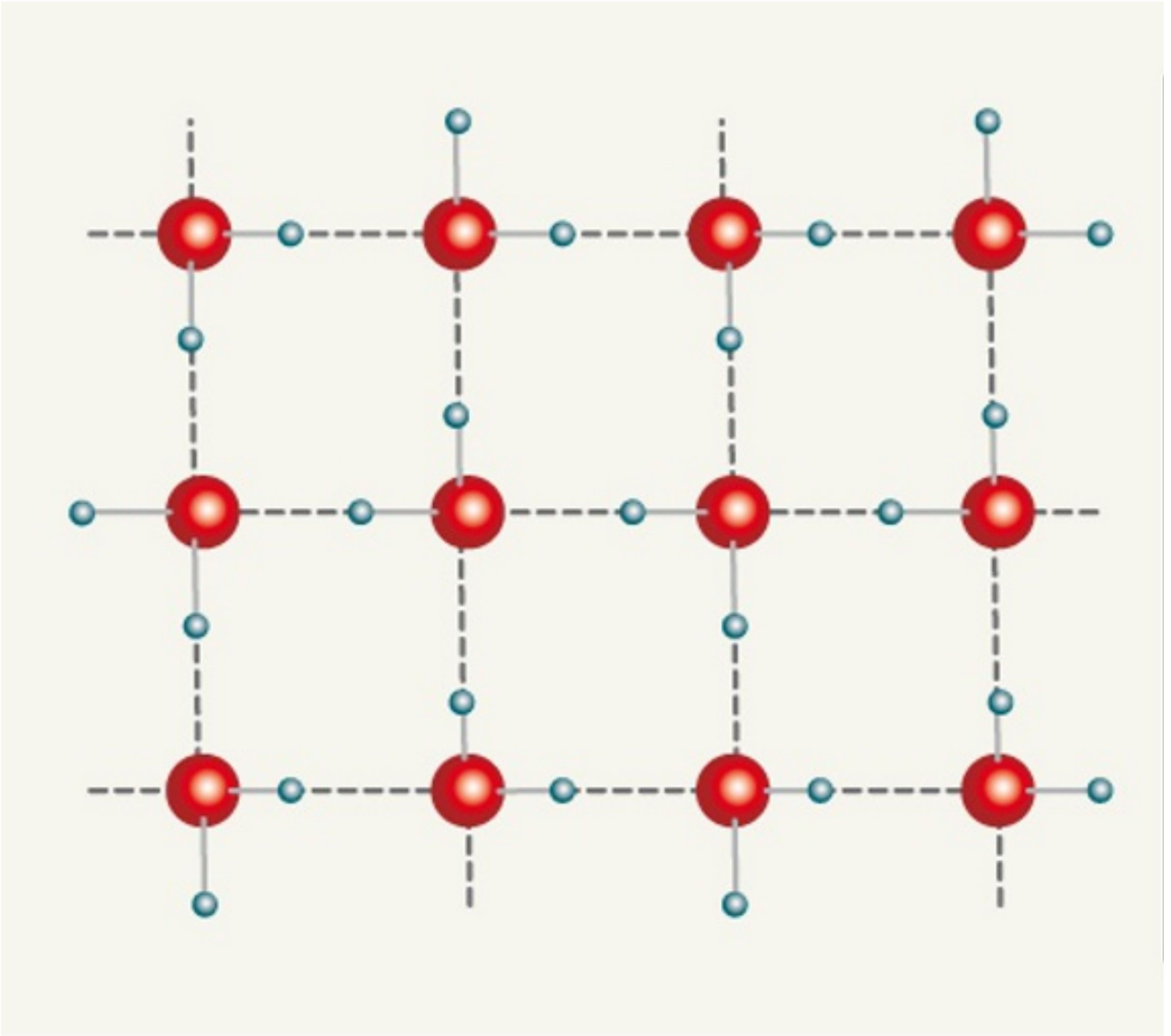
(Figure by Jonas Greitemann)

ice condition

- Each O has 4 nearest neighbors with O-H-O bonds
- Each H is in 2 possible positions (closer to one O or another)
- Each O must be surrounded by two H's near to it and two on the far side

Exact Solvable Model: Square Ice

[Lieb, 1967]



(Figure by Mark Peplow)

two-dimensional version of real ice:
square lattice with ice condition

Lieb's square ice constant

N: the number of O's

Z: the ***partition function***

The *exact solution* (partition function peer vertex)

$$W = \lim_{N\to\infty} Z^{1/N} = \left(\frac{4}{3}\right)^{3/2} \approx 1.5396007\cdots$$

Six-Vertex Model on Graphs

The six-vertex model is defined on 4-*regular* graphs

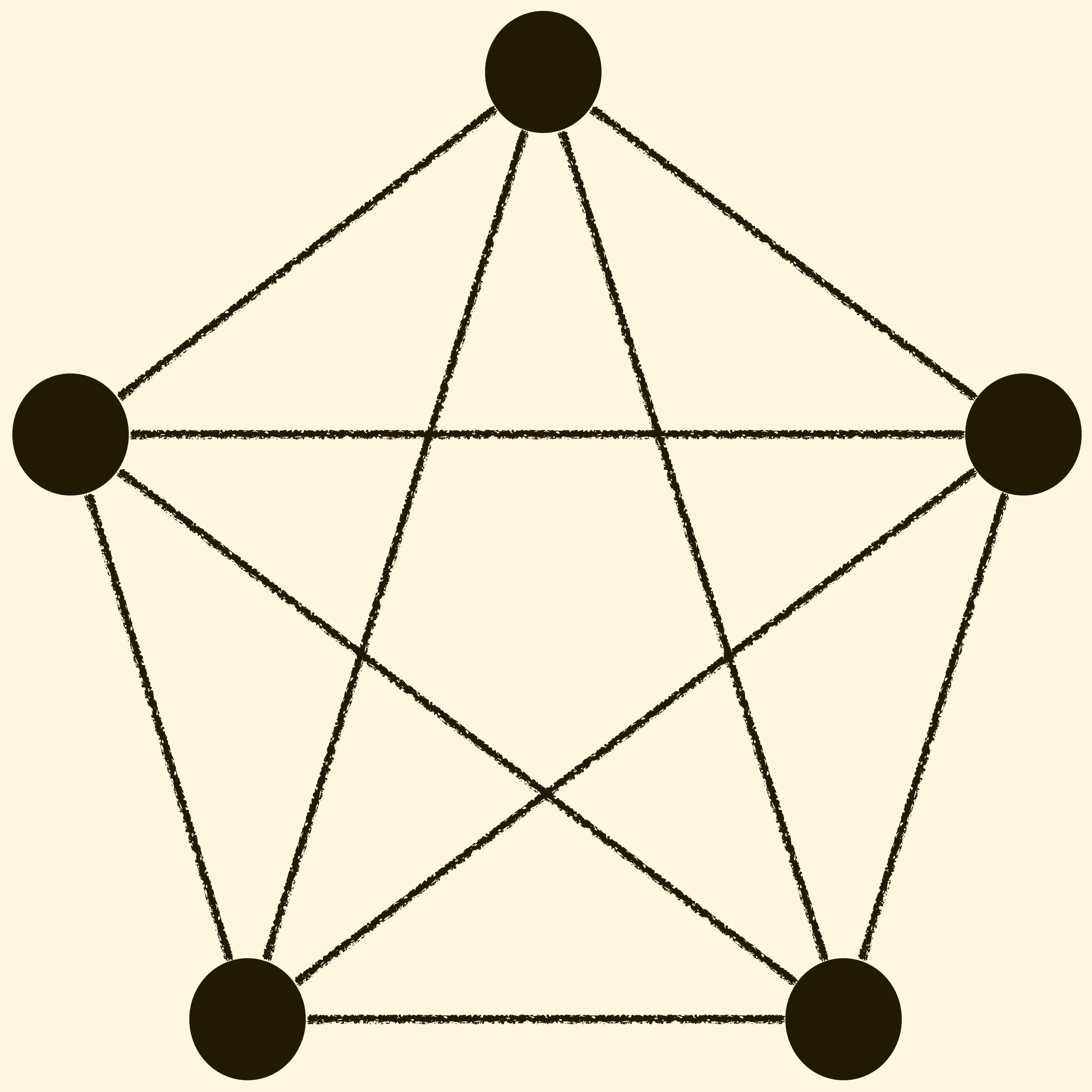
O — vertex of a graph H — arrows on edges

Six-Vertex Model on Graphs

The six-vertex model is defined on 4-*regular* graphs

O — vertex of a graph

H — arrows on edges



Six-Vertex Model on Graphs

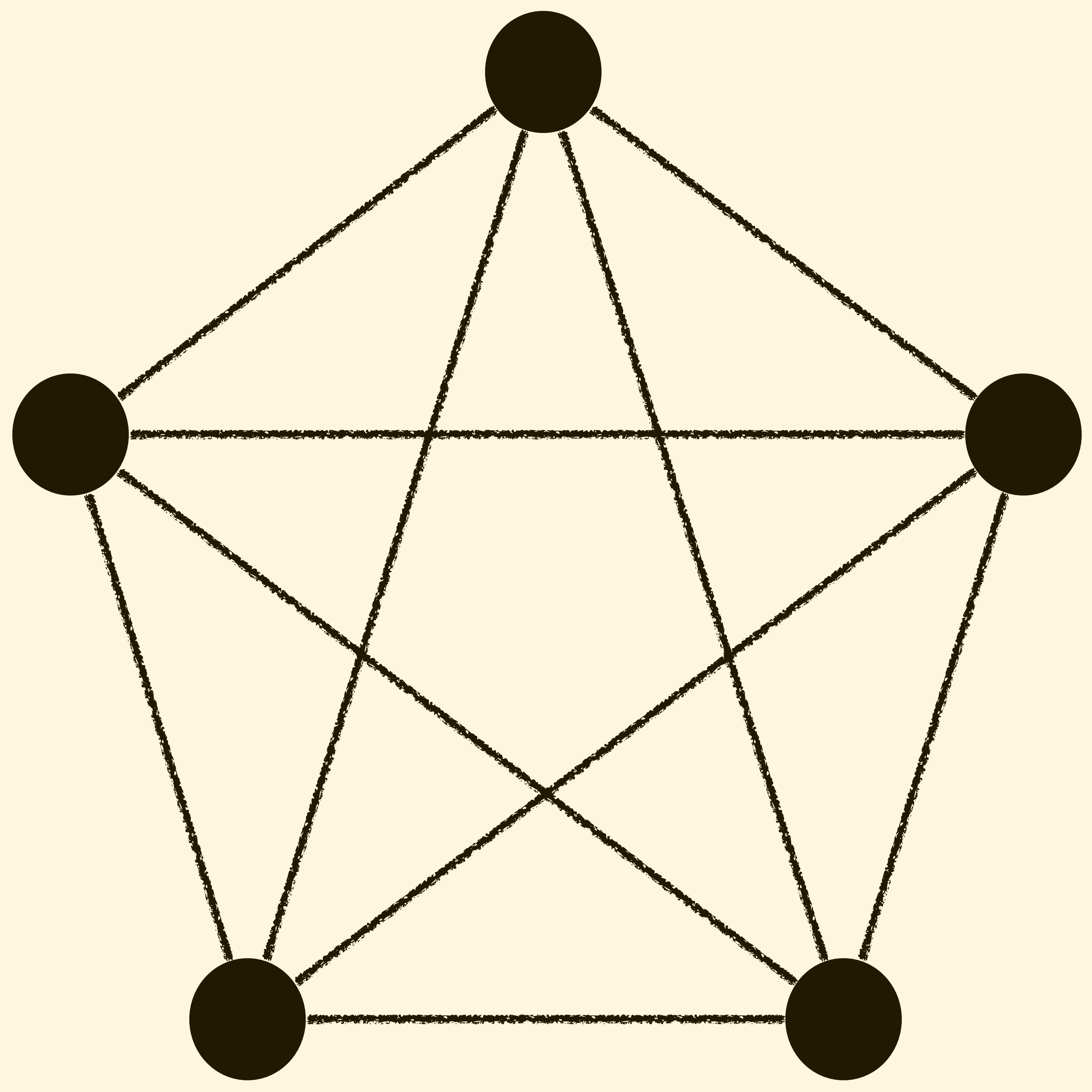
The six-vertex model is defined on 4-*regular* graphs

O — vertex of a graph H — arrows on edges

Valid global configurations

Eulerian orientations of the underlying graph:
each vertex has equal in-degree and out-degree

ice condition



Six-Vertex Model on Graphs

The six-vertex model is defined on 4-*regular* graphs

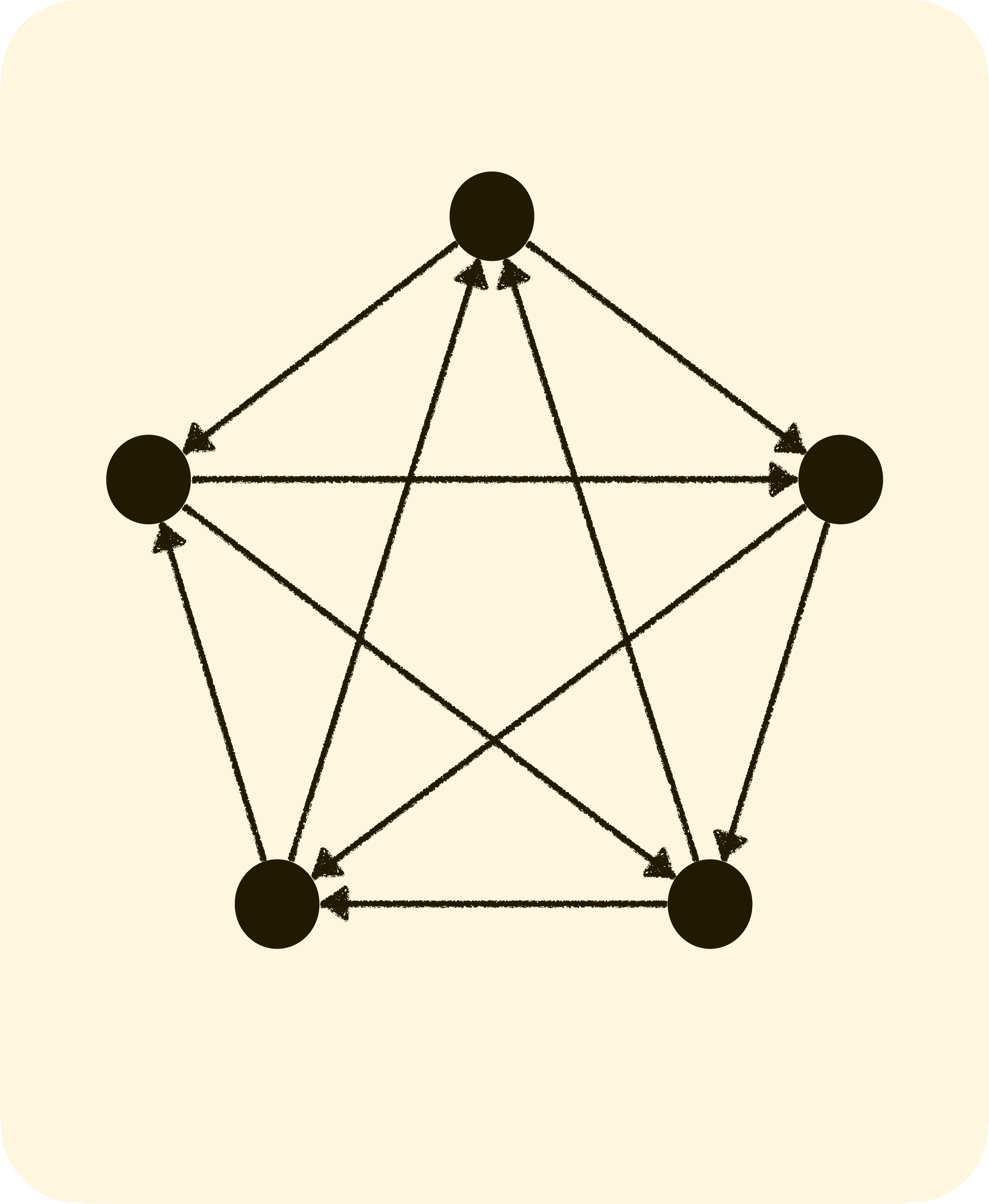
O — vertex of a graph

H — arrows on edges

Valid global configurations

Eulerian orientations of the underlying graph:
each vertex has equal in-degree and out-degree

ice condition



Six-Vertex Model on Graphs

The six-vertex model is defined on 4-*regular* graphs

O — vertex of a graph H — arrows on edges

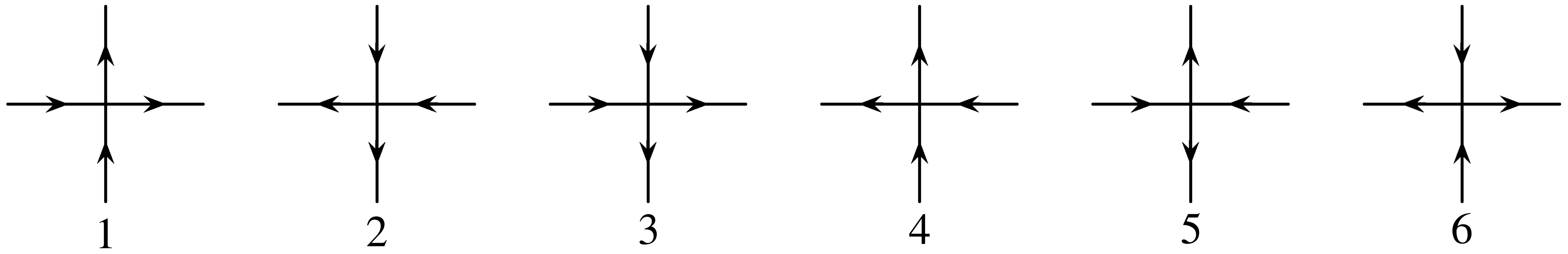
Valid global configurations

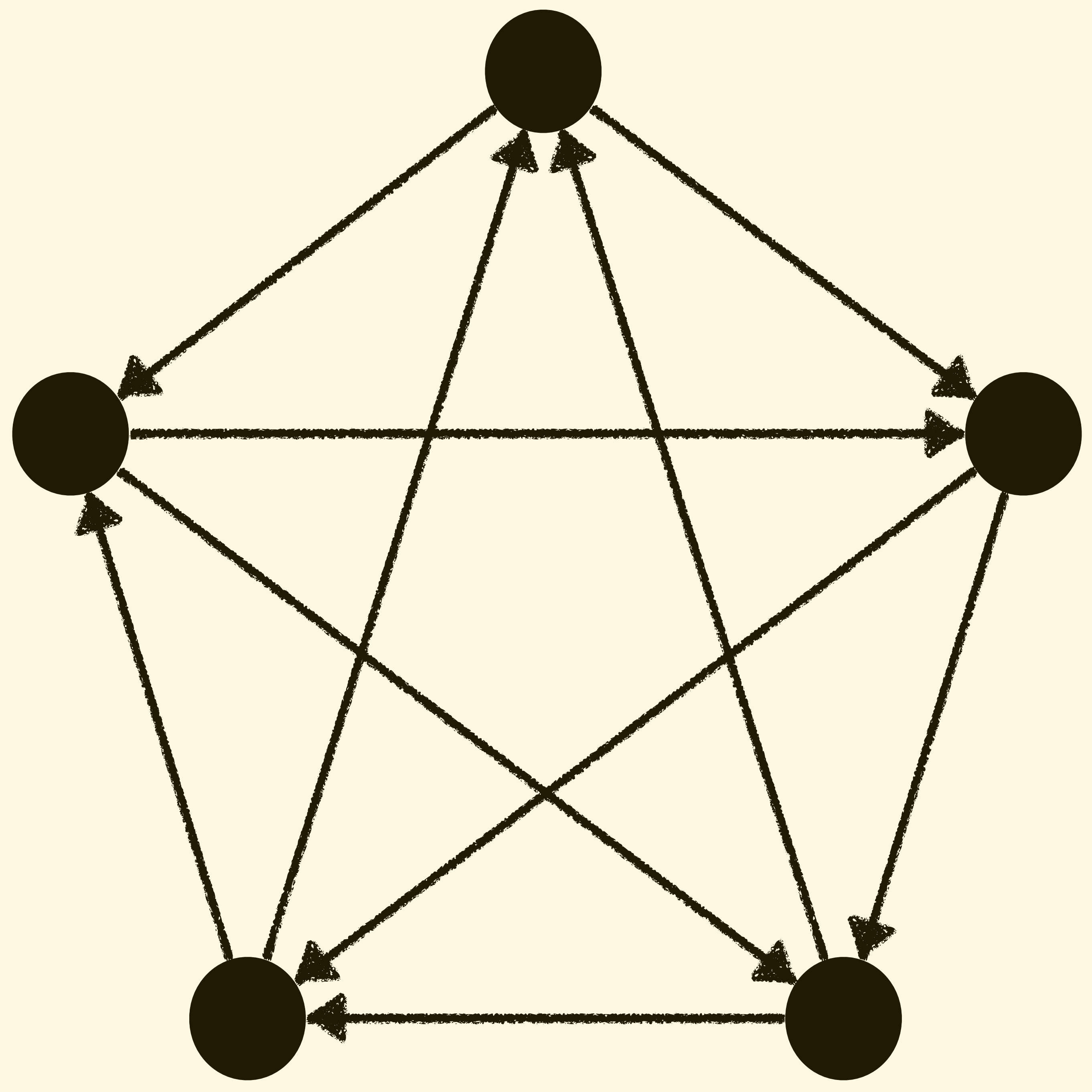
Eulerian orientations of the underlying graph:
each vertex has equal in-degree and out-degree

ice condition

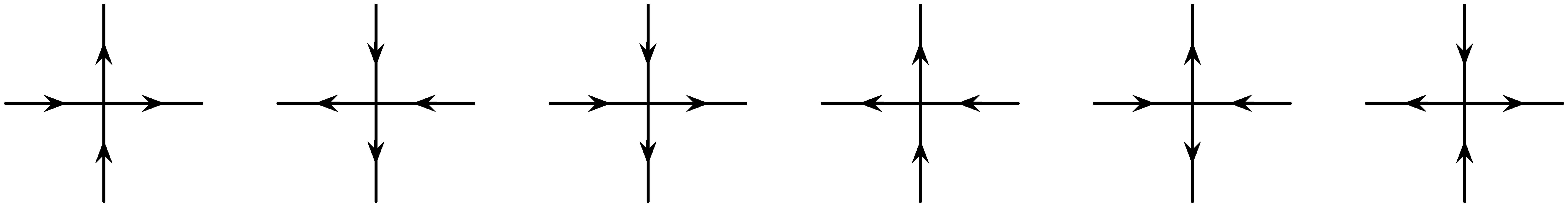
Valid local configurations

$$\binom{4}{2} = 6 \text{ types}$$



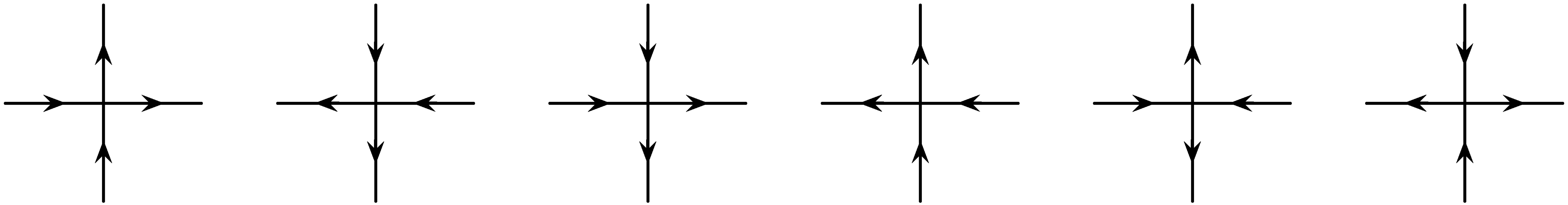


Partition Function



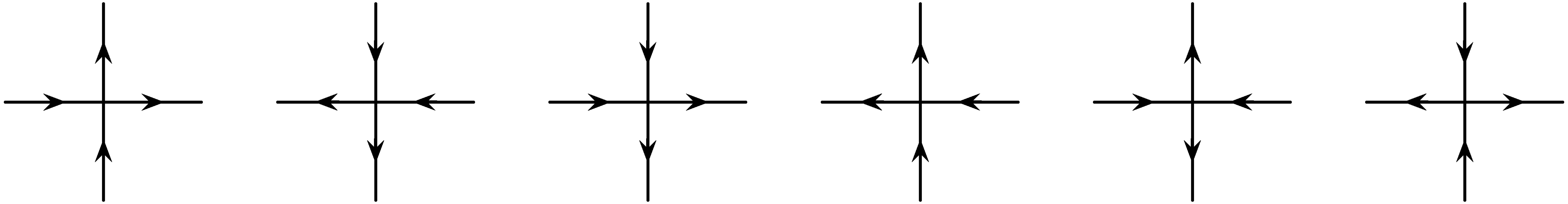
1 2 3 4 5 6

Partition Function



	1	2	3	4	5	6
energy parameter	ε_1	ε_2	ε_3	ε_4	ε_5	ε_6

Partition Function



	1	2	3	4	5	6
energy parameter	ε_1	ε_2	ε_3	ε_4	ε_5	ε_6

inverse temperature

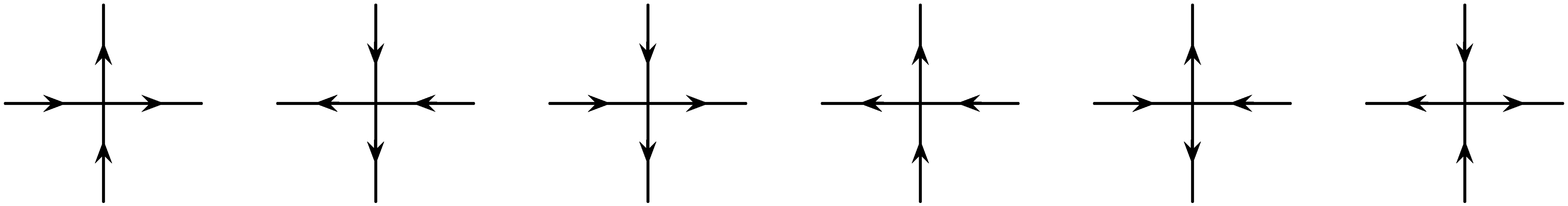
partition function

$$Z_{6\mathrm{V}}(G; \varepsilon_1, \varepsilon_2, \cdots, \varepsilon_6) = \sum_{\tau \in \mathscr{E}\mathscr{O}(G)} \exp(-\beta(n_1\varepsilon_1 + n_2\varepsilon_2 + \cdots + n_6\varepsilon_6))$$

n_i = # type i vertices under τ

the set of Eulerian orientations of G

Partition Function



energy parameter	1	2	3	4	5	6
	ε_1	ε_2	ε_3	ε_4	ε_5	ε_6

inverse temperature

partition function

$$Z_{6V}(G; \varepsilon_1, \varepsilon_2, \cdots, \varepsilon_6) = \sum_{\tau \in \mathscr{E}\mathscr{O}(G)} \exp(-\beta(n_1\varepsilon_1 + n_2\varepsilon_2 + \cdots + n_6\varepsilon_6))$$

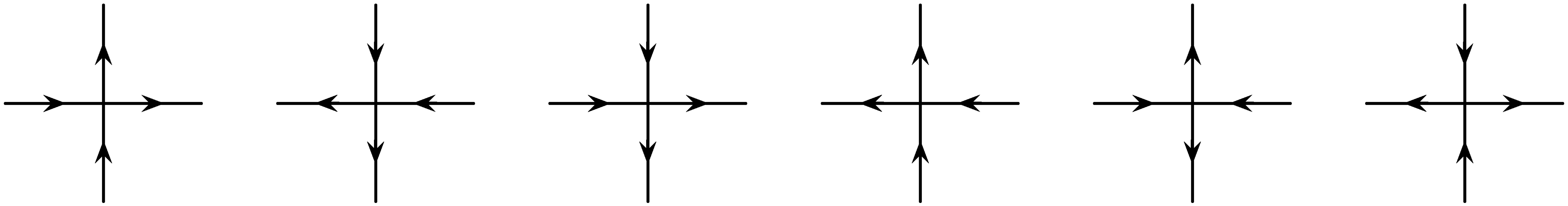
$n_i = \#$ type i vertices under τ

the set of Eulerian orientations of G

arrow reversal symmetry
(zero field assumption)

$$\exp(-\beta\varepsilon_1) = \exp(-\beta\varepsilon_2) = a \geq 0$$
$$\exp(-\beta\varepsilon_3) = \exp(-\beta\varepsilon_4) = b \geq 0$$
$$\exp(-\beta\varepsilon_5) = \exp(-\beta\varepsilon_6) = c \geq 0$$

Partition Function



energy parameter	1	2	3	4	5	6
	ε_1	ε_2	ε_3	ε_4	ε_5	ε_6

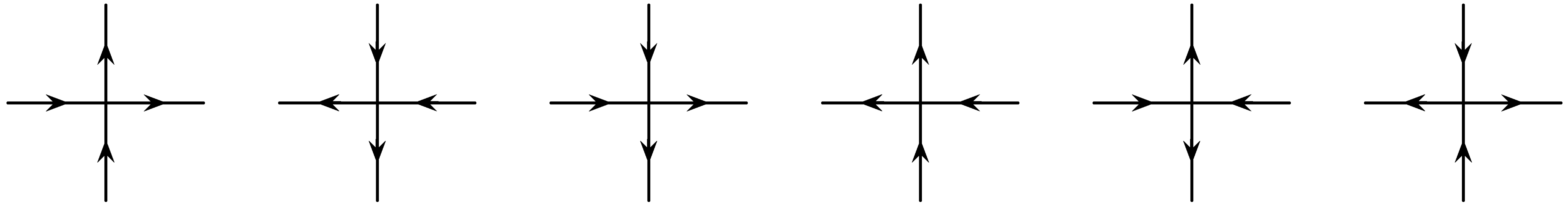
partition function

$$Z_{6V}(G; a, b, c) = \sum_{\tau \in \mathcal{E}\mathcal{O}(G)} a^{n_1+n_2} b^{n_3+n_4} c^{n_5+n_6}$$

arrow reversal symmetry
(zero field assumption)

$$\exp(-\beta\varepsilon_1) = \exp(-\beta\varepsilon_2) = a \geq 0$$
$$\exp(-\beta\varepsilon_3) = \exp(-\beta\varepsilon_4) = b \geq 0$$
$$\exp(-\beta\varepsilon_5) = \exp(-\beta\varepsilon_6) = c \geq 0$$

Partition Function



energy parameter	1	2	3	4	5	6
	ε_1	ε_2	ε_3	ε_4	ε_5	ε_6

partition function

$$Z_{6V}(G; a, b, c) = \sum_{\tau \in \mathscr{E}\mathscr{O}(G)} a^{n_1+n_2} b^{n_3+n_4} c^{n_5+n_6}$$

arrow reversal symmetry (zero field assumption)

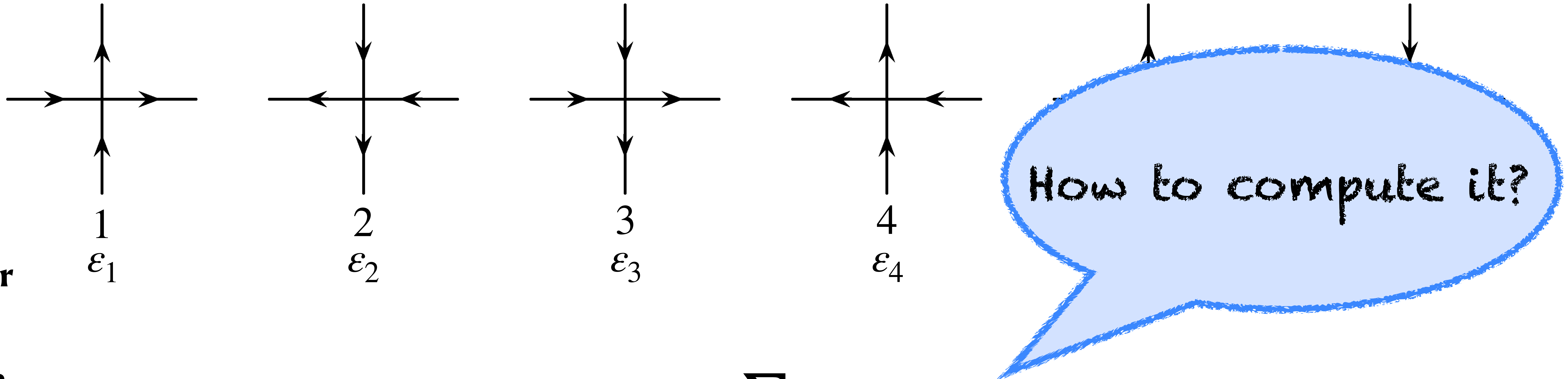
$$\exp(-\beta\varepsilon_1) = \exp(-\beta\varepsilon_2) = a \geq 0$$
$$\exp(-\beta\varepsilon_3) = \exp(-\beta\varepsilon_4) = b \geq 0$$
$$\exp(-\beta\varepsilon_5) = \exp(-\beta\varepsilon_6) = c \geq 0$$

Some applications

- Water ice: $a = b = c = 1$
- KDP model: $a = b > 1, c = 1$
- Rys F model: $a = b = 1, c > 1$

Partition Function

energy parameter



partition function

$$Z_{6\mathrm{V}}(G; a, b, c) = \sum_{\tau \in \mathscr{E}\mathscr{O}(G)} a^{n_1+n_2} b^{n_3+n_4} c^{n_5+n_6}$$

arrow reversal symmetry
(zero field assumption)

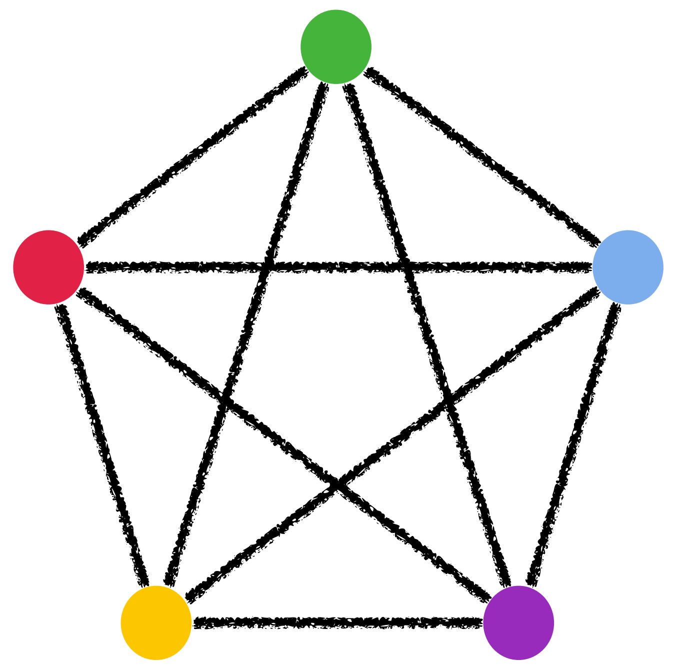
$$\exp(-\beta\varepsilon_1) = \exp(-\beta\varepsilon_2) = a \geq 0$$
$$\exp(-\beta\varepsilon_3) = \exp(-\beta\varepsilon_4) = b \geq 0$$
$$\exp(-\beta\varepsilon_5) = \exp(-\beta\varepsilon_6) = c \geq 0$$

Some applications

- Water ice: $a = b = c = 1$
- KDP model: $a = b > 1, c = 1$
- Rys F model: $a = b = 1, c > 1$

Six-Vertex Model as Holant Problem

Six-Vertex Model as Holant Problem



Edge-vertex incidence graph

bipartite graph $G' = (U_E, U_V, E')$

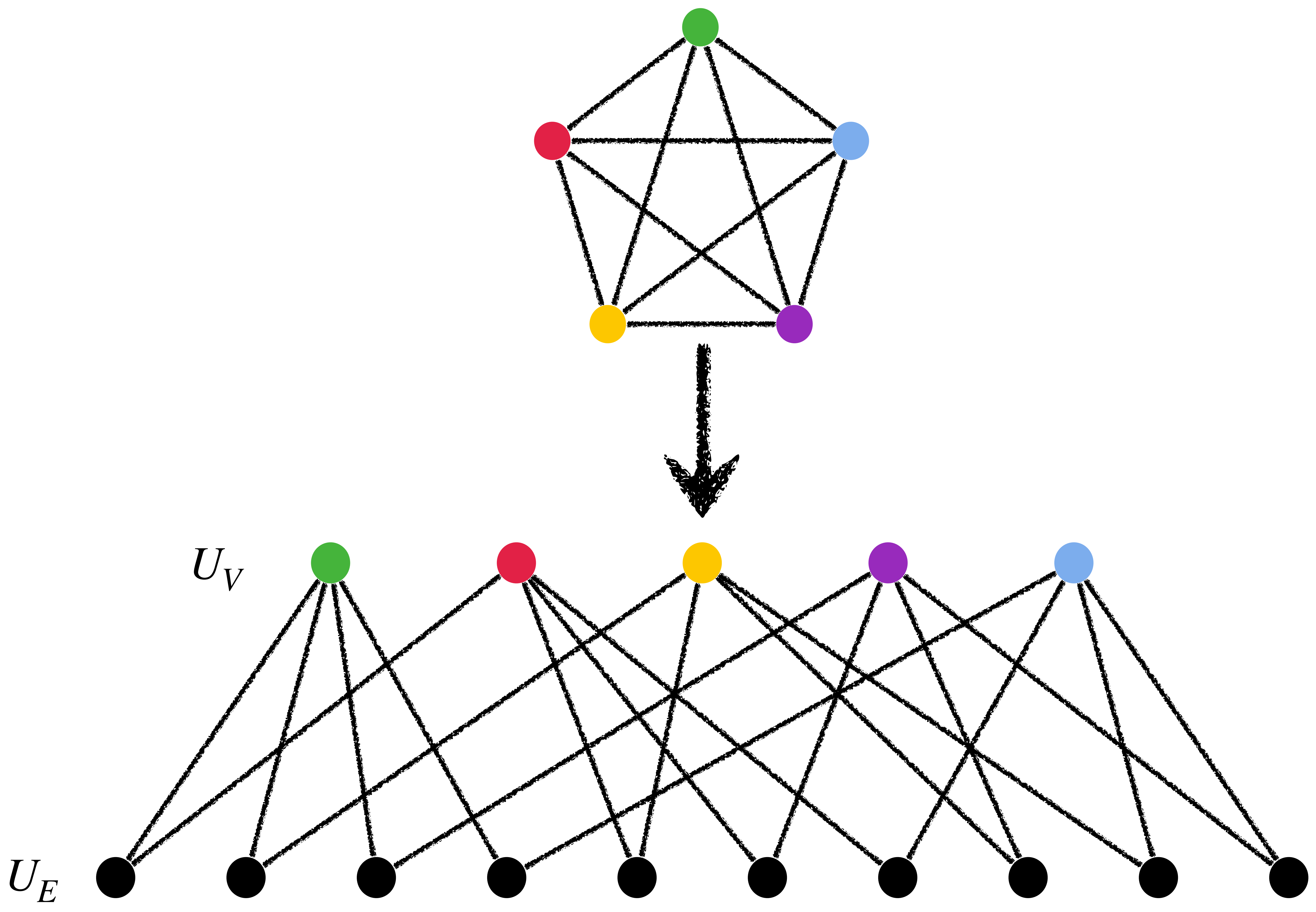
edge $(u_e, u_v) \in E'$ iff $e \in E(v)$

Six-Vertex Model as Holant Problem

Edge-vertex incidence graph

bipartite graph $G' = (U_E, U_V, E')$

edge $(u_e, u_v) \in E'$ iff $e \in E(v)$

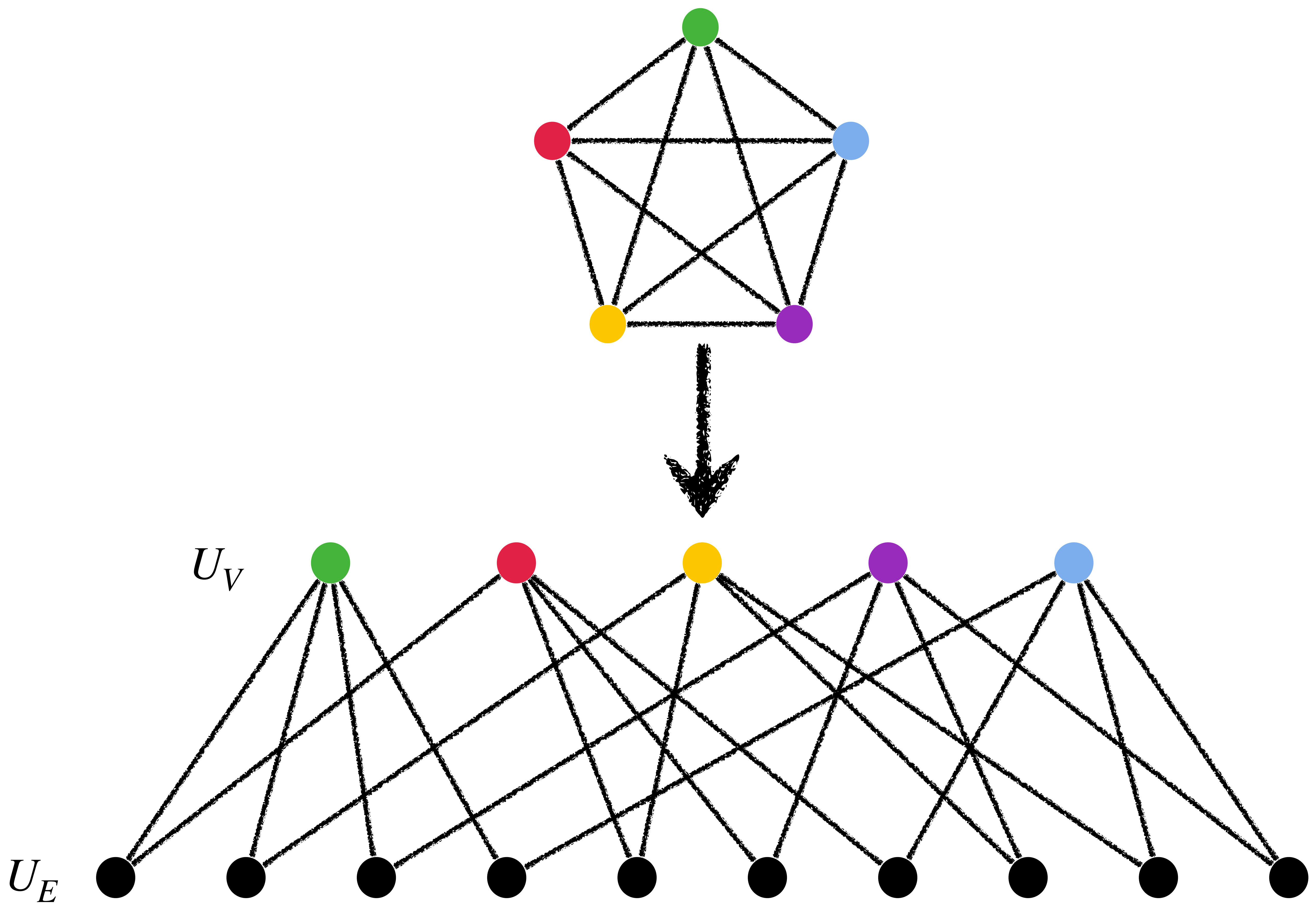


Six-Vertex Model as Holant Problem

Edge-vertex incidence graph

bipartite graph $G' = (U_E, U_V, E')$

edge $(u_e, u_v) \in E'$ iff $e \in E(v)$



Bipartite Holant problem

when $G = (U, V, E)$ is a bipartite graph

mapping π

- assigns to each vertex $u \in U$ an $f_u \in \mathcal{F}$
- assigns to each vertex $v \in V$ an $f_v \in \mathcal{G}$
- a linear order of the incident edges at v and u.

the problem is written as $\text{Holant}_\Omega(\mathcal{F} \mid \mathcal{G})$

Six-Vertex Model as Holant Problem

Edge-vertex incidence graph

bipartite graph $G' = (U_E, U_V, E')$

edge $(u_e, u_v) \in E'$ iff $e \in E(v)$

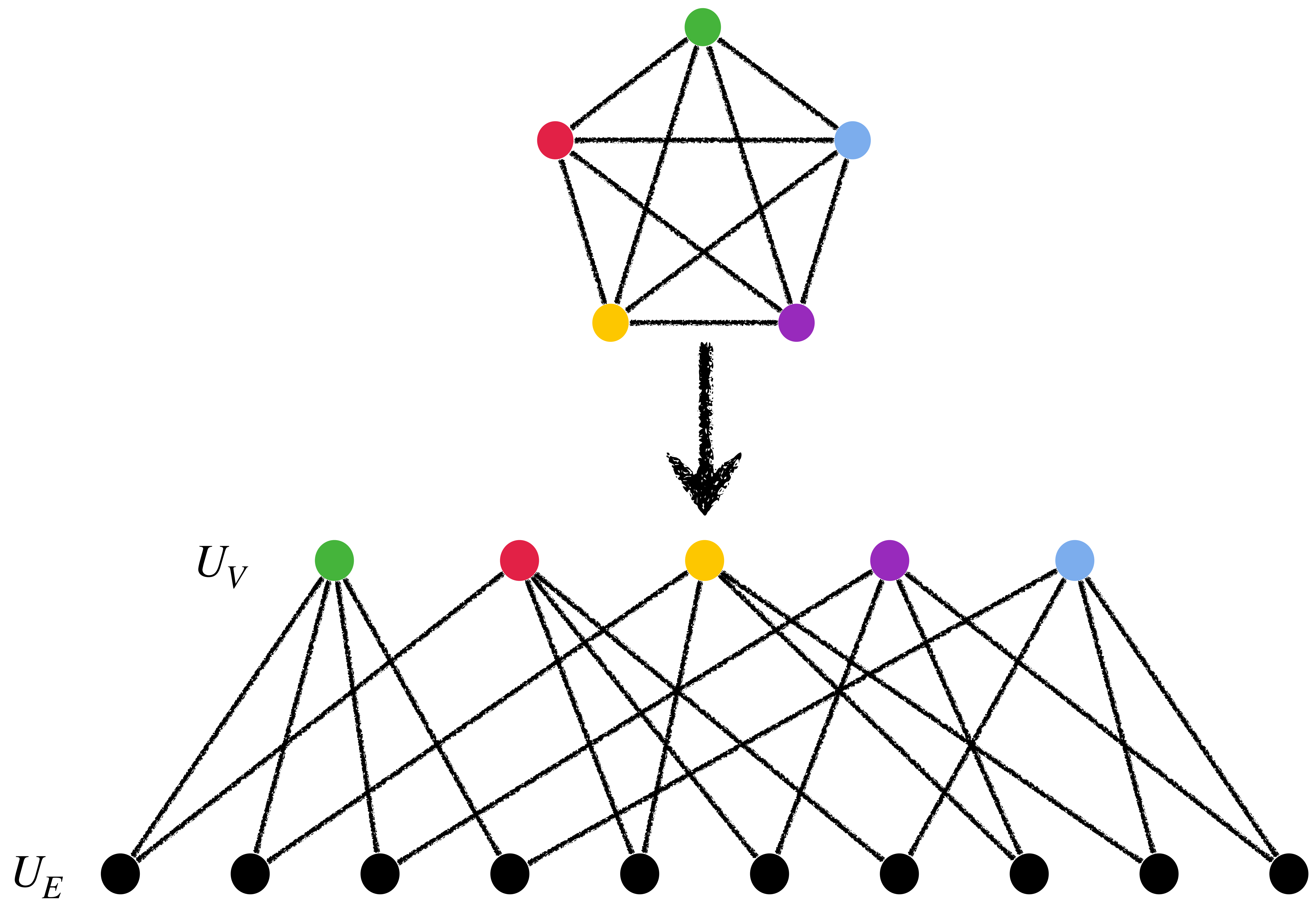
Bipartite Holant problem

Assign $f_{6\mathrm{V}}$ to each $u_v \in U_V$, where

$$f_{6\mathrm{V}}\left(\begin{bmatrix} 0000 & 0010 & 0001 & 0011 \\ 0100 & 0110 & 0101 & 0111 \\ 1000 & 1010 & 1001 & 1011 \\ 1100 & 1110 & 1101 & 1111 \end{bmatrix}\right) = \begin{bmatrix} 0 & 0 & 0 & a \\ 0 & b & c & 0 \\ 0 & c & b & 0 \\ a & 0 & 0 & 0 \end{bmatrix}$$

Assign $\neq_2$ to each $u_e \in U_E$, where

$$\neq_2\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$



Six-Vertex Model as Holant Problem

Edge-vertex incidence graph

bipartite graph $G' = (U_E, U_V, E')$

edge $(u_e, u_v) \in E'$ iff $e \in E(v)$

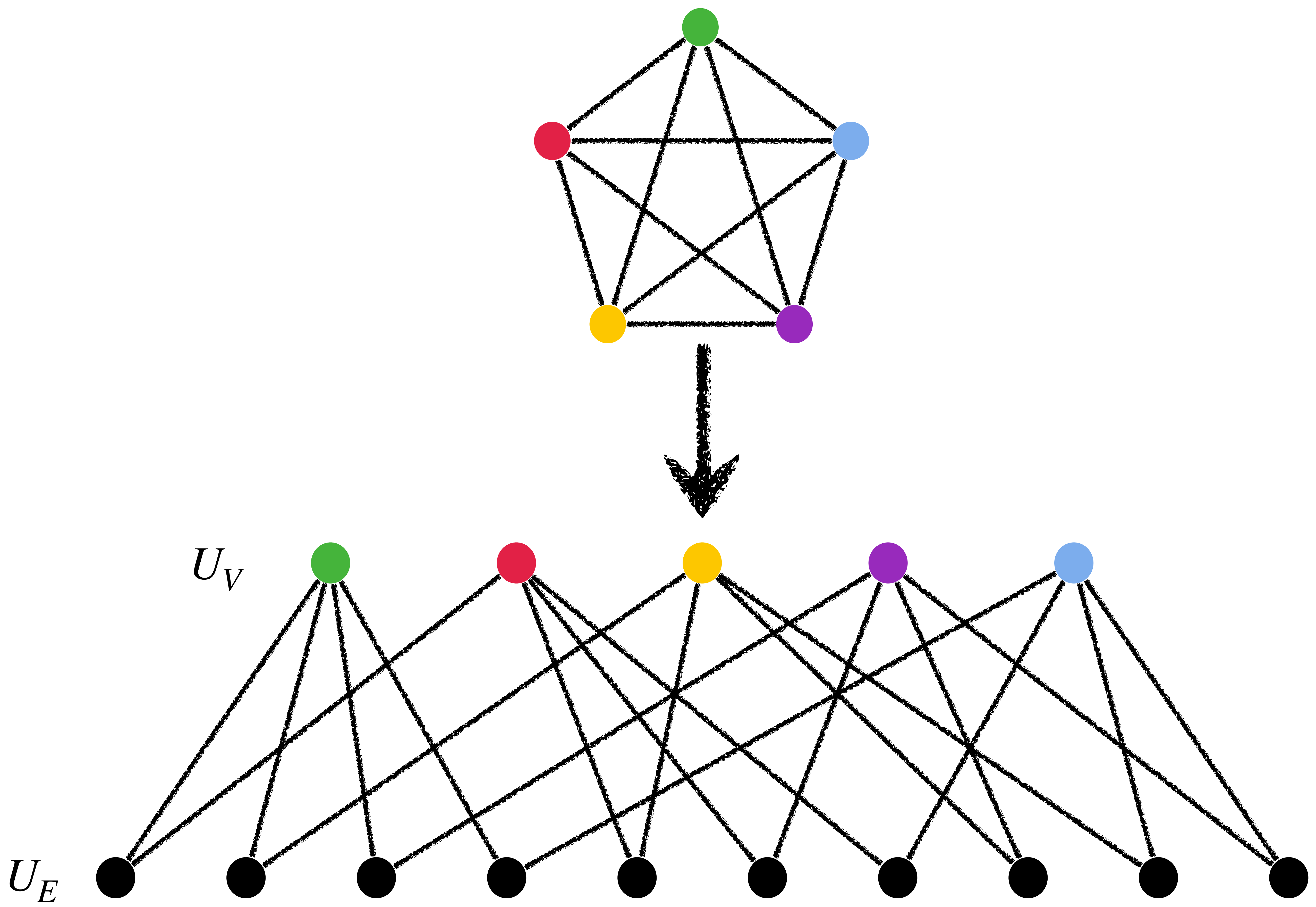
Bipartite Holant problem

Assign f_{6V} to each $u_v \in U_V$, where

$$f_{6V}\left(\begin{bmatrix} 0000 & 0010 & 0001 & 0011 \\ 0100 & 0110 & 0101 & 0111 \\ 1000 & 1010 & 1001 & 1011 \\ 1100 & 1110 & 1101 & 1111 \end{bmatrix}\right) = \begin{bmatrix} 0 & 0 & 0 & a \\ 0 & b & c & 0 \\ 0 & c & b & 0 \\ a & 0 & 0 & 0 \end{bmatrix}$$

Assign $\neq_2$ to each $u_e \in U_E$, where

$$\neq_2\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$



$$Z_{6V}(G; a, b, c) = \text{Hoalnt}_{\Omega}(\neq_2 | f_{6V})$$

Exact Computational Complexity

Exact Computational Complexity

For #EulerianOrientation (unweighted case):

It is #P-complete on

- general Eulerian graphs [Mihail and Winkler, 1992]
- even degree regular graphs [Huang and Lu, 2012]
- planar 4-regular graphs [Guo and Williams, 2013]

Exact Computational Complexity

For #EulerianOrientation (unweighted case):

It is #P-complete on

- general Eulerian graphs [Mihail and Winkler, 1992]
- even degree regular graphs [Huang and Lu, 2012]
- planar 4-regular graphs [Guo and Williams, 2013]

For six-vertex model (with complex weights):

- dichotomy for general 4-regular graphs [Cai, Fu, and Xia, 2018]
- *trichotomy* for planar 4-regular graphs [Cai, Fu, and Shao, 2017]

Exact Computational Complexity

For #EulerianOrientation (unweighted case):

It is #P-complete on

- general Eulerian graphs [Mihail and Winkler, 1992]
- even degree regular graphs [Huang and Lu, 2012]
- planar 4-regular graphs [Guo and Williams, 2013]

For six-vertex model (with complex weights):

- dichotomy for general 4-regular graphs [Cai, Fu, and Xia, 2018]
- *trichotomy* for planar 4-regular graphs [Cai, Fu, and Shao, 2017]

- tractable
- traceable on planar graphs, but #P-hard on general graphs
- #P-hard even on planar graphs

Approximate Counting and Sampling

Approximate Counting and Sampling

A series of results for unweighted case:

[Mihail and Winkler, 1992; Luby, Randall, and Sinclair, 1995;
Randall and Tetali, 1998; Goldberg, Martin, and Paterson, 2002]

Approximate Counting and Sampling

A series of results for unweighted case:

[Mihail and Winkler, 1992; Luby, Randall, and Sinclair, 1995; Randall and Tetali, 1998; Goldberg, Martin, and Paterson, 2002]

For weighted (in $\mathbb{R}^+$) case: [Cai, Liu, and Lu, 2017]

There is ***an FPRAS*** for $Z_{6V}(G; a, b, c)$ if $a^2 \leq b^2 + c^2, b^2 \leq a^2 + c^2$, and $c^2 \leq a^2 + b^2$.

There is ***no FPRAS*** for $Z_{6V}(G; a, b, c)$ if $a > b + c$ or $b > a + c$ or $c > a + b$ unless $\mathrm{RP} = \mathrm{NP}$.

Approximate Counting and Sampling

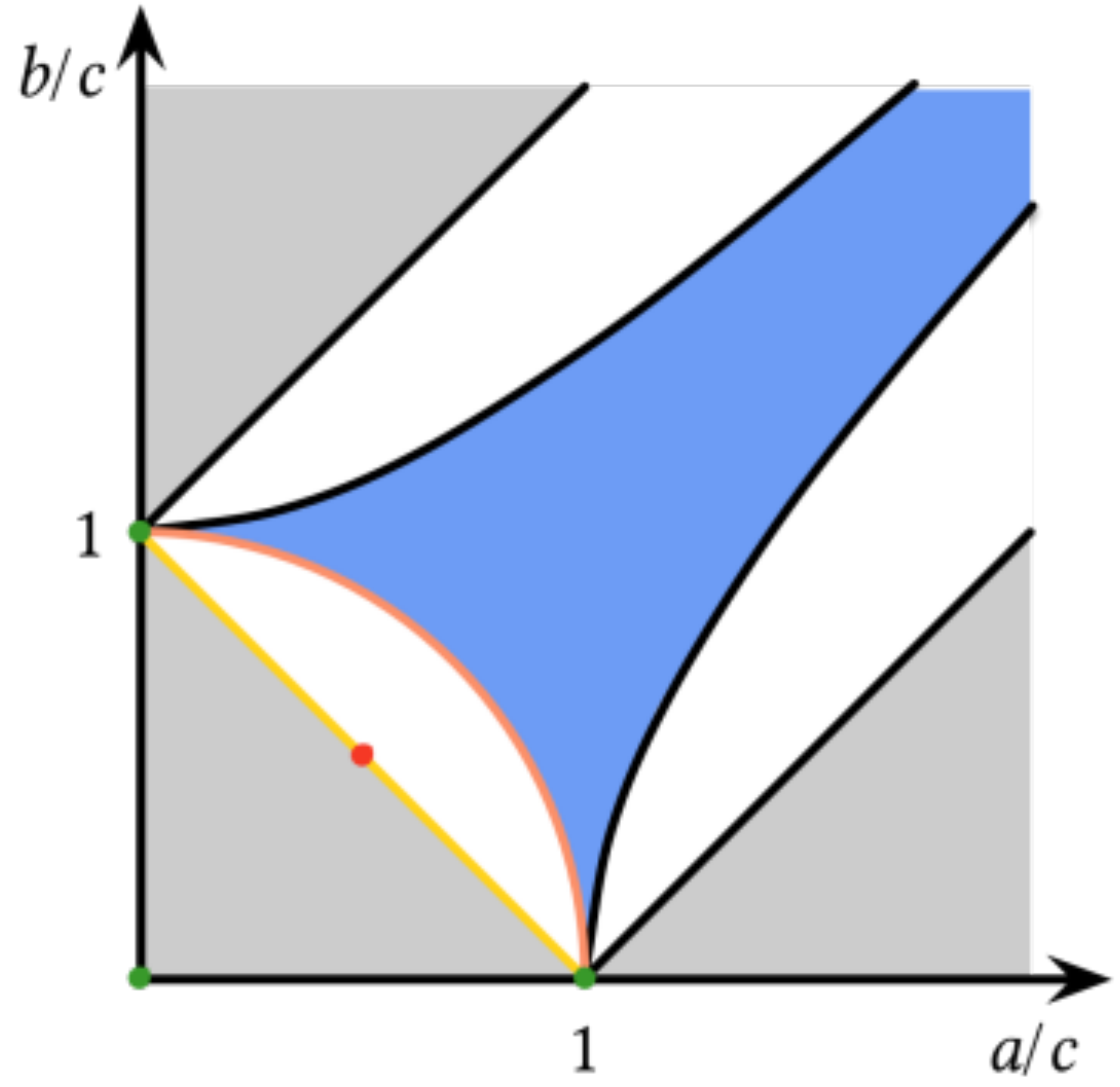
A series of results for unweighted case:

[Mihail and Winkler, 1992; Luby, Randall, and Sinclair, 1995; Randall and Tetali, 1998; Goldberg, Martin, and Paterson, 2002]

For weighted (in $\mathbb{R}^+$) case: [Cai, Liu, and Lu, 2017]

There is ***an FPRAS*** for $Z_{6V}(G; a, b, c)$ if $a^2 \leq b^2 + c^2, b^2 \leq a^2 + c^2$, and $c^2 \leq a^2 + b^2$.

There is ***no FPRAS*** for $Z_{6V}(G; a, b, c)$ if $a > b + c$ or $b > a + c$ or $c > a + b$ unless RP $=$ NP.



Approximate Counting and Sampling

A series of results for unweighted case:

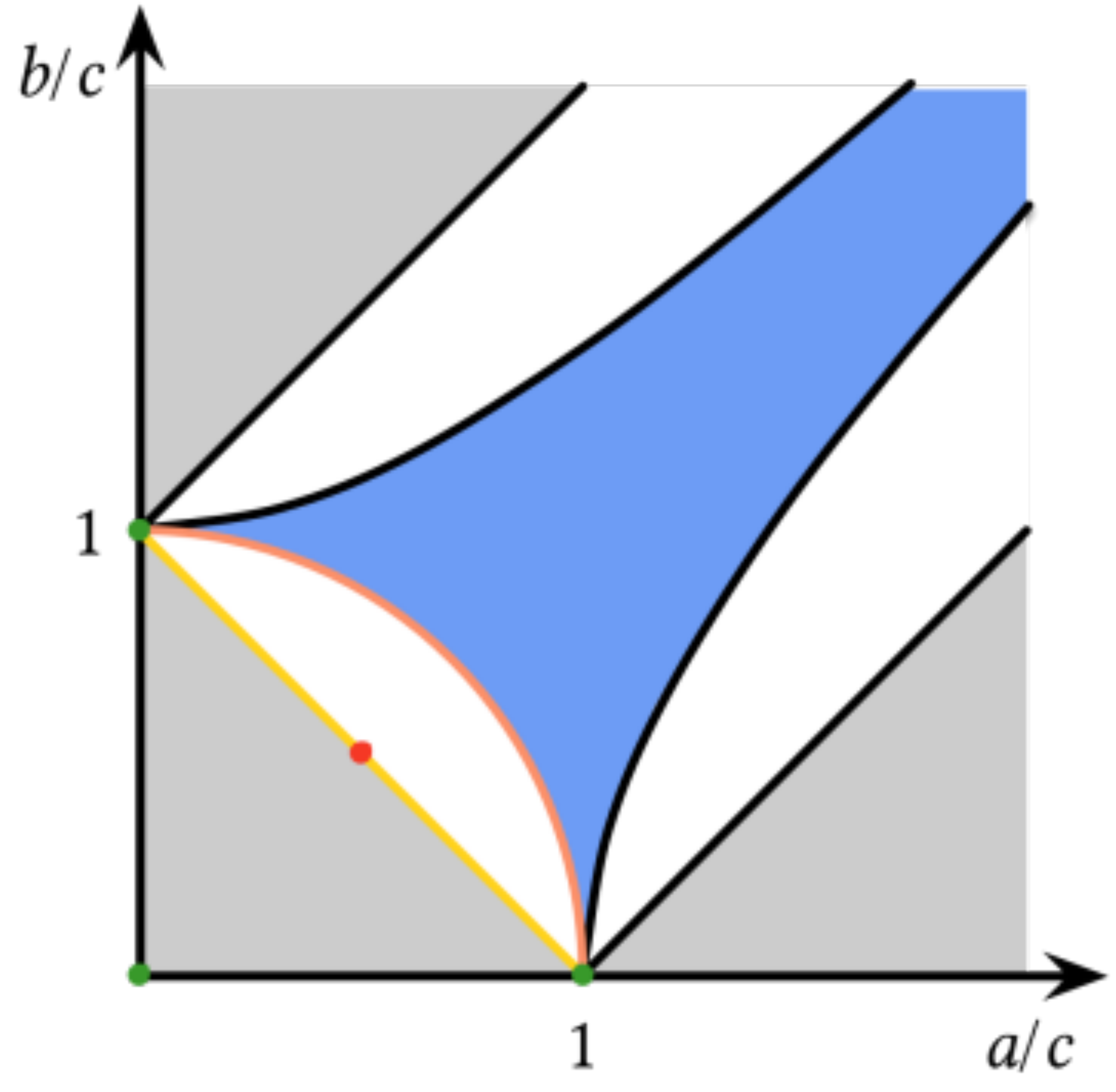
[Mihail and Winkler, 1992; Luby, Randall, and Sinclair, 1995; Randall and Tetali, 1998; Goldberg, Martin, and Paterson, 2002]

For weighted (in $\mathbb{R}^+$) case: [Cai, Liu, and Lu, 2017]

Windability

There is ***an FPRAS*** for $Z_{6V}(G; a, b, c)$ if $a^2 \leq b^2 + c^2, b^2 \leq a^2 + c^2$, and $c^2 \leq a^2 + b^2$.

There is ***no FPRAS*** for $Z_{6V}(G; a, b, c)$ if $a > b + c$ or $b > a + c$ or $c > a + b$ unless $\mathrm{RP} = \mathrm{NP}$.



Phase Transition

Phase Transition

In statistical physics, the six-vertex model exhibits different macroscopic behaviors under different parameters (a, b, c) (due to the change in temperature).

$$\Delta = \frac{a^2 + b^2 - c^2}{2ab}$$

Phase Transition

In statistical physics, the six-vertex model exhibits different macroscopic behaviors under different parameters (a, b, c) (due to the change in temperature).

$$\Delta = \frac{a^2 + b^2 - c^2}{2ab}$$

Ferroelectric Phase: $\Delta > 1$

$\implies a > b + c$ or $b > a + c$

Anti-ferroelectric Phase: $\Delta < -1$

$\implies c > a + b$

Disorder Phase: $-1 < \Delta < 1$

$\implies c \leq a + b, b \leq a + c,$ and $a \leq b + c$

Phase Transition

In statistical physics, the six-vertex model exhibits different macroscopic behaviors under different parameters (a, b, c) (due to the change in temperature).

$$\Delta = \frac{a^2 + b^2 - c^2}{2ab}$$

Ferroelectric Phase: $\Delta > 1$

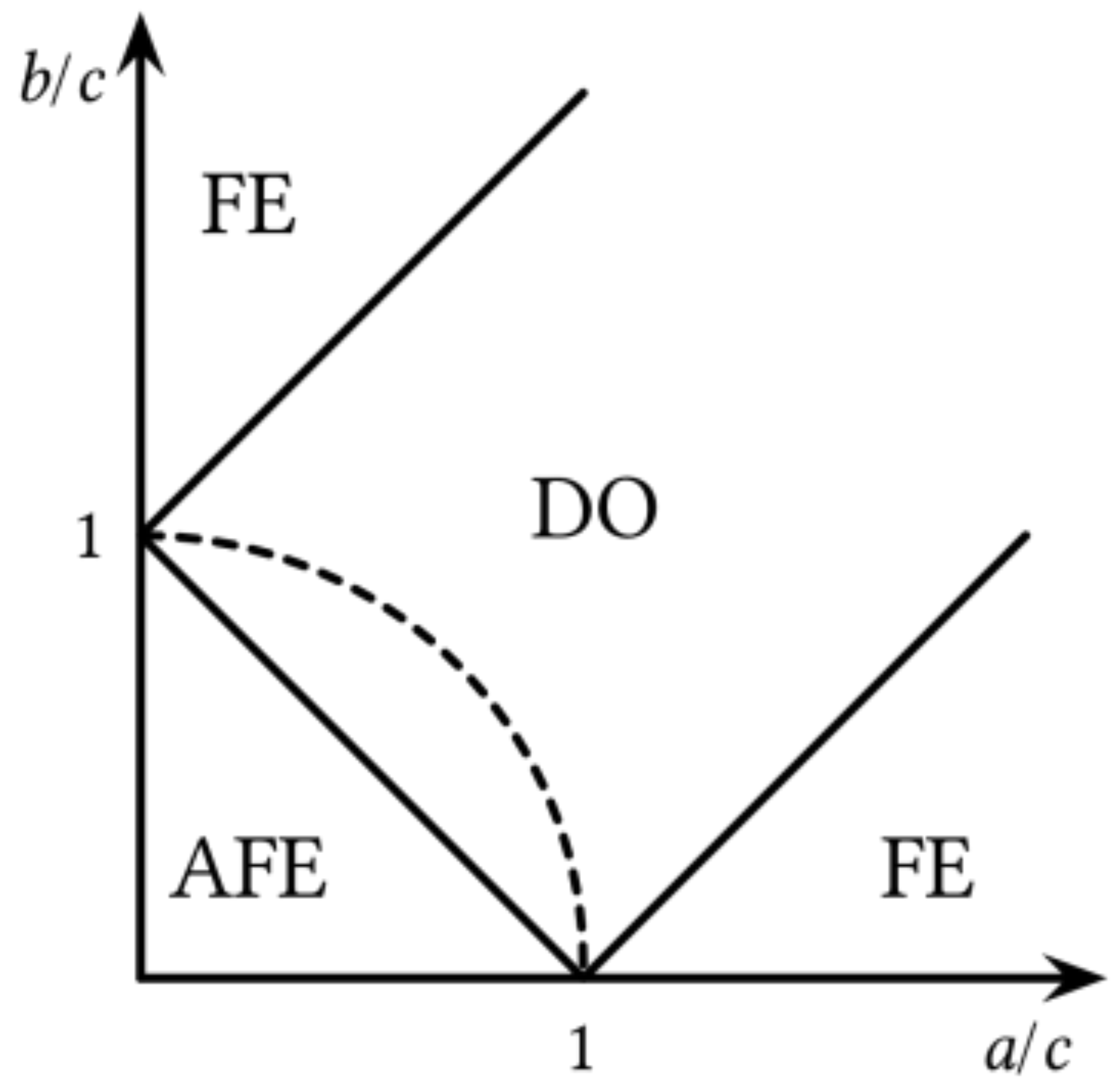
$$\Longrightarrow a > b + c \text{ or } b > a + c$$

Anti-ferroelectric Phase: $\Delta < -1$

$$\Longrightarrow c > a + b$$

Disorder Phase: $-1 < \Delta < 1$

$$\Longrightarrow c \leq a + b, b \leq a + c, \text{ and } a \leq b + c$$



Phase Transition

In statistical physics, the six-vertex model exhibits different macroscopic behaviors under different parameters (a, b, c) (due to the change in temperature).

$$\Delta = \frac{a^2 + b^2 - c^2}{2ab}$$

Ferroelectric Phase: $\Delta > 1$

$\implies a > b + c$ or $b > a + c$

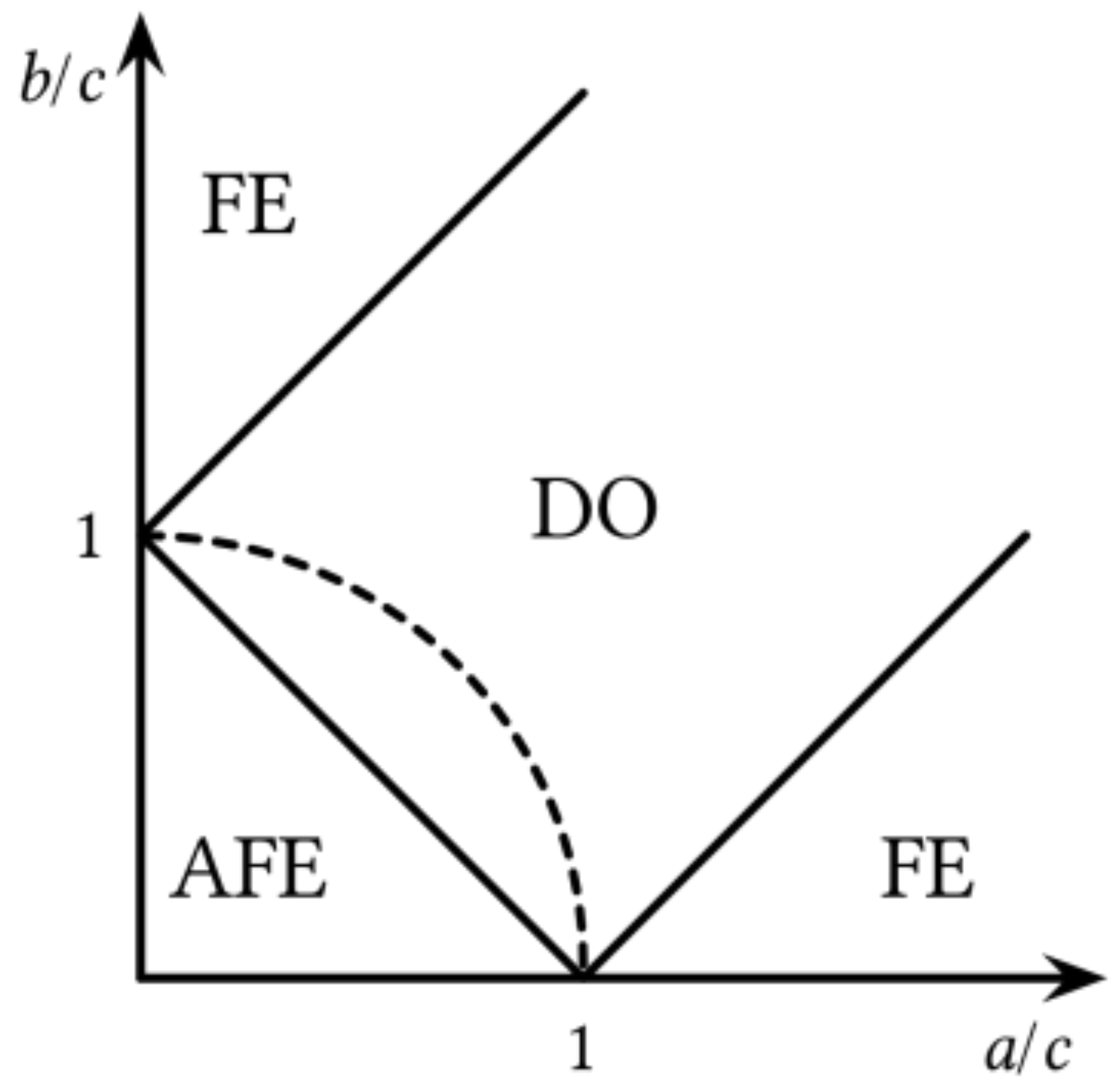
Anti-ferroelectric Phase: $\Delta < -1$

$\implies c > a + b$

Disorder Phase: $-1 < \Delta < 1$

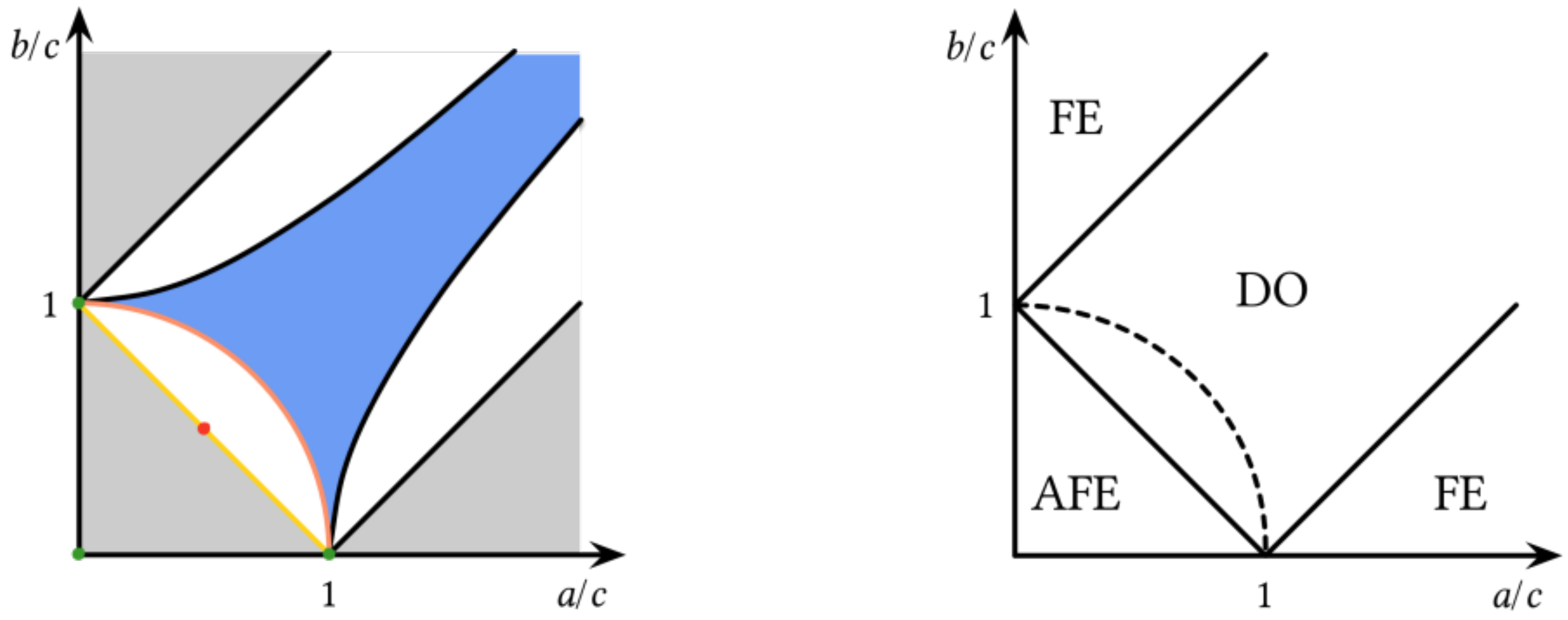
$\implies c \le a + b, b \le a + c,$ and $a \le b + c$

low temperature regime



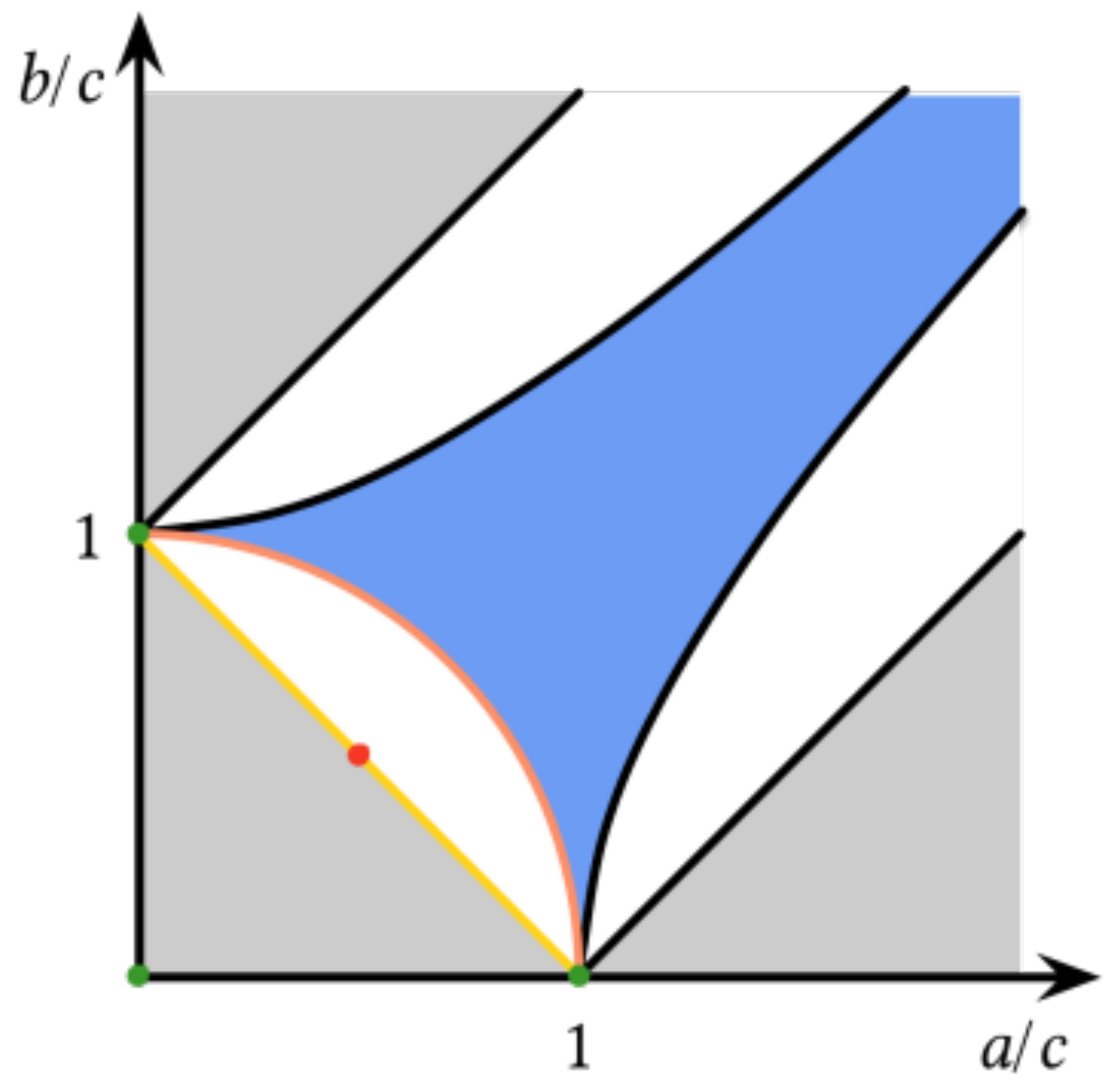
high temperature regime

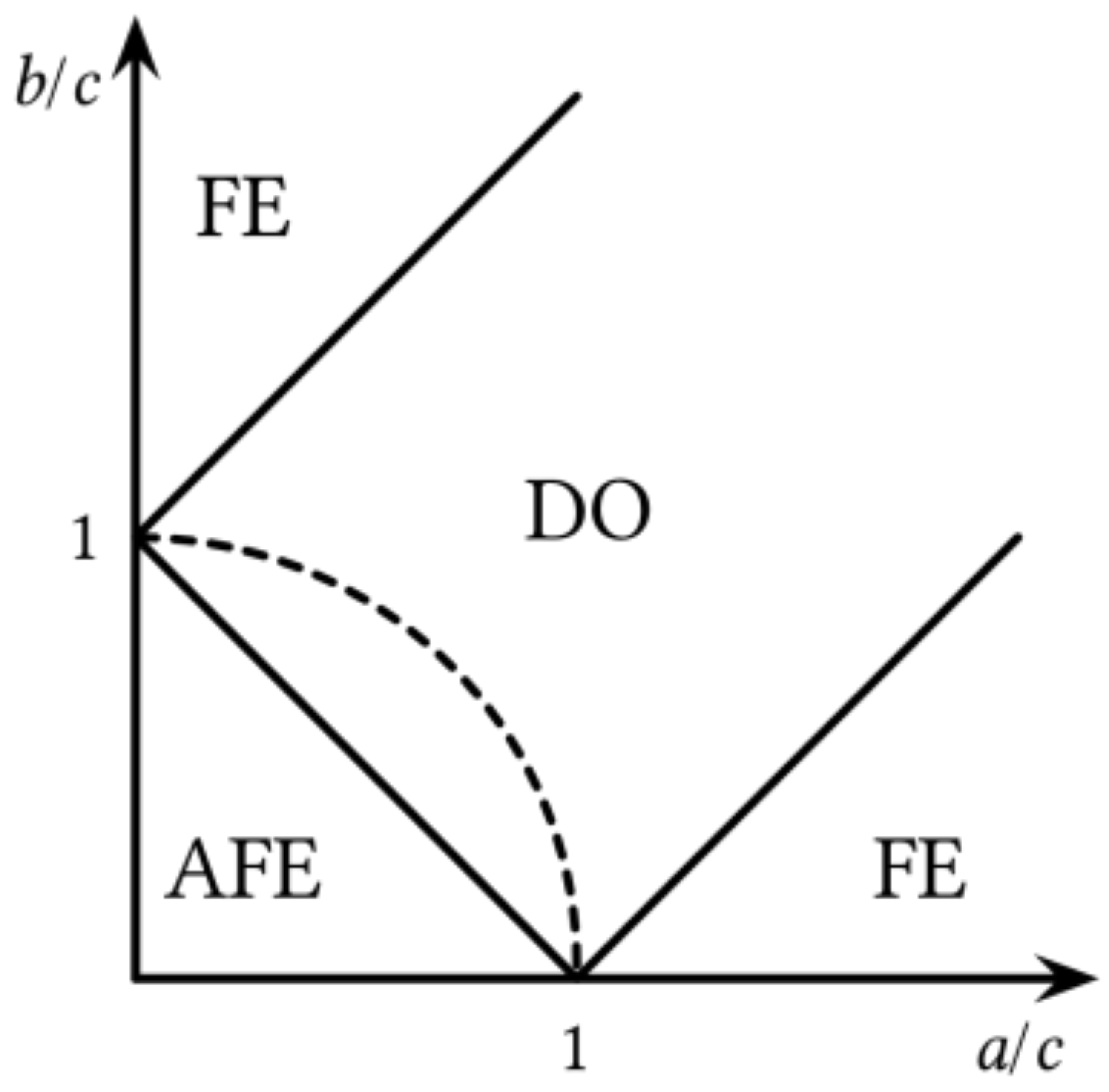
Approximability v.s. Phase Transition



Approximability of the six-vertex model (roughly) conforms the phase transition phenomenon in physics!

Approximability v.s. Phase Transition

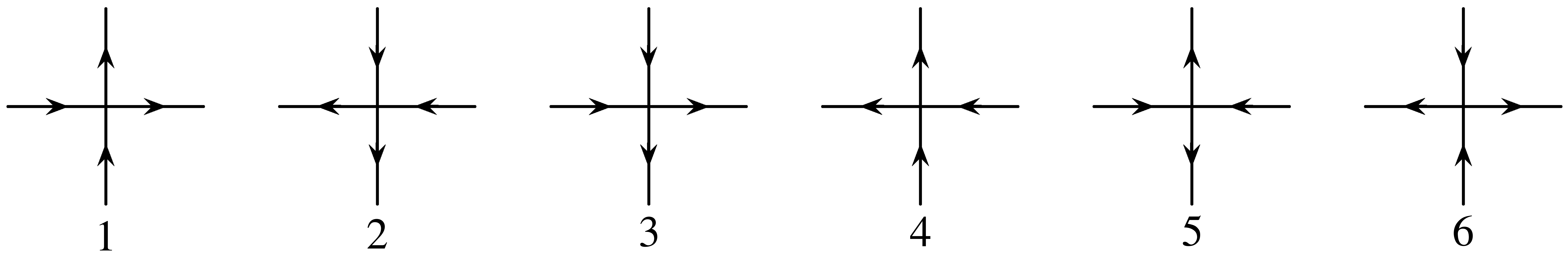




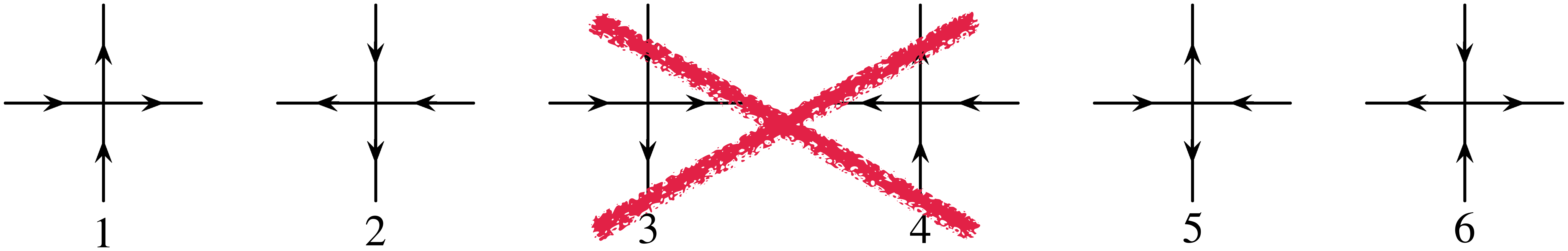
Approximability of the six-vertex model (roughly) conforms the phase transition phenomenon in physics!

How to bridge the gap?

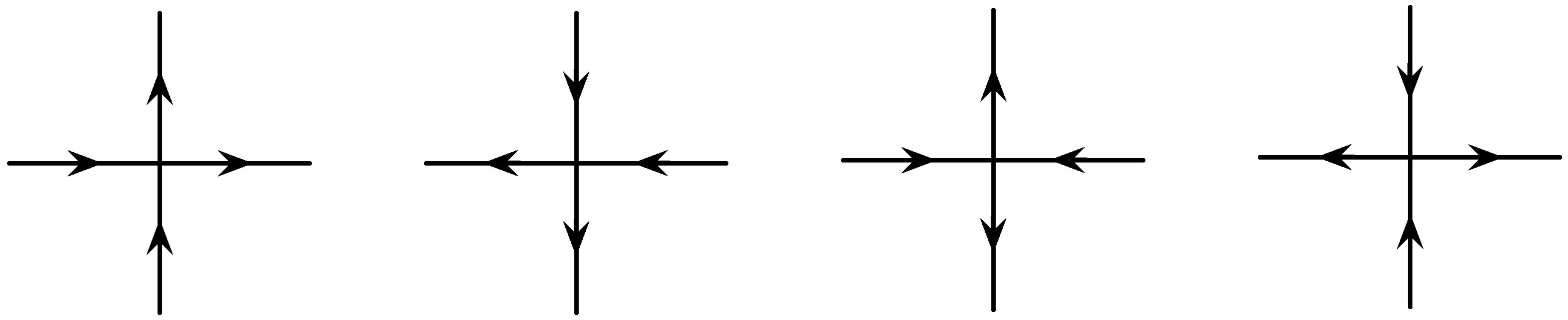
Attempt: Four-Vertex Model



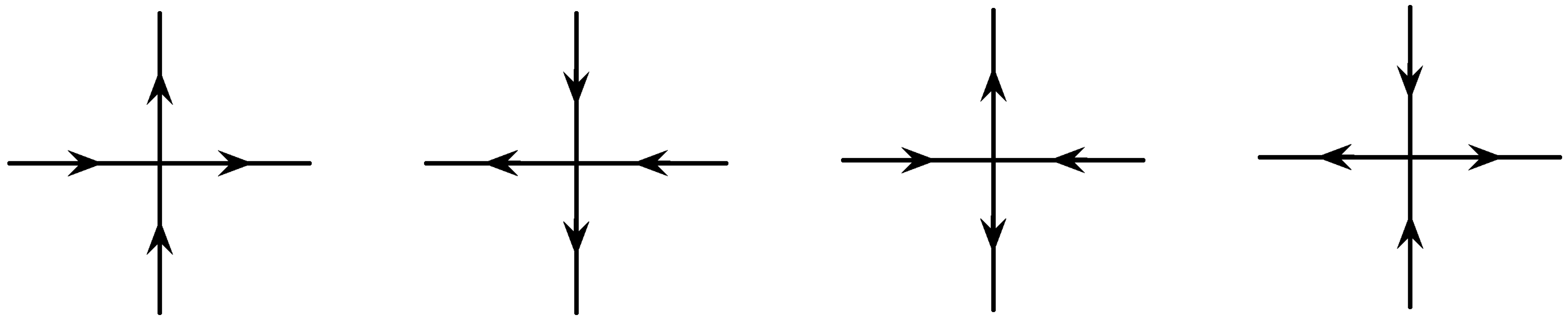
Attempt: Four-Vertex Model



Attempt: Four-Vertex Model



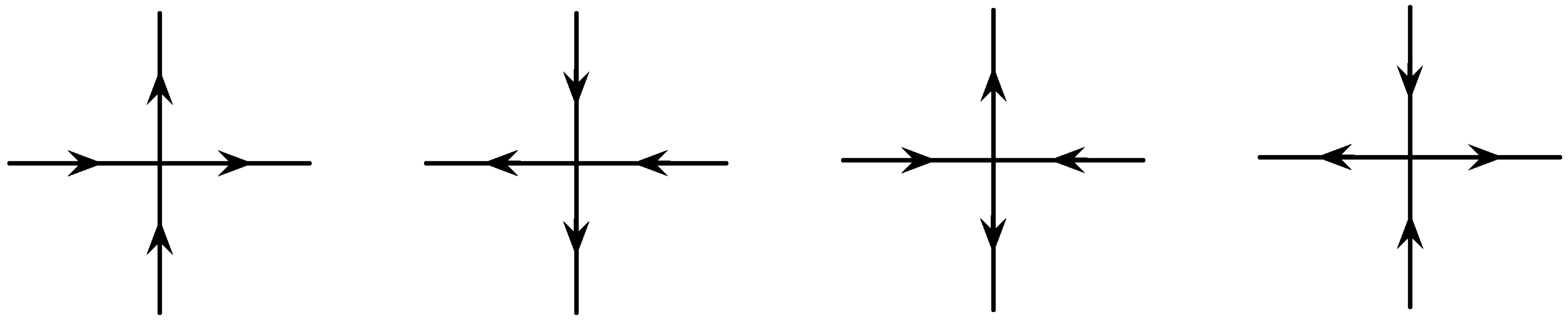
Attempt: Four-Vertex Model



normalized weight	β	β	1	1

(WLOG, $\beta > 1$)

Attempt: Four-Vertex Model



normalized weight	β	β	1	1

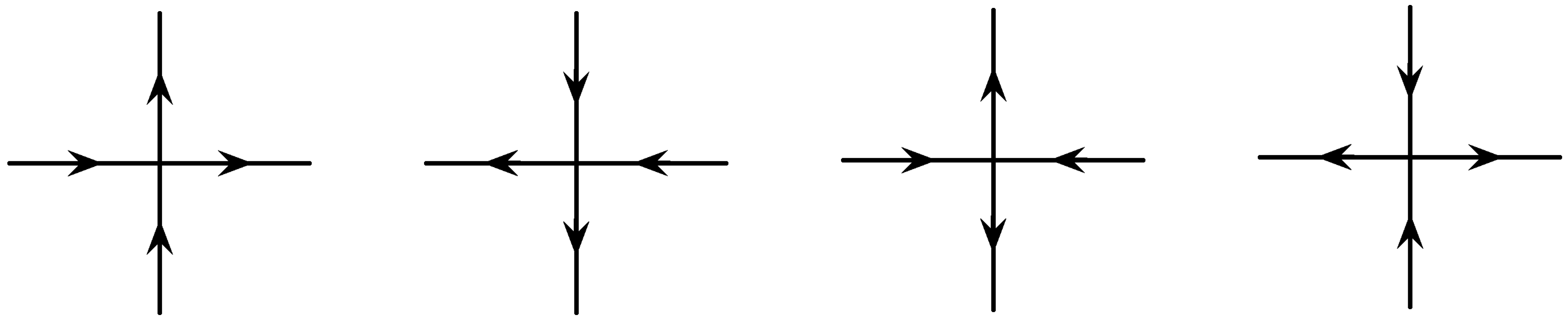
(WLOG, $\beta > 1$)

Goal: compute

$$Z_{4\text{V}}(G;\beta) = Z_{6\text{V}}(G;\beta,0,1) = \text{Holant}_{\Omega}(\neq_2 \mid f_{4\text{V}})$$

$$f_{4\text{V}}\left(\begin{bmatrix} 0000 & 0010 & 0001 & 0011 \\ 0100 & 0110 & 0101 & 0111 \\ 1000 & 1010 & 1001 & 1011 \\ 1100 & 1110 & 1101 & 1111 \end{bmatrix}\right) = \begin{bmatrix} 0 & 0 & 0 & \beta \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \beta & 0 & 0 & 0 \end{bmatrix}$$

Attempt: Four-Vertex Model



normalized weight	β	β	1	1

(WLOG, $\beta > 1$)

Goal: compute

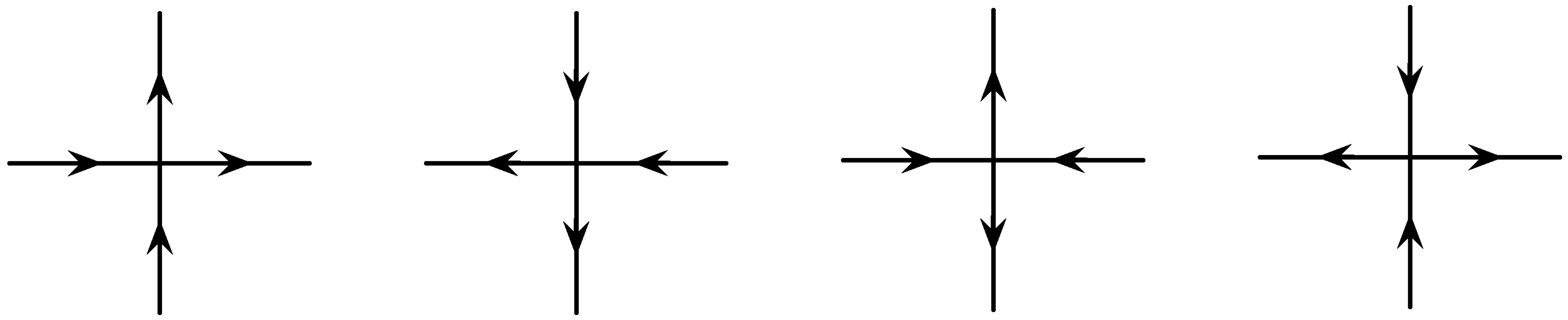
$$Z_{4V}(G;\beta) = Z_{6V}(G;\beta,0,1) = \text{Hoalnt}_{\Omega}(\neq_2 \mid f_{4V})$$

$$f_{4V}\left(\begin{bmatrix} 0000 & 0010 & 0001 & 0011 \\ 0100 & 0110 & 0101 & 0111 \\ 1000 & 1010 & 1001 & 1011 \\ 1100 & 1110 & 1101 & 1111 \end{bmatrix}\right) = \begin{bmatrix} 0 & 0 & 0 & \beta \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \beta & 0 & 0 & 0 \end{bmatrix}$$

Known results

- Exact computation is #P-hard
- There is no FPRAS for it *in general*

Attempt: Four-Vertex Model



normalized weight β β 1 1

(WLOG, $\beta > 1$)

Goal: compute

$$Z_{4\mathrm{V}}(G;\beta) = Z_{6\mathrm{V}}(G;\beta,0,1) = \mathrm{Holant}_{\Omega}(\neq_2 \mid f_{4\mathrm{V}})$$

$$f_{4\mathrm{V}}\left(\begin{bmatrix} 0000 & 0010 & 0001 & 0011 \\ 0100 & 0110 & 0101 & 0111 \\ 1000 & 1010 & 1001 & 1011 \\ 1100 & 1110 & 1101 & 1111 \end{bmatrix}\right) = \begin{bmatrix} 0 & 0 & 0 & \beta \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \beta & 0 & 0 & 0 \end{bmatrix}$$

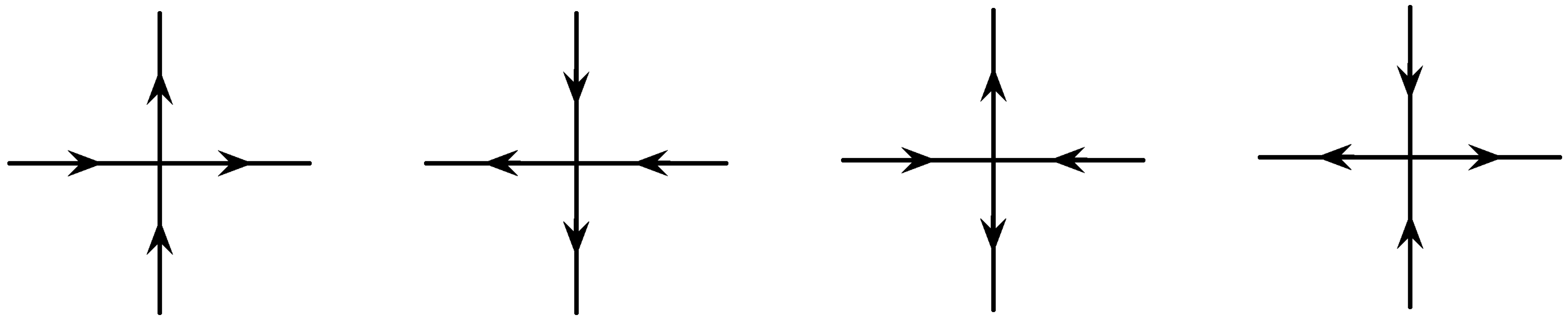
Known results

- Exact computation is #P-hard
- There is no FPRAS for it *in general*

Our result

There is an algebra criterion to decide whether it admits FPRAS

Attempt: Four-Vertex Model



normalized weight	β	β	1	1

(WLOG, $\beta > 1$)

Goal: compute

$$Z_{4V}(G;\beta) = Z_{6V}(G;\beta,0,1) = \mathrm{Hoalnt}_\Omega(\neq_2 \mid f_{4V}) \qquad f_{4V}\left(\begin{bmatrix} 0000 & 0010 & 0001 & 0011 \\ 0100 & 0110 & 0101 & 0111 \\ 1000 & 1010 & 1001 & 1011 \\ 1100 & 1110 & 1101 & 1111 \end{bmatrix}\right) = \begin{bmatrix} 0 & 0 & 0 & \beta \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \beta & 0 & 0 & 0 \end{bmatrix}$$

Unwindable

Known results

- Exact computation is #P-hard
- There is no FPRAS for it *in general*

Our result

There is an algebra criterion to decide whether it admits FPRAS

Circuit Decomposition

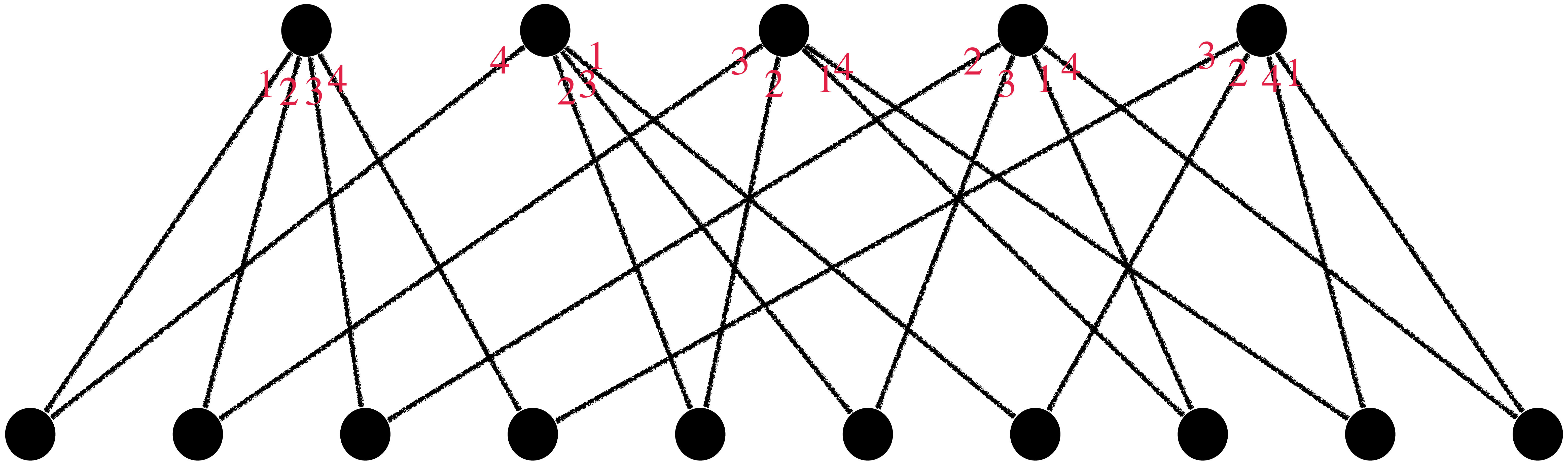
Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for $f_{4\text{V}}$ in any valid configurations

Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for $f_{4\mathrm{V}}$ in any valid configurations

Circuit decomposition

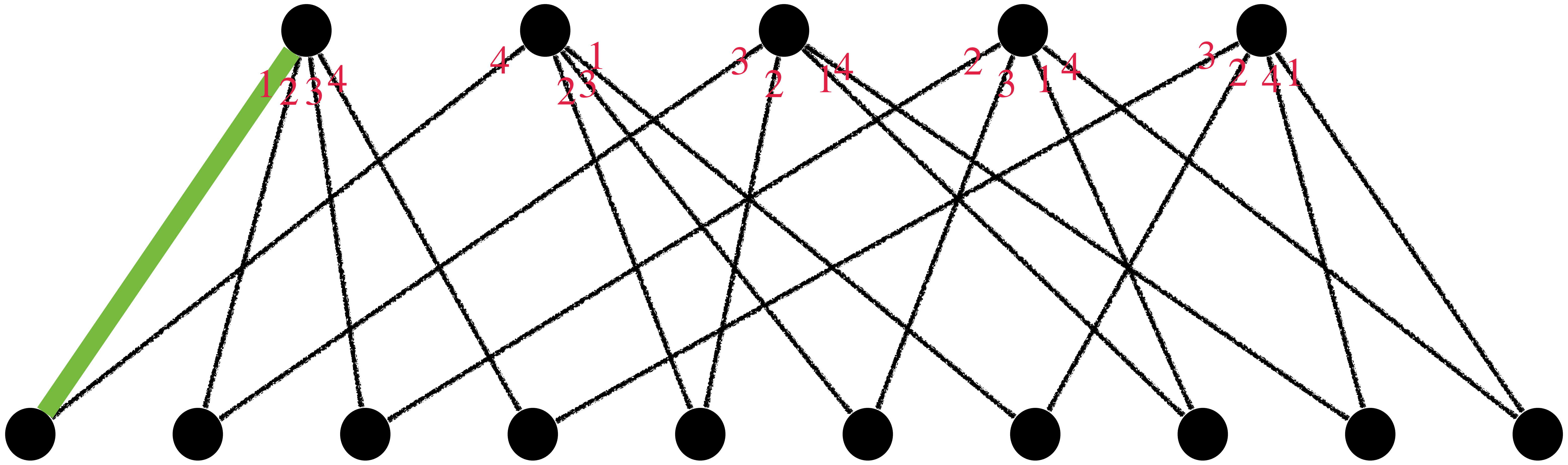


Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for $f_{4\mathrm{V}}$ in any valid configurations

Circuit decomposition

- Choose a vertex and an incident edge as the initial edge and trace a trail

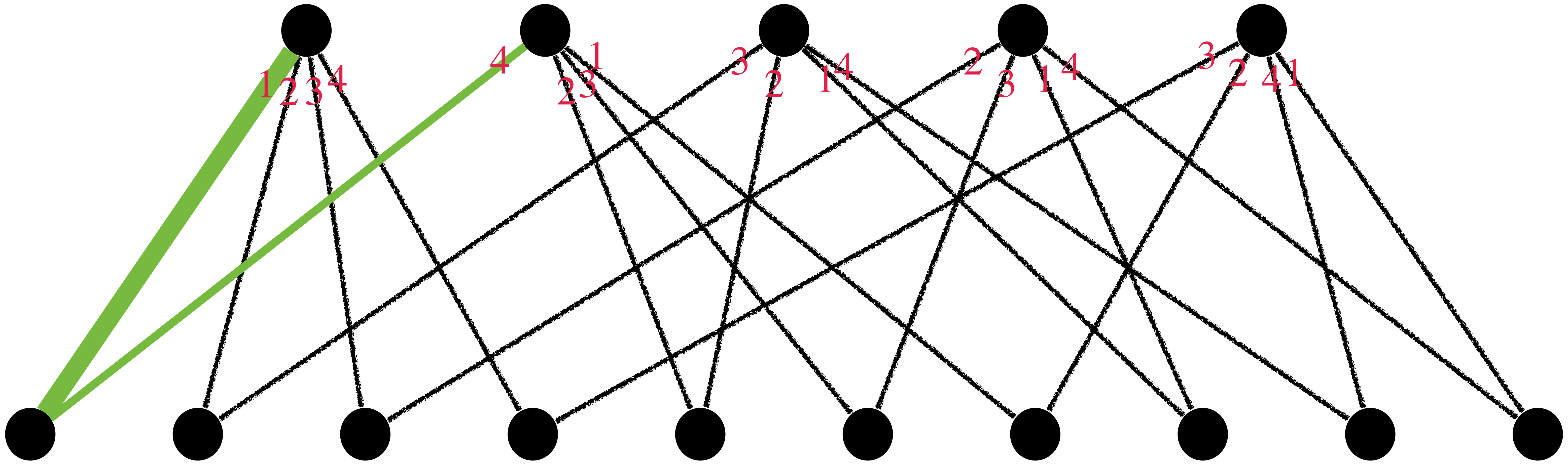


Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for $f_{4\text{V}}$ in any valid configurations

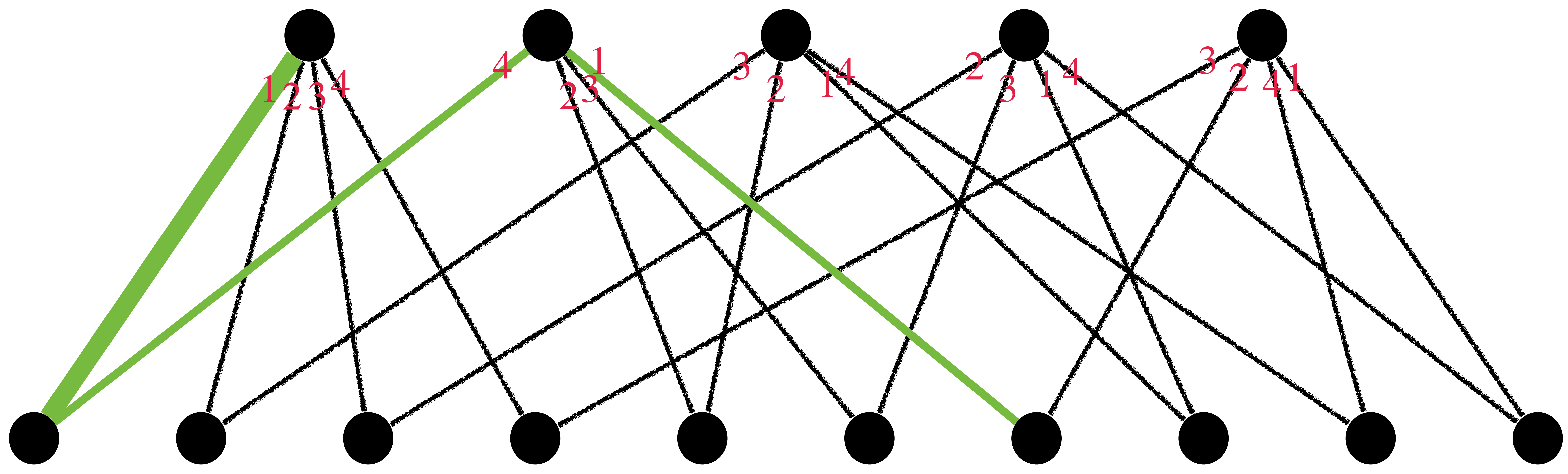
Circuit decomposition

- Choose a vertex and an incident edge as the initial edge and trace a trail
- Trail proceeds directly through vertices in U_E



Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for $f_{4\text{V}}$ in any valid configurations



Circuit decomposition

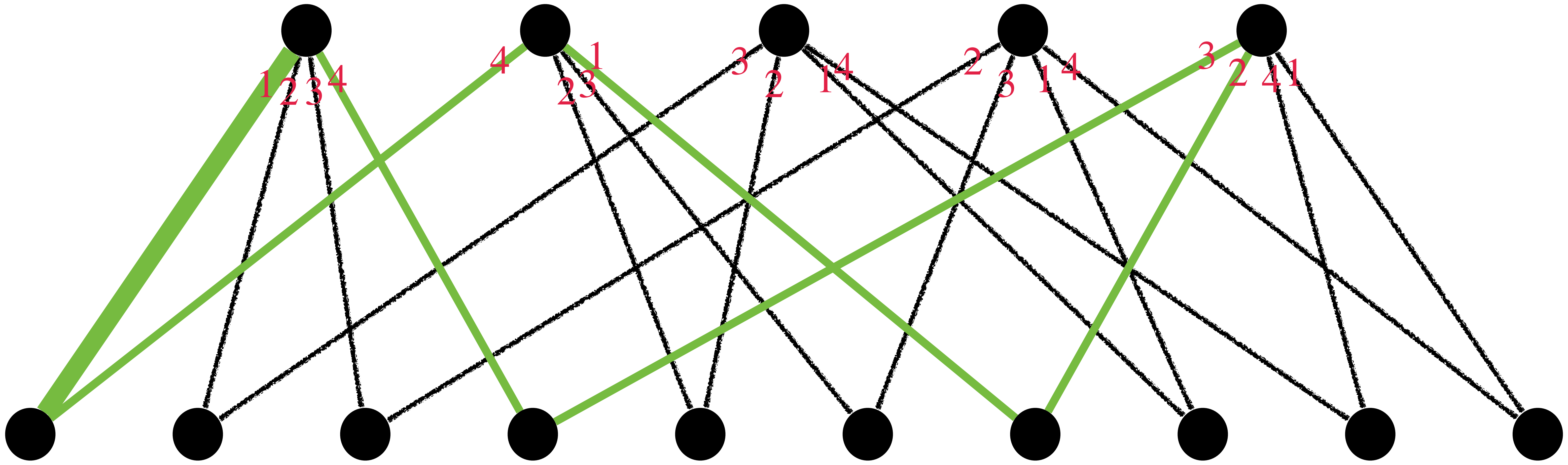
- Choose a vertex and an incident edge as the initial edge and trace a trail
- Trail proceeds directly through vertices in U_E
- Trail proceeds vertices in U_V by choosing the "opposite " variable

Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for $f_{4\mathrm{V}}$ in any valid configurations

Circuit decomposition

- Choose a vertex and an incident edge as the initial edge and trace a trail
- Trail proceeds directly through vertices in U_E
- Trail proceeds vertices in U_V by choosing the "opposite " variable

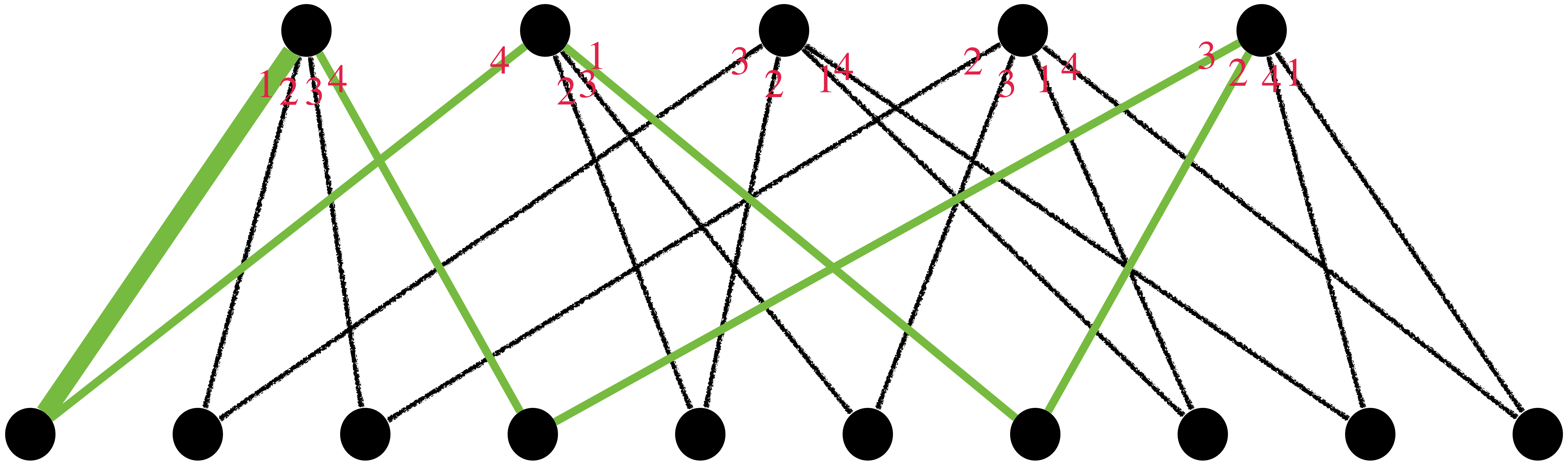


Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for $f_{4\mathrm{V}}$ in any valid configurations

Circuit decomposition

- Choose a vertex and an incident edge as the initial edge and trace a trail
- Trail proceeds directly through vertices in U_E
- Trail proceeds vertices in U_V by choosing the "opposite " variable
- When return to the starting vertex, delete the circuit and repeat the process

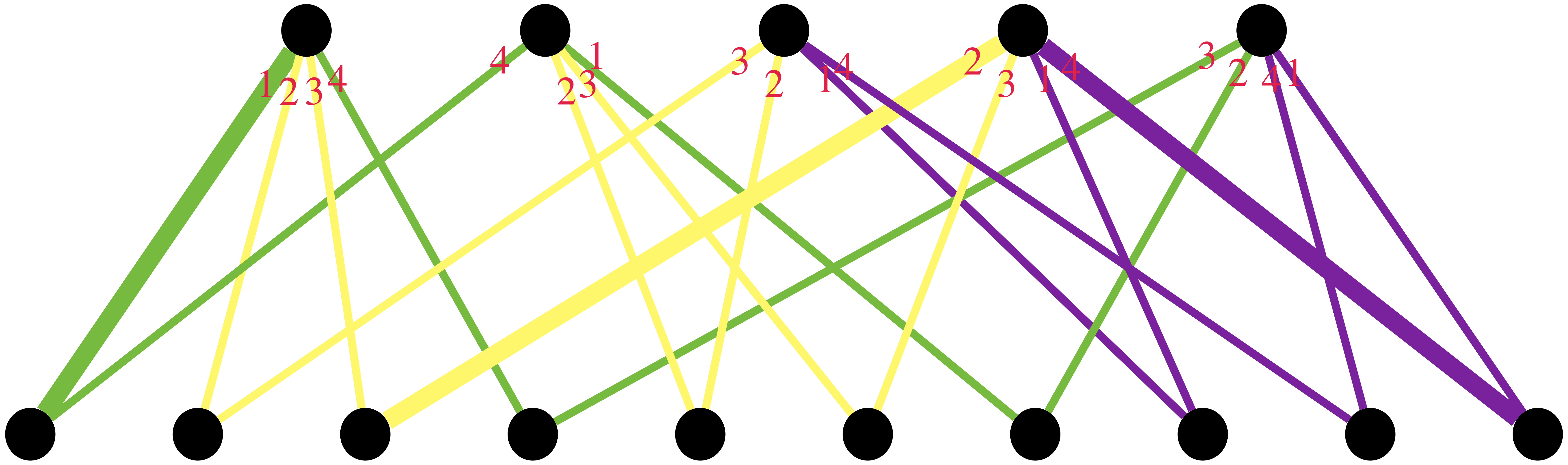


Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for $f_{4\mathrm{V}}$ in any valid configurations

Circuit decomposition

- Choose a vertex and an incident edge as the initial edge and trace a trail
- Trail proceeds directly through vertices in U_E
- Trail proceeds vertices in U_V by choosing the "opposite " variable
- When return to the starting vertex, delete the circuit and repeat the process

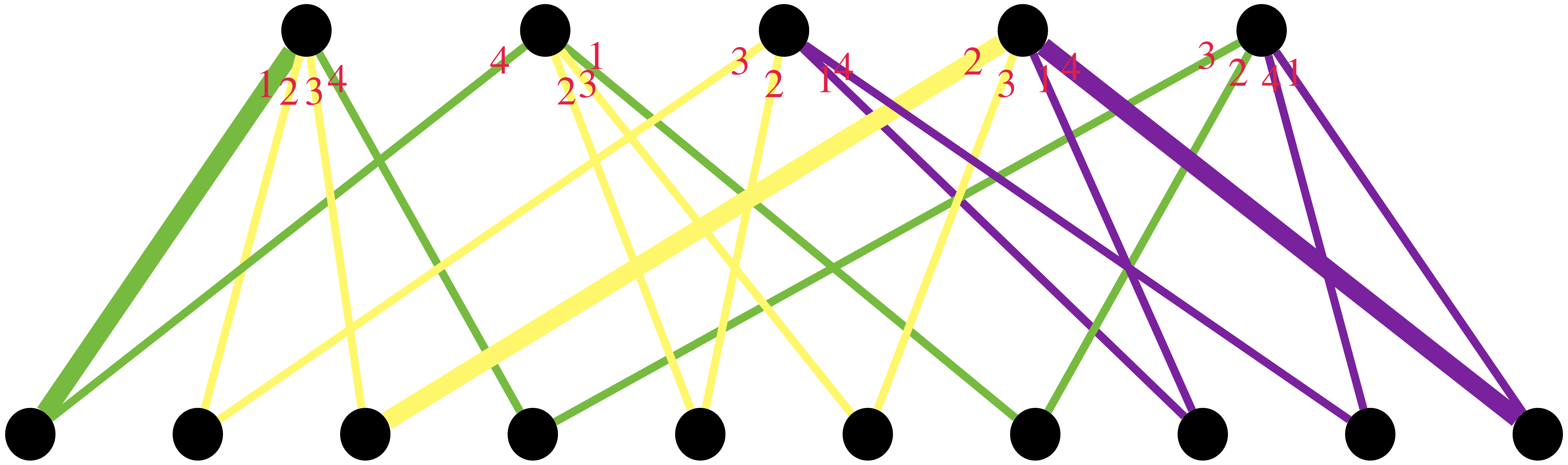


Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for $f_{4\mathrm{V}}$ in any valid configurations

Circuit decomposition

- Choose a vertex and an incident edge as the initial edge and trace a trail
- Trail proceeds directly through vertices in U_E
- Trail proceeds vertices in U_V by choosing the "opposite" variable
- When return to the starting vertex, delete the circuit and repeat the process



Graph of circuits

$$G_C = (\mathscr{C}, E_C)$$

$\mathscr{C} = \{C_1, C_2 \cdots\}$: set of circuits

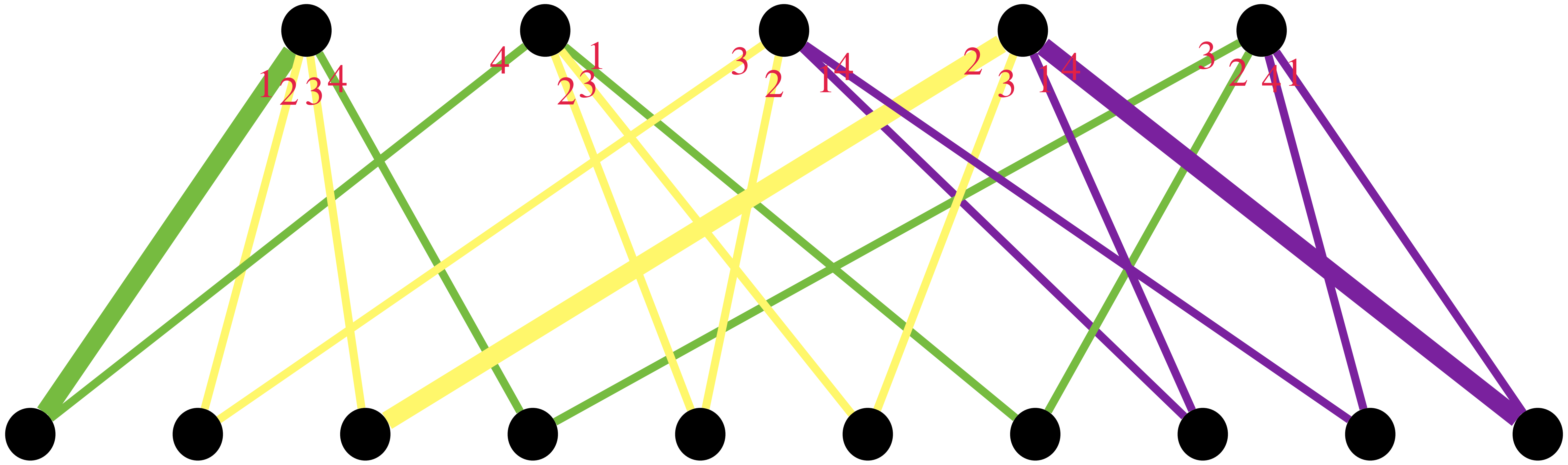
$(C_i, C_j) \in E_C$ iff C_i and C_j have common vertex

Circuit Decomposition

Key observation $x_1 \neq x_4$ and $x_2 \neq x_3$ always hold for f_{4V} in any valid configurations

Circuit decomposition

- Choose a vertex and an incident edge as the initial edge and trace a trail
- Trail proceeds directly through vertices in U_E
- Trail proceeds vertices in U_V by choosing the "opposite " variable
- When return to the starting vertex, delete the circuit and repeat the process

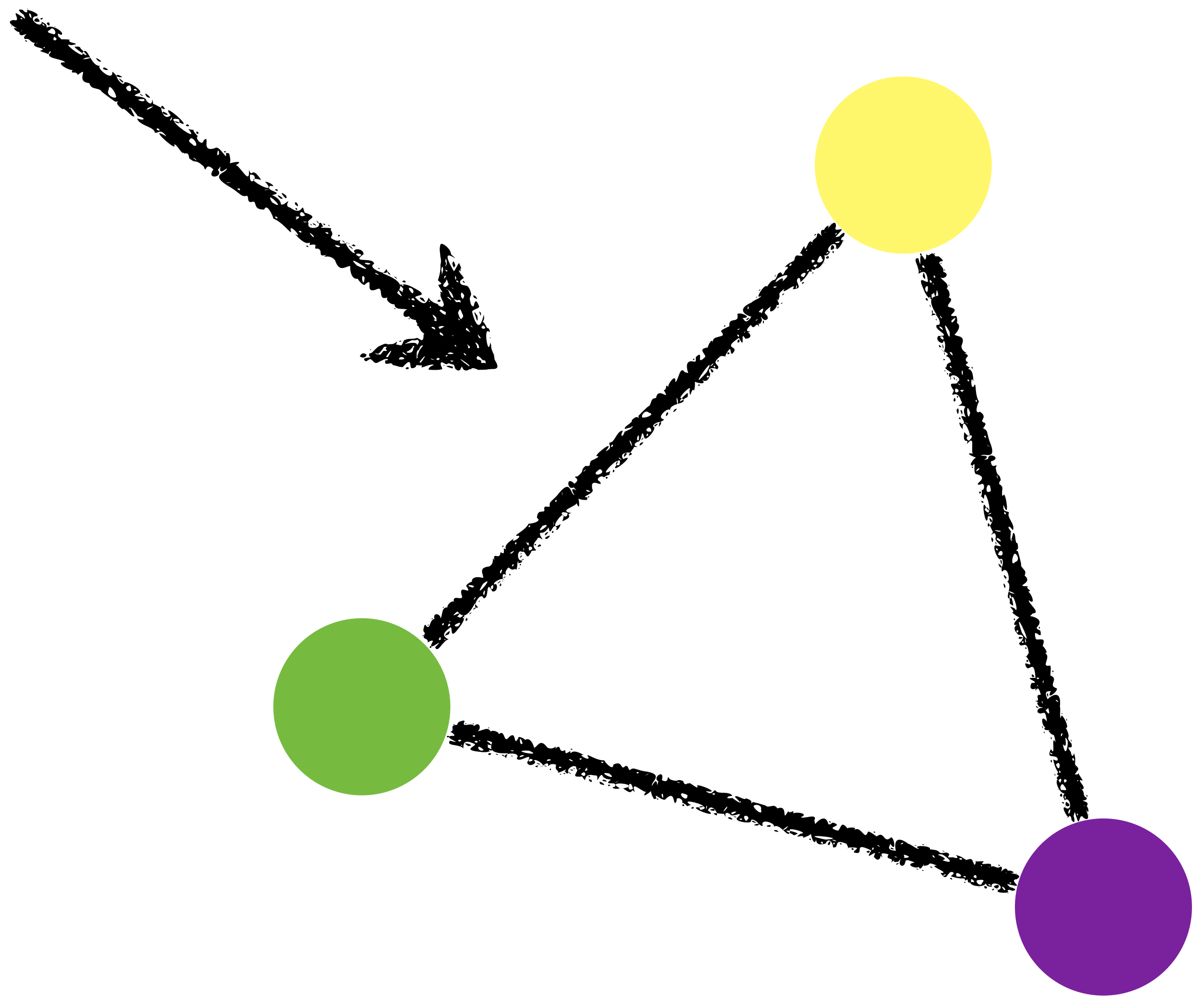


Graph of circuits

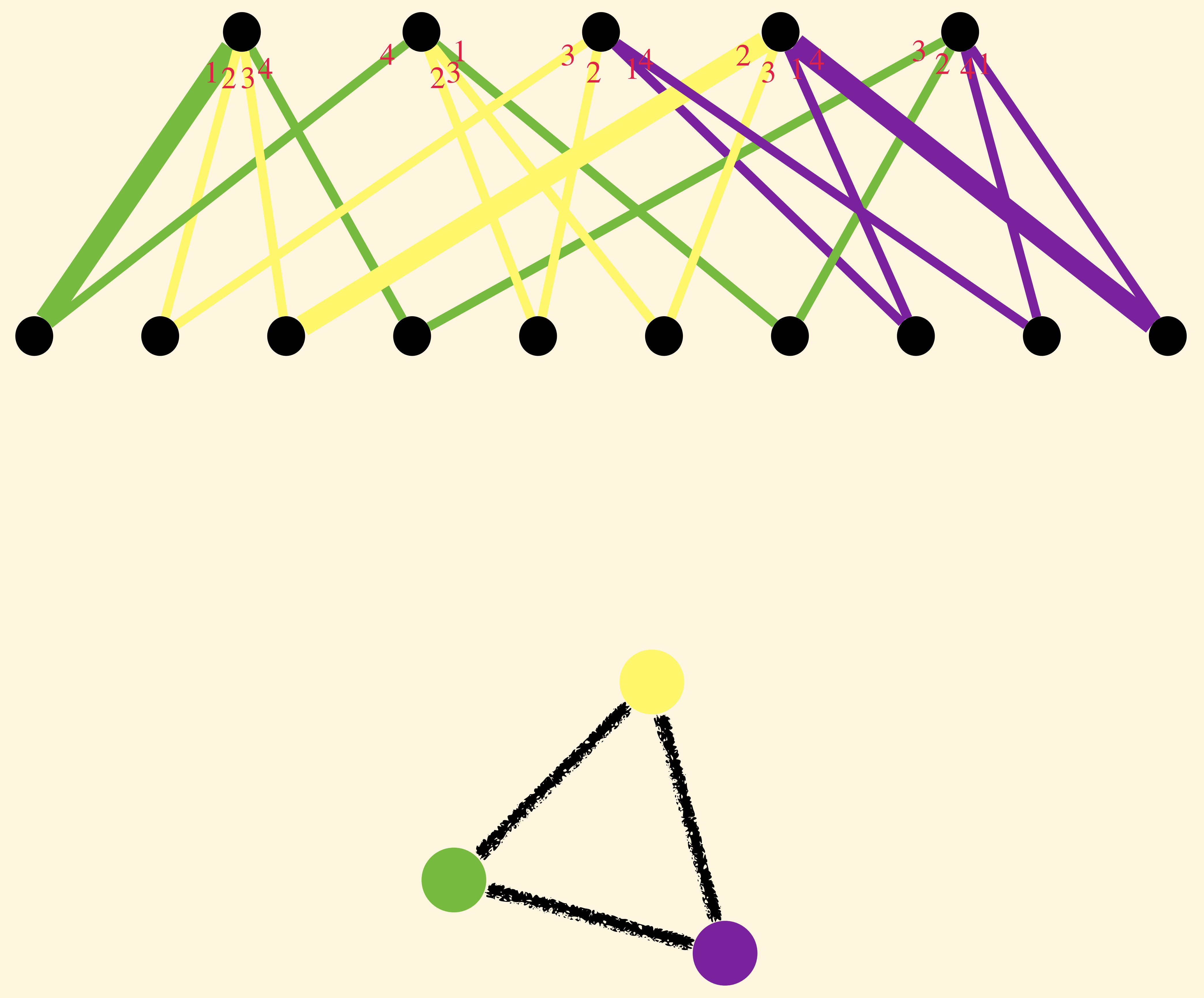
$$G_C = (\mathscr{C}, E_C)$$

$\mathscr{C} = \{C_1, C_2 \cdots\}$: set of circuits

$(C_i, C_j) \in E_C$ iff C_i and C_j have common vertex



Reduction to Ising Model

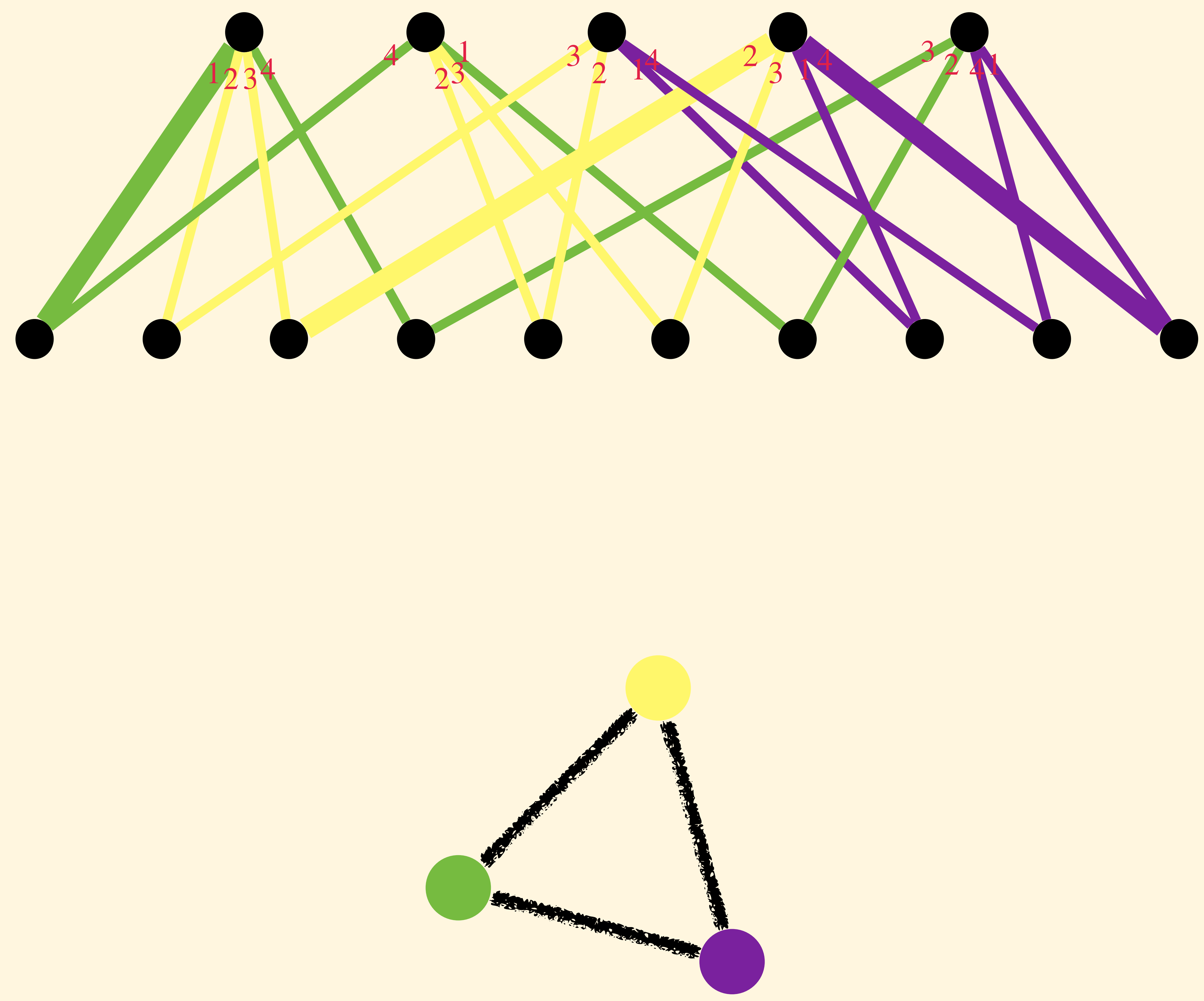


Reduction to Ising Model

Reduce configurations

The value of a circuit is the value of its initial edge

$$\sigma : E' \to \{0,1\} \implies \tau : \mathscr{C} \to \{0,1\}$$

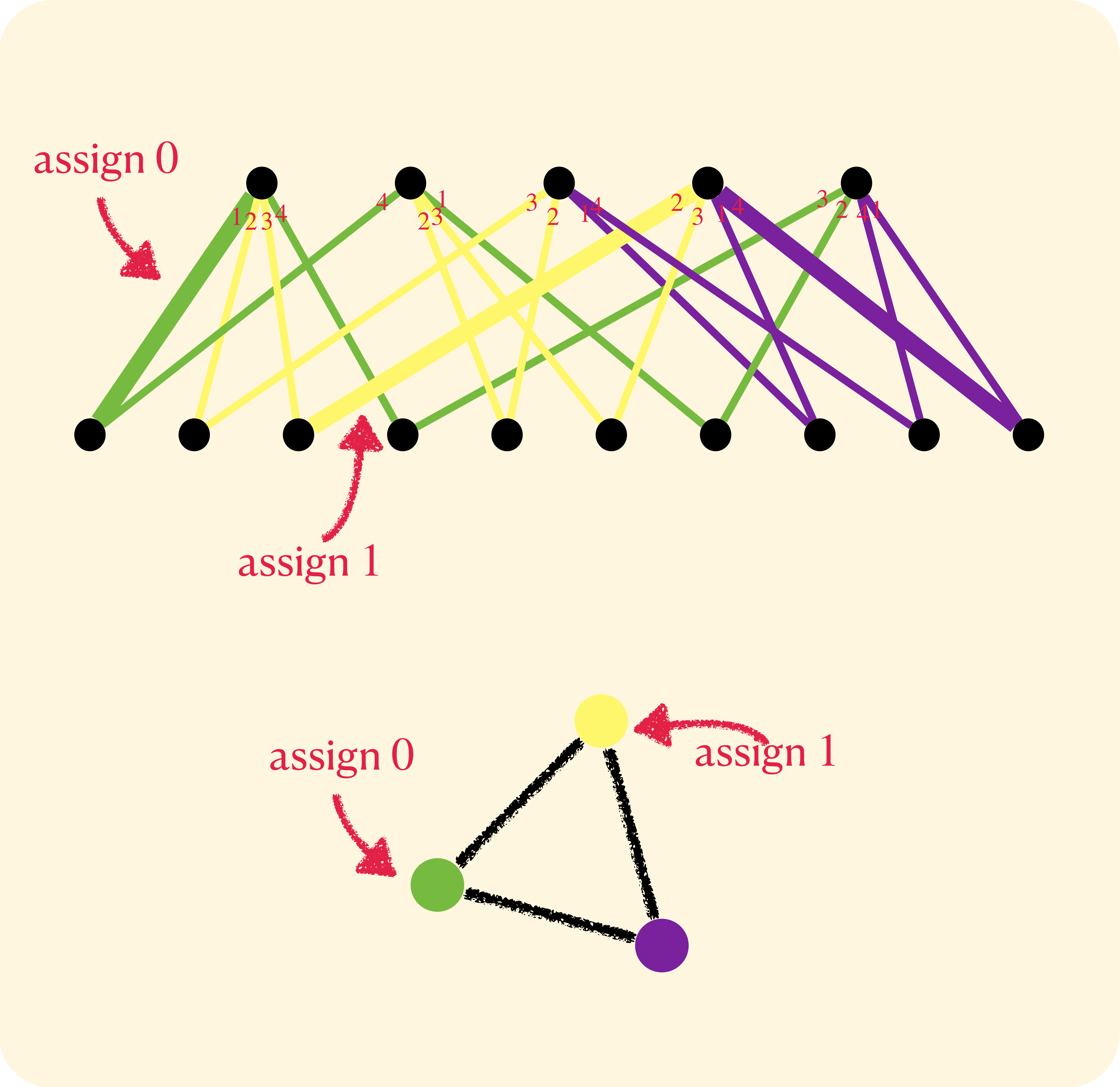


Reduction to Ising Model

Reduce configurations

The value of a circuit is the value of its initial edge

$$\sigma : E' \to \{0,1\} \Longrightarrow \tau : \mathscr{C} \to \{0,1\}$$



Reduction to Ising Model

Reduce configurations

The value of a circuit is the value of its initial edge

$$\sigma : E' \to \{0,1\} \Longrightarrow \tau : \mathscr{C} \to \{0,1\}$$

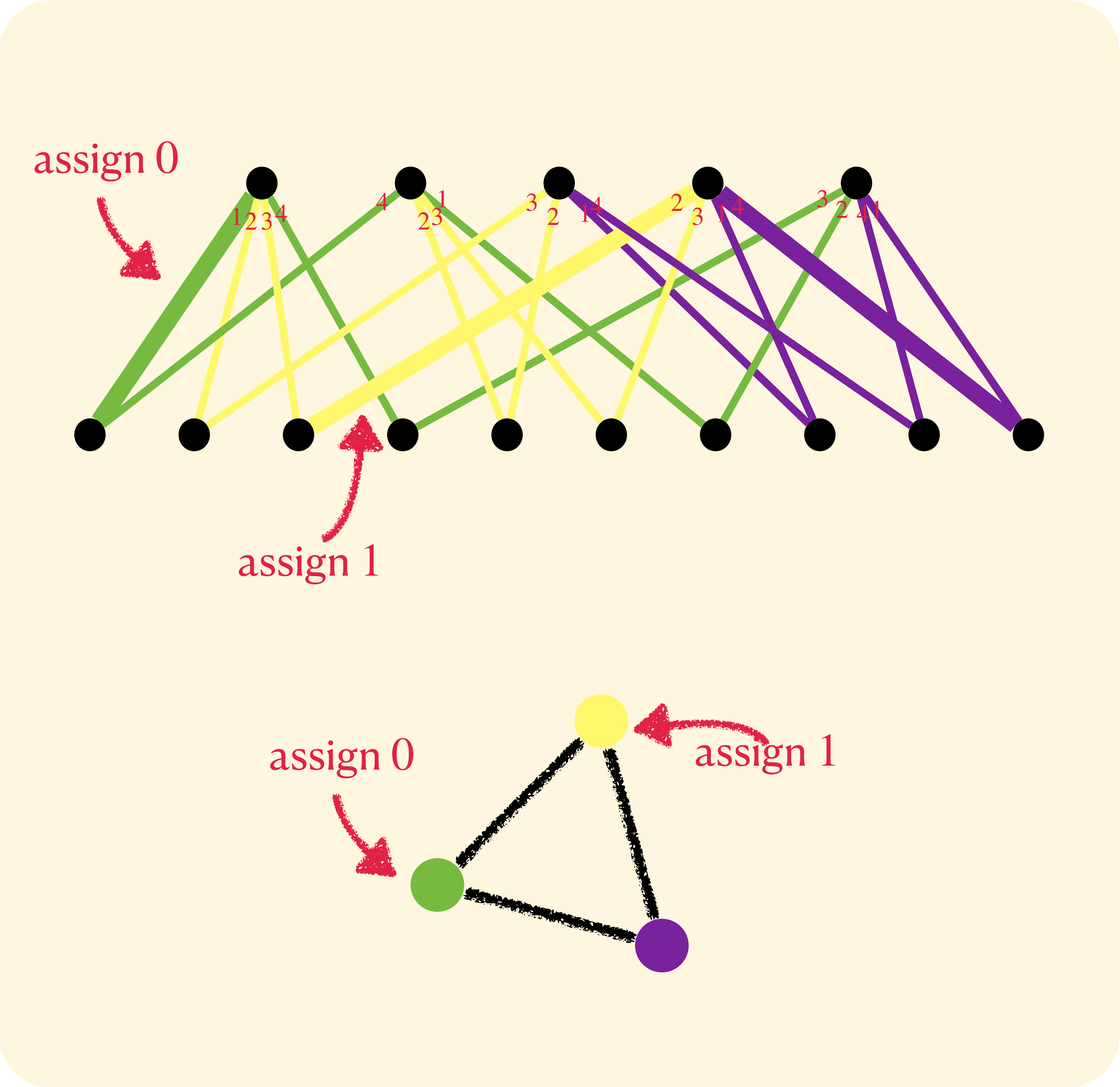
Reduce signatures

Classify common vertex between $u, v \in \mathscr{C}$

agree vertex: take value β if $\tau(u) = \tau(v)$

take value 1 if $\tau(u) \neq \tau(v)$

disagree vertex is the opposite



Reduction to Ising Model

Reduce configurations

The value of a circuit is the value of its initial edge

$$\sigma : E' \to \{0,1\} \implies \tau : \mathscr{C} \to \{0,1\}$$

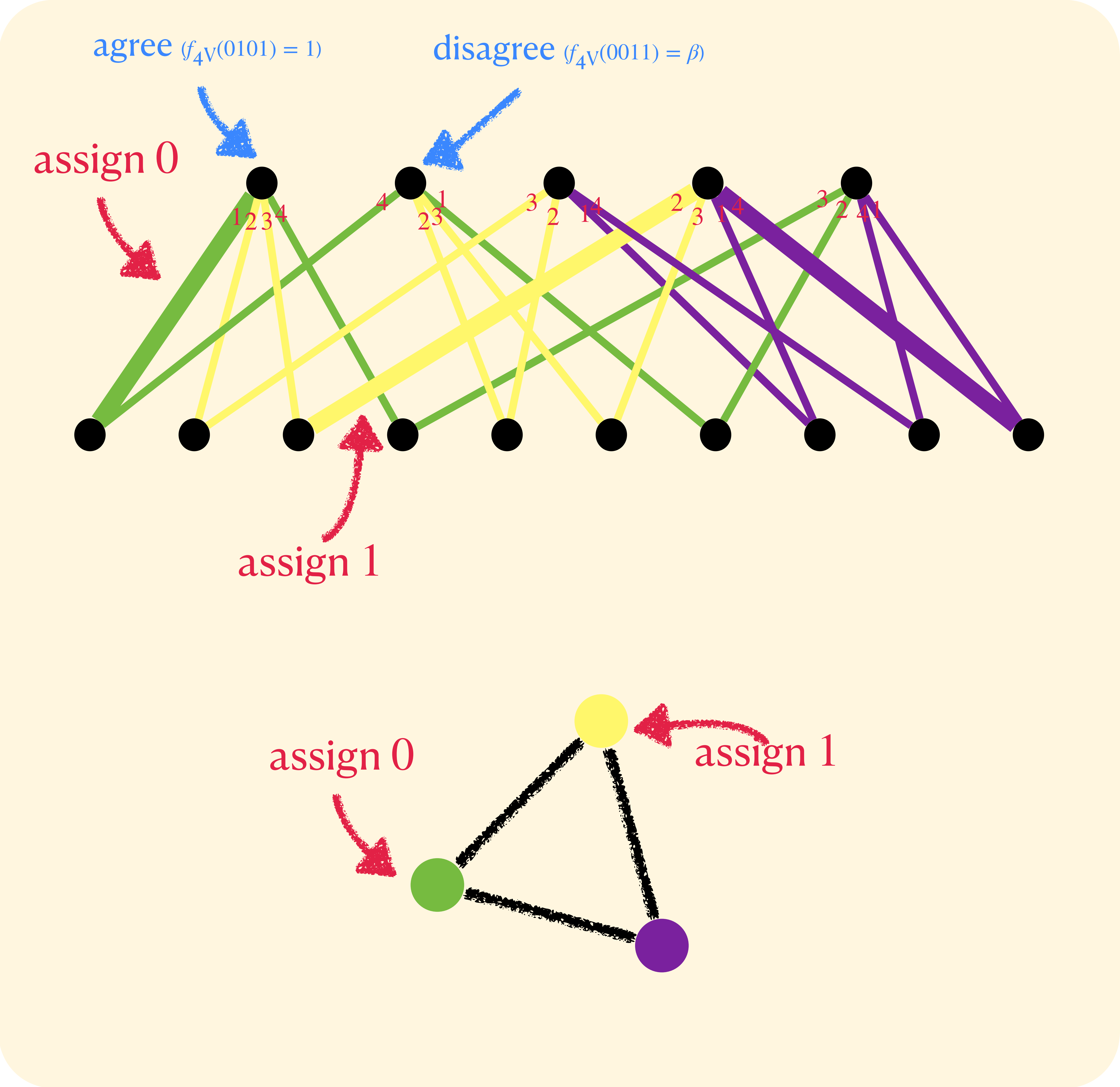
Reduce signatures

Classify common vertex between $u, v \in \mathscr{C}$

agree vertex: take value β if $\tau(u) = \tau(v)$

take value 1 if $\tau(u) \neq \tau(v)$

disagree vertex is the opposite



Reduction to Ising Model

Reduce configurations

The value of a circuit is the value of its initial edge

$$\sigma : E' \to \{0,1\} \Longrightarrow \tau : \mathscr{C} \to \{0,1\}$$

Reduce signatures

Classify common vertex between $u, v \in \mathscr{C}$

agree vertex: take value β if $\tau(u) = \tau(v)$

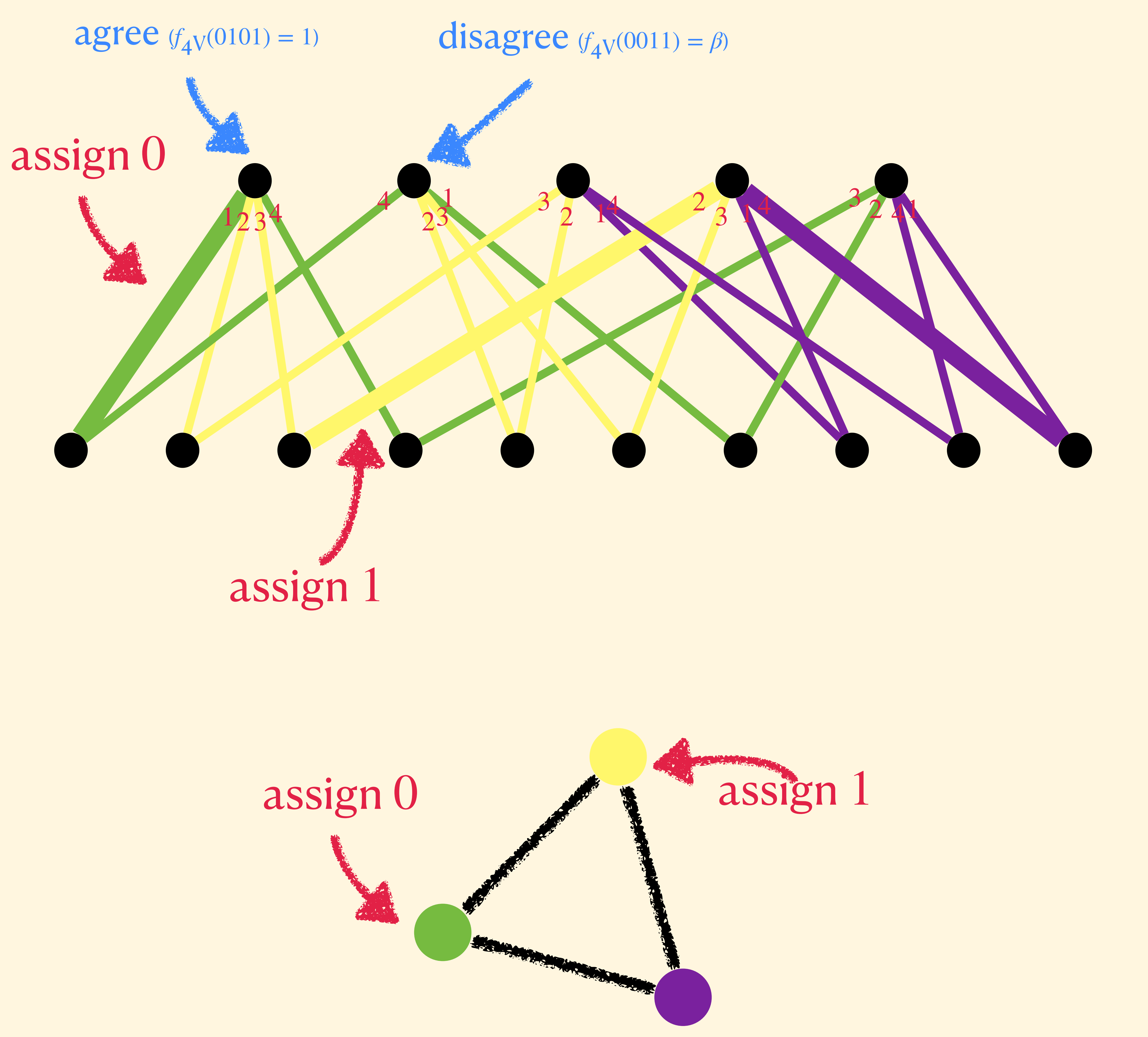
take value 1 if $\tau(u) \neq \tau(v)$

disagree vertex is the opposite

$$f_{4\mathrm{V}}\left(\begin{bmatrix} 0000 & 0010 & 0001 & 0011 \\ 0100 & 0110 & 0101 & 0111 \\ 1000 & 1010 & 1001 & 1011 \\ 1100 & 1110 & 1101 & 1111 \end{bmatrix}\right) = \begin{bmatrix} 0 & 0 & 0 & \beta \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \beta & 0 & 0 & 0 \end{bmatrix}$$

$$\Longrightarrow f_{uv}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} \beta^{A_{uv}} & \beta^{D_{uv}} \\ \beta^{D_{uv}} & \beta^{A_{uv}} \end{bmatrix} = \beta^{D_{uv}} \begin{bmatrix} \beta^{A_{uv}-D_{uv}} & 1 \\ 1 & \beta^{A_{uv}-D_{uv}} \end{bmatrix}$$

$$A_{uv} = \#\mathrm{AgreeVertex}, \quad D_{uv} = \#\mathrm{DisagreeVertex}$$



Reduction to Ising Model

Reduce configurations

The value of a circuit is the value of its initial edge

$$\sigma : E' \to \{0,1\} \Longrightarrow \tau : \mathscr{C} \to \{0,1\}$$

Reduce signatures

Classify common vertex between $u, v \in \mathscr{C}$

agree vertex: take value β if $\tau(u) = \tau(v)$

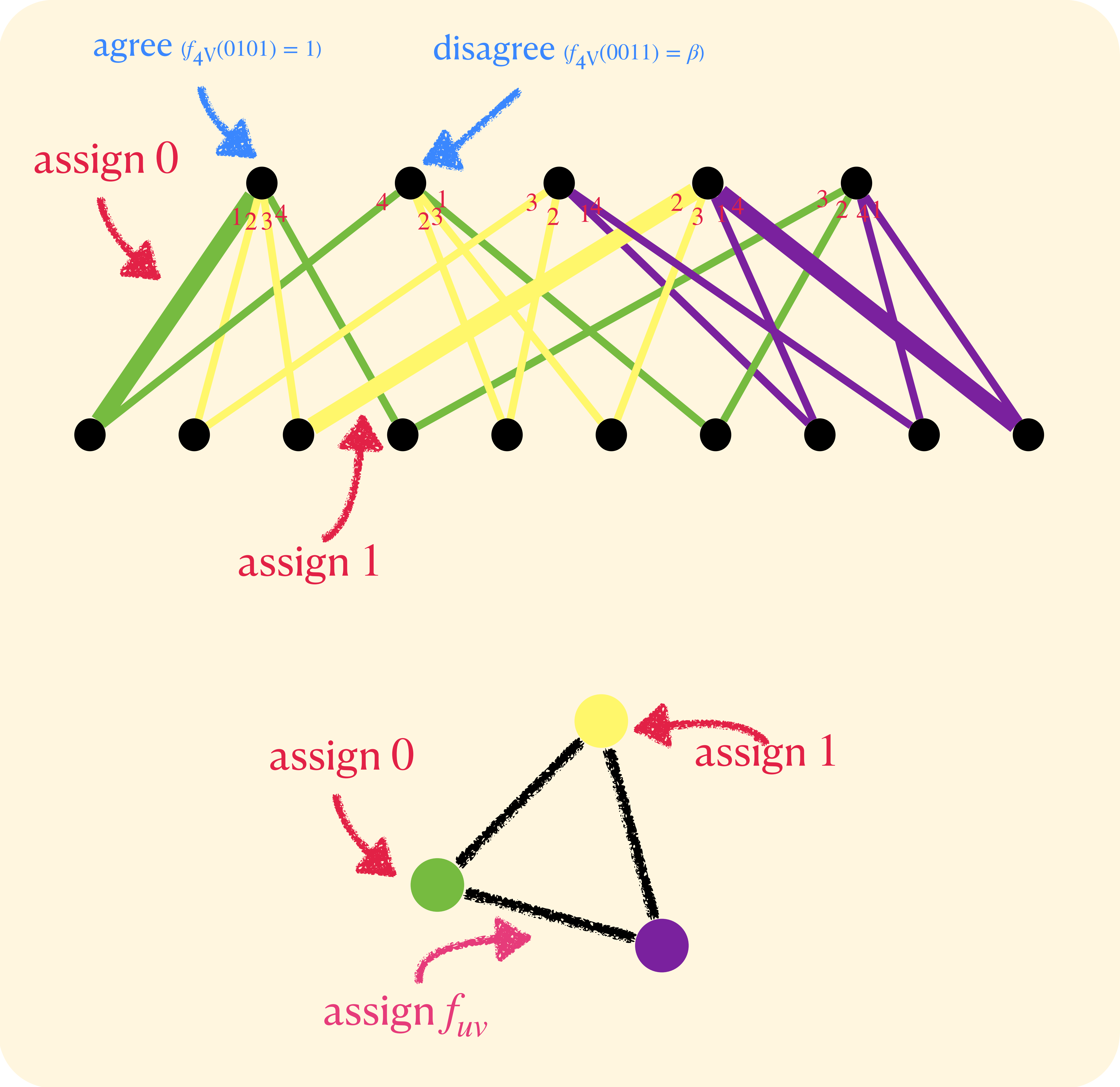
take value 1 if $\tau(u) \neq \tau(v)$

disagree vertex is the opposite

$$f_{4\mathrm{V}}\left(\begin{bmatrix} 0000 & 0010 & 0001 & 0011 \\ 0100 & 0110 & 0101 & 0111 \\ 1000 & 1010 & 1001 & 1011 \\ 1100 & 1110 & 1101 & 1111 \end{bmatrix}\right) = \begin{bmatrix} 0 & 0 & 0 & \beta \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ \beta & 0 & 0 & 0 \end{bmatrix}$$

$$\Longrightarrow f_{uv}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} \beta^{A_{uv}} & \beta^{D_{uv}} \\ \beta^{D_{uv}} & \beta^{A_{uv}} \end{bmatrix} = \beta^{D_{uv}} \begin{bmatrix} \beta^{A_{uv}-D_{uv}} & 1 \\ 1 & \beta^{A_{uv}-D_{uv}} \end{bmatrix}$$

$$A_{uv} = \#\text{AgreeVertex}, \quad D_{uv} = \#\text{DisagreeVertex}$$



Reduction to Ising Model

Reduction to Ising Model

Ising model [Lenz, 1920]

graph $G = (V, E)$ and local interaction $\beta = \{\beta_e\}_{e \in E}$

$$Z_{\text{Ising}}(G; \beta) = \sum_{\tau: V \to \{0,1\}} \prod_{e=(u,v) \in E} \beta_e^{\mathbf{I}(\tau(u)=\tau(v))}$$

Reduction to Ising Model

Ising model **[Lenz, 1920]**

graph $G = (V, E)$ and local interaction $\beta = \{\beta_e\}_{e \in E}$

$$Z_{\text{Ising}}(G; \beta) = \sum_{\tau: V \to \{0,1\}} \prod_{e=(u,v) \in E} \beta_e^{\mathbf{I}(\tau(u)=\tau(v))}$$

$$f_{uv}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} \beta^{A_{uv}-D_{uv}} & 1 \\ 1 & \beta^{A_{uv}-D_{uv}} \end{bmatrix}$$

is a Ising-type signature

Reduction to Ising Model

Ising model [Lenz, 1920]

graph $G = (V, E)$ and local interaction $\beta = \{\beta_e\}_{e \in E}$

$$Z_{\text{Ising}}(G;\beta) = \sum_{\tau: V \to \{0,1\}} \prod_{e=(u,v)\in E} \beta_e^{\mathbb{I}(\tau(u)=\tau(v))}$$

Approximability of Ising model [Jerrum and Sinclair, 1993]

If $\beta_e > 1$ for all $e \in E$, the system is *ferromagnetic*

$\Longrightarrow$ There ***are FPRASes*** for computing $Z_{\text{Ising}}(G;\beta)$

we use

- Jerrum-Sinclair chain
- **Worm process** [Prokof'ev and Svistunov, 2001]
- Swendsen-Wang dynamics [Guo and Jerrum, 2018]

$$f_{uv}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} \beta^{A_{uv}-D_{uv}} & 1 \\ 1 & \beta^{A_{uv}-D_{uv}} \end{bmatrix}$$

is a Ising-type signature

Reduction to Ising Model

Ising model [Lenz, 1920]

graph $G = (V, E)$ and local interaction $\beta = \{\beta_e\}_{e \in E}$

$$Z_{\text{Ising}}(G; \beta) = \sum_{\tau: V \to \{0,1\}} \prod_{e=(u,v) \in E} \beta_e^{\mathbf{I}(\tau(u)=\tau(v))}$$

Approximability of Ising model [Jerrum and Sinclair, 1993]

If $\beta_e > 1$ for all $e \in E$, the system is *ferromagnetic*

$\Longrightarrow$ There ***are FPRASes*** for computing $Z_{\text{Ising}}(G; \beta)$

- Jerrum-Sinclair chain
- **Worm process** [Prokof'ev and Svistunov, 2001]
- Swendsen-Wang dynamics [Guo and Jerrum, 2018]

we use

If $\beta_e < 1$ for some $e \in E$, the system is *anti-ferromagnetic*

$\Longrightarrow$ There is ***no FPRAS*** for computing $Z_{\text{Ising}}(G; \beta)$

$$f_{uv}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} \beta^{A_{uv}-D_{uv}} & 1 \\ 1 & \beta^{A_{uv}-D_{uv}} \end{bmatrix}$$

is a Ising-type signature

Reduction to Ising Model

Ising model **[Lenz, 1920]**

graph $G = (V, E)$ and local interaction $\beta = \{\beta_e\}_{e \in E}$

$$Z_{\text{Ising}}(G;\beta) = \sum_{\tau : V \to \{0,1\}} \prod_{e=(u,v)\in E} \beta_e^{\mathbf{I}(\tau(u)=\tau(v))}$$

Approximability of Ising model **[Jerrum and Sinclair, 1993]**

If $\beta_e > 1$ for all $e \in E$, the system is *ferromagnetic*

$\Longrightarrow$ There ***are FPRASes*** for computing $Z_{\text{Ising}}(G;\beta)$

we use

- Jerrum-Sinclair chain
- **Worm process** [Prokof'ev and Svistunov, 2001]
- Swendsen-Wang dynamics [Guo and Jerrum, 2018]

If $\beta_e < 1$ for some $e \in E$, the system is *anti-ferromagnetic*

$\Longrightarrow$ There is ***no FPRAS*** for computing $Z_{\text{Ising}}(G;\beta)$

$$f_{uv}\left(\begin{bmatrix} 00 & 01 \\ 10 & 11 \end{bmatrix}\right) = \begin{bmatrix} \beta^{A_{uv}-D_{uv}} & 1 \\ 1 & \beta^{A_{uv}-D_{uv}} \end{bmatrix}$$

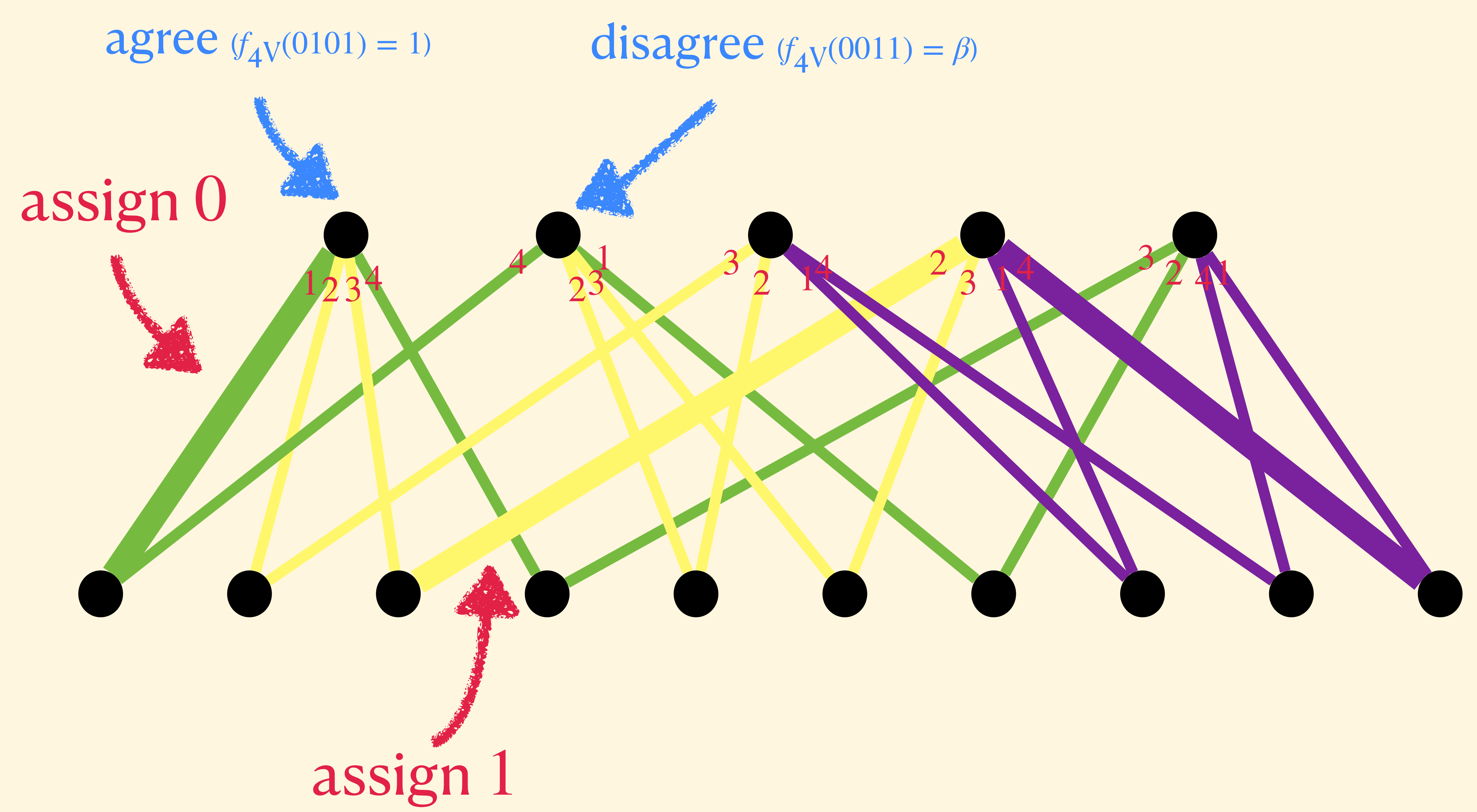
is a Ising-type signature

Ferromagnetic
Or
Anti-ferromagnetic?

Choose initial edges “wisely”

Observation

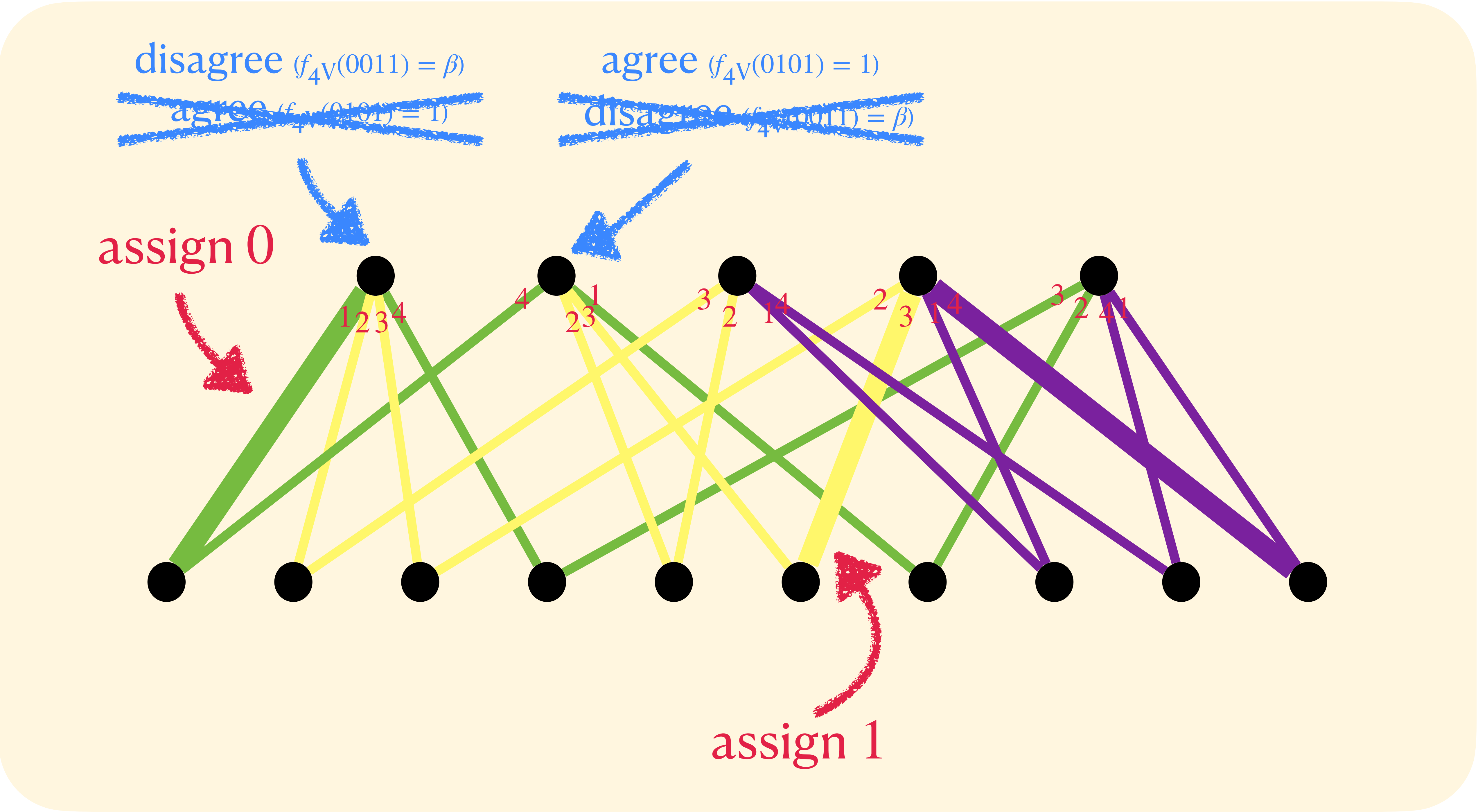
Changing the initial edge of a circuit will swap the agree and disagree vertices between it and its neighbors



Choose initial edges “wisely”

Observation

Changing the initial edge of a circuit will swap the agree and disagree vertices between it and its neighbors



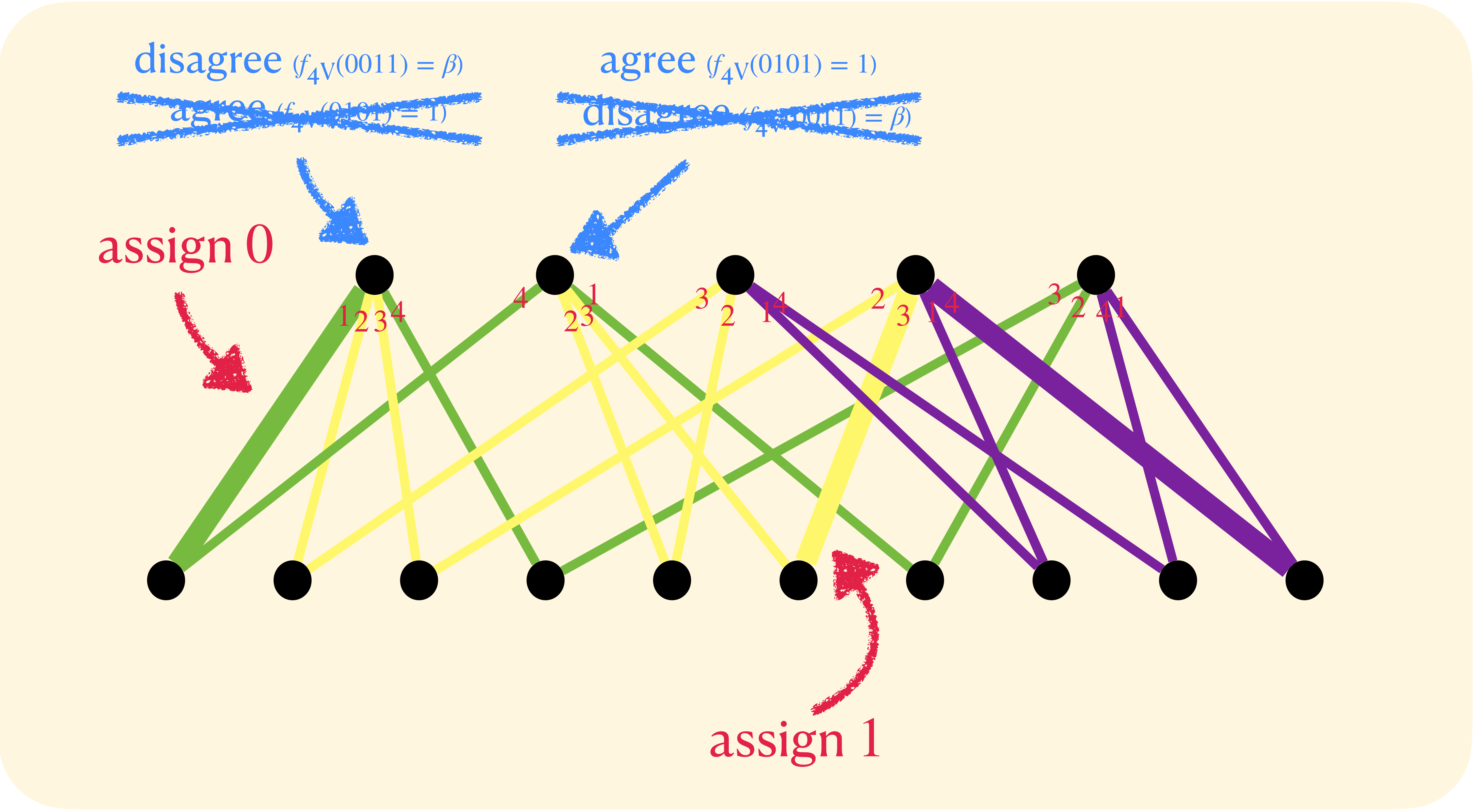
Choose initial edges "wisely"

Observation

Changing the initial edge of a circuit will swap the agree and disagree vertices between it and its neighbors

Idea

Change some initial edges to make $A_{uv} > D_{uv}$ for all $(u, v) \in E_C$ — obtain a ferromagnetic system



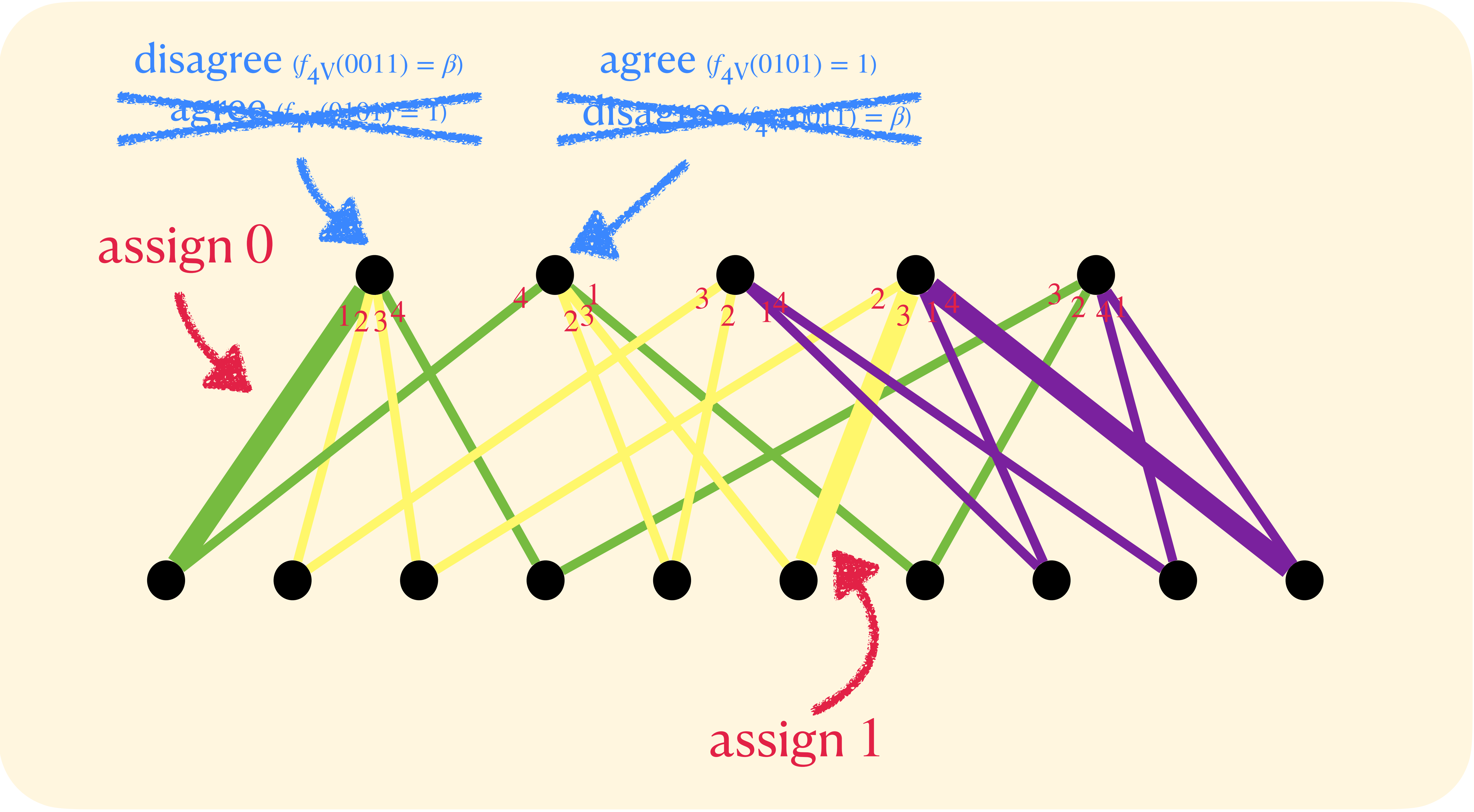
Choose initial edges “wisely”

Observation

Changing the initial edge of a circuit will swap the agree and disagree vertices between it and its neighbors

Idea

Change some initial edges to make $A_{uv} > D_{uv}$ for all $(u, v) \in E_C$ — obtain a ferromagnetic system



- Choose initial edges randomly
- Let Boolean variable X_u to indicate whether change u’s initial edge ($X_u = 1$ means “change”)
- Solve XOR equation system $X_u \oplus X_v = \begin{cases} 1 \text{ if } A_{uv} < D_{uv} \\ 0 \text{ if } A_{uv} > D_{uv} \end{cases}$

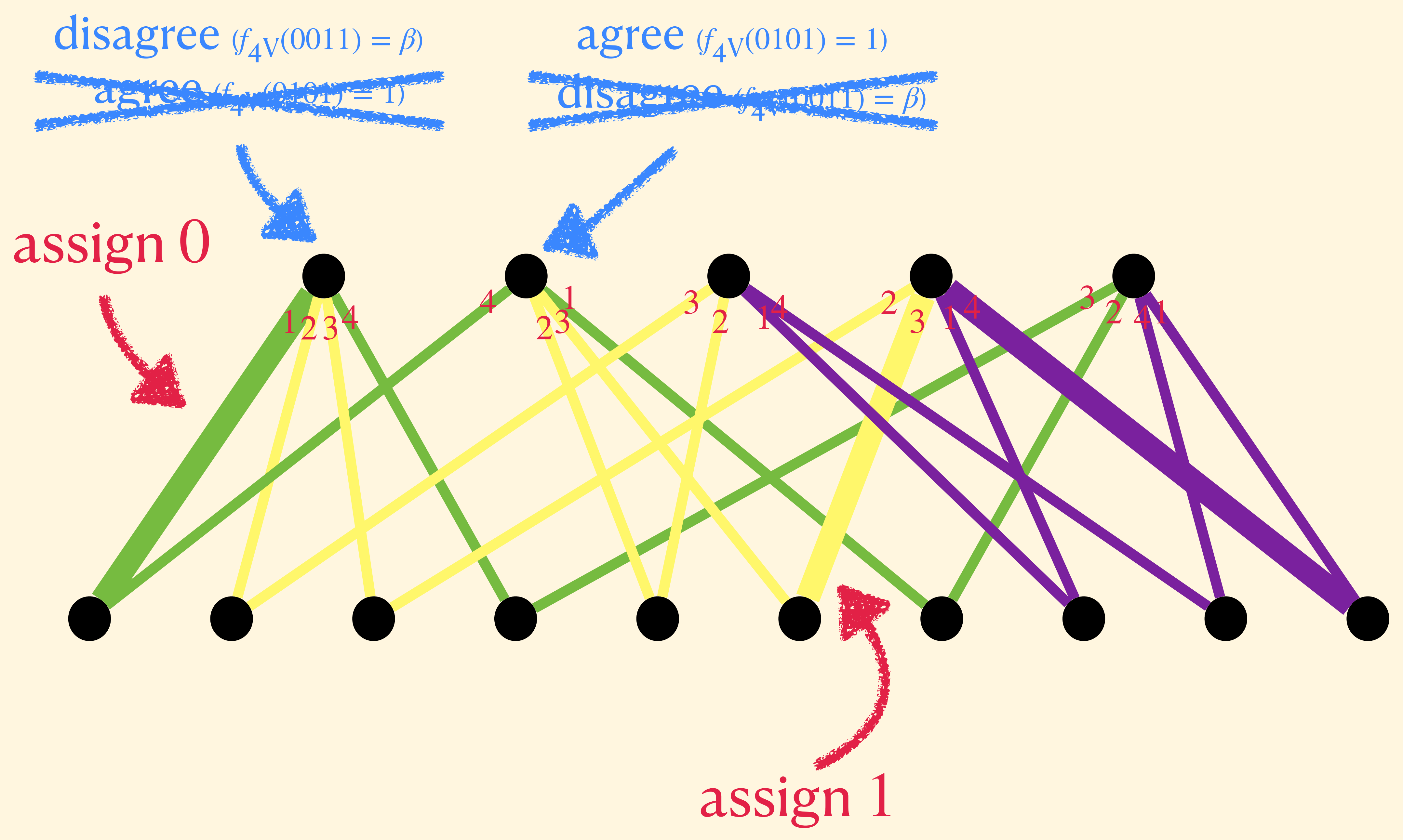
Choose initial edges "wisely"

Observation

Changing the initial edge of a circuit will swap the agree and disagree vertices between it and its neighbors

Idea

Change some initial edges to make $A_{uv} > D_{uv}$ for all $(u, v) \in E_C$ — obtain a ferromagnetic system



- Choose initial edges randomly
- Let Boolean variable X_u to indicate whether change u's initial edge ($X_u = 1$ means "change")
- Solve XOR equation system $X_u \oplus X_v = \begin{cases} 1 \text{ if } A_{uv} < D_{uv} \\ 0 \text{ if } A_{uv} > D_{uv} \end{cases}$

Algebra criterion to compute Z_{4V}

If this system has a solution, there is ***an FPRAS***.

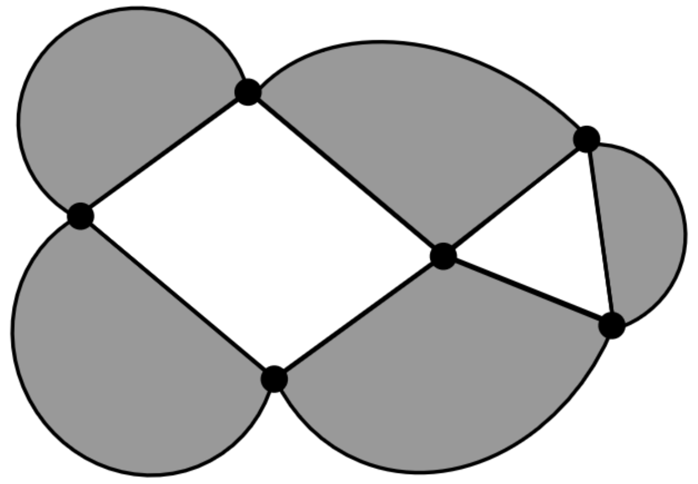
If this system has no solution, there is ***no FPRAS***.

Planar Graphs and Beyond

Planar Graphs and Beyond

Face 2-coloring

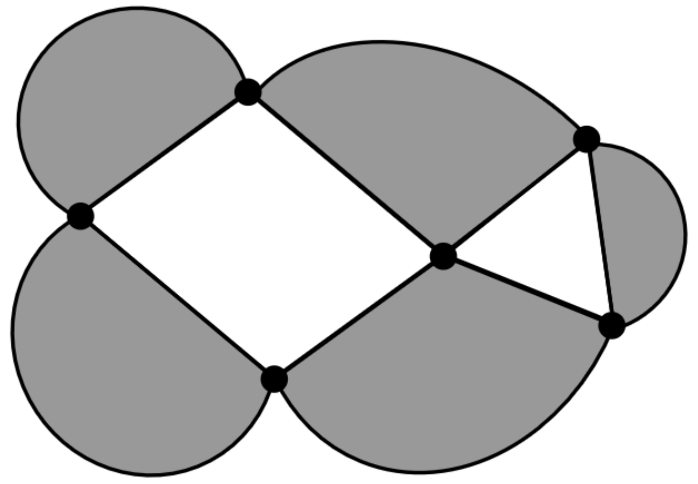
The dual of a planar 4-regular graph is bipartite [Welsh, 1969]



Planar Graphs and Beyond

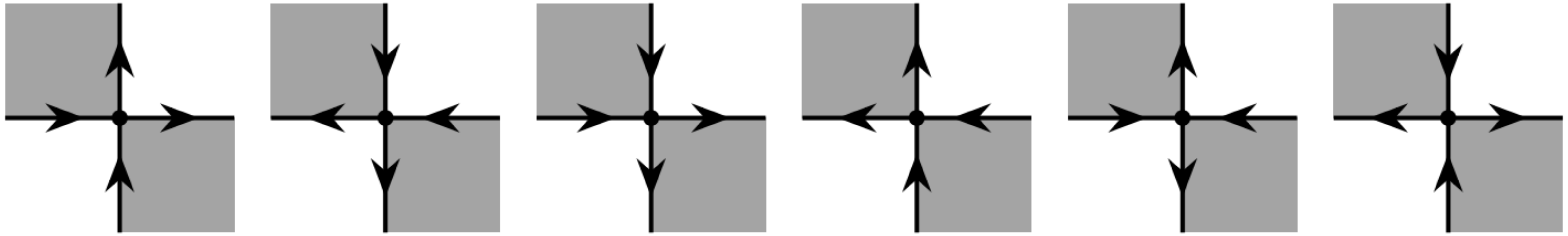
Face 2-coloring

The dual of a planar 4-regular graph is bipartite [Welsh, 1969]



Canonical labeling

Based on face 2-coloring, we design a canonical labeling

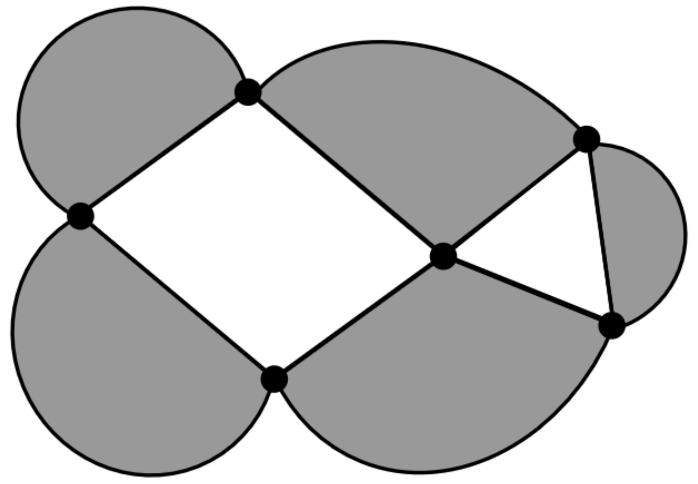


1 2 3 4 5 6

Planar Graphs and Beyond

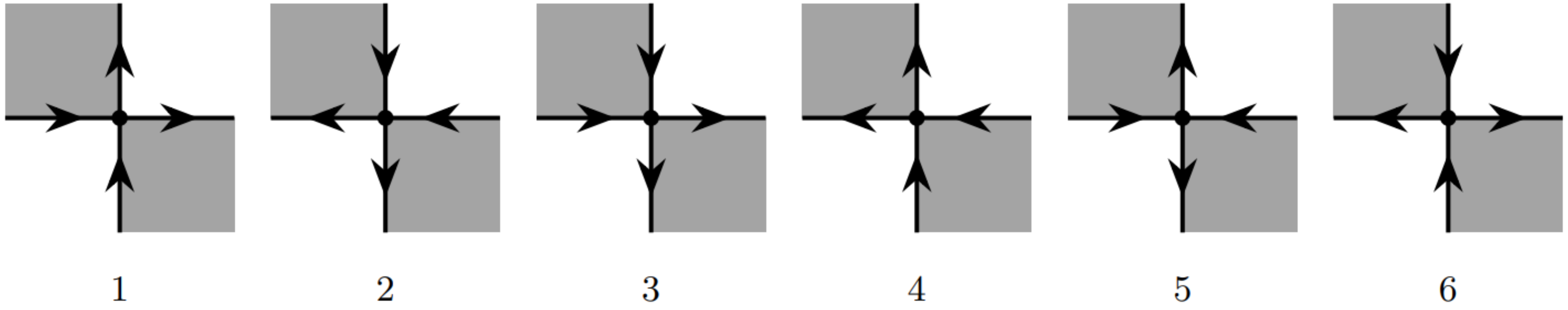
Face 2-coloring

The dual of a planar 4-regular graph is bipartite [Welsh, 1969]



Canonical labeling

Based on face 2-coloring, we design a canonical labeling



- The canonical labeling already encodes the *staggered ice-type model*
- Under canonical labeling, the XOR equation system is always ***solvable***
- This method also works for *torus grid graphs*.

Thank you!

Theor. Comput. Sci. 114491

Open Problems

- Fully approximability of six-vertex model
- FPTAS for six-vertex model on graphs in the low-temperature regime
- Better or optimal mixing time for six-vertex model
- Spectral independence for six-vertex model