Hardness of 4-colouring G-colourable graphs

Gianluca Tasinato*

Joint work with:
S. Avvakumov†, M. Filakovský‡, J. Opršal§, U. Wagner*

†Tel Aviv University; ‡Masaryk University; §University of Birmingham; *ISTA

Approximate graph colourings

Deciding if a given graph is 3-colourable is a classical NP-hard problem. [Karp; '72]

Graph = undirected simple finite graph on n vertices.

Approximate graph colourings

Deciding if a given graph is 3-colourable is a classical NP-hard problem. [Karp; '72]

Approximate Graph Colouring problem [Search]: Fix $k \geq 4$. Given a 3-colourable graph G, find a k-colouring.

Graph = undirected simple finite graph on n vertices.

Approximate graph colourings

Deciding if a given graph is 3-colourable is a classical NP-hard problem. [Karp; '72]

Approximate Graph Colouring problem [Decision]: Fix $k \geq 4$. Given a graph G, decide if G is 3-colourable or not even k-colourable.

Graph = undirected simple finite graph on n vertices.

Approximate graph colourings

Deciding if a given graph is 3-colourable is a classical NP-hard problem. [Karp; '72]

Approximate Graph Colouring problem [Decision]: Fix $k \geq 4$. Given a graph G, decide if G is 3-colourable or not even k-colourable.

k	Complexity
4	NP-hard

[Khanna, Linial, Safra; '00]

Graph = undirected simple finite graph on n vertices.

Approximate graph colourings

Deciding if a given graph is 3-colourable is a classical NP-hard problem. [Karp; '72]

Approximate Graph Colouring problem [Decision]: Fix $k \geq 4$. Given a graph G, decide if G is 3-colourable or not even k-colourable.

k	Complexity	
4	NP-hard	[Khanna, Linial, Safra; '00]
5	NP-hard	[Bulín, Krokhin, Opršal; '19]

Graph = undirected simple finite graph on n vertices.

Approximate graph colourings

Deciding if a given graph is 3-colourable is a classical NP-hard problem. [Karp; '72]

Approximate Graph Colouring problem [Decision]: Fix $k \geq 4$. Given a graph G, decide if G is 3-colourable or not even k-colourable.

k	Complexity	
4	NP-hard	[Khanna, Linial, Safra; '00]
5	NP-hard	[Bulín, Krokhin, Opršal; '19]
6	???	NP-hard modulo UGC
		[Dinur, Mossel, Regev; '09]
⋮	⋮	[Guruswami, Sandeep; '20]
		[Breverman, Khot, Lifshitz, Mulzer; '21]

Graph = undirected simple finite graph on n vertices.

Approximate graph colourings

Deciding if a given graph is 3-colourable is a classical NP-hard problem. [Karp; '72]

Approximate Graph Colouring problem [Decision]: Fix $k \geq 4$. Given a graph G, decide if G is 3-colourable or not even k-colourable.

k	Complexity	
4	NP-hard	[Khanna, Linial, Safra; '00]
5	NP-hard	[Bulín, Krokhin, Opršal; '19]
6	???	NP-hard modulo UGC
		[Dinur, Mossel, Regev; '09]
⋮	⋮	[Guruswami, Sandeep; '20]
		[Breverman, Khot, Lifshitz, Mulzer; '21]
$n^{0.19996}$	P	[Kawarabayashi, Thorup; '17]

Graph = undirected simple finite graph on n vertices.

Graph homomorphisms

Definition (Graph Homomorphism)

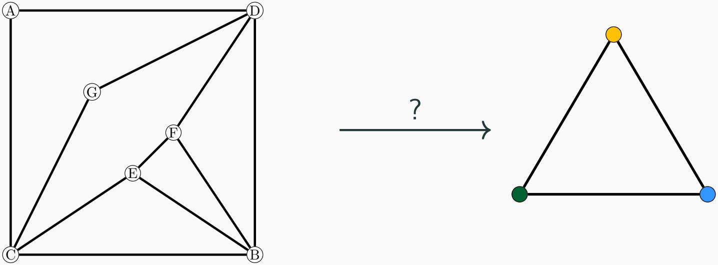
Given $G = (V_G; E_G)$ and $H = (V_H; E_H)$ graphs, a **graph homomorphism** is a map $f : V_G \to V_H$ that respects edges, i.e. for all $(u, v) \in E_G$, $(f(u), f(v)) \in E_H$.

Graph homomorphisms

Definition (Graph Homomorphism)

Given $G = (V_G; E_G)$ and $H = (V_H; E_H)$ graphs, a **graph homomorphism** is a map $f : V_G \to V_H$ that respects edges, i.e. for all $(u, v) \in E_G$, $(f(u), f(v)) \in E_H$.

Example: k-colouring of G is graph homomorphism $G \to K_k$.



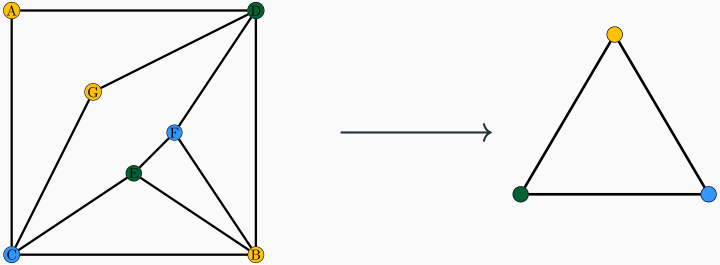
K_k = complete graph on k vertices.

Graph homomorphisms

Definition (Graph Homomorphism)

Given $G = (V_G; E_G)$ and $H = (V_H; E_H)$ graphs, a **graph homomorphism** is a map $f : V_G \to V_H$ that respects edges, i.e. for all $(u, v) \in E_G$, $(f(u), f(v)) \in E_H$.

Example: k-colouring of G is graph homomorphism $G \to K_k$.



K_k = complete graph on k vertices.

Promise Graph Homomorphism

Let G, H graphs such that $G \to H$. The (decision) Promise Graph Homomorphism Problem $\mathrm{PCSP}(G, H)$ is the following problem:

Input: a graph I *Output:* **<u>Yes</u>** if $I \to G$; **<u>No</u>** if $I \not\to H$

Promise Graph Homomorphism

Let G, H graphs such that $G \to H$. The (decision) Promise Graph Homomorphism Problem $\mathrm{PCSP}(G, H)$ is the following problem:

Input: a graph I *Output:* **Yes** if $I \to G$; **No** if $I \nrightarrow H$

Conj: If G and H are non bipartite graphs with $G \to H$, then $\mathrm{PCSP}(G, H)$ is NP-hard. [Brakensiek, Guruswami; '18]

Promise Graph Homomorphism

Let G, H graphs such that $G \to H$. The (decision) Promise Graph Homomorphism Problem $\mathrm{PCSP}(G, H)$ is the following problem:

Input: a graph I *Output:* **<u>Yes</u>** if $I \to G$; **<u>No</u>** if $I \nrightarrow H$

Conj: If G and H are non bipartite graphs with $G \to H$, then $\mathrm{PCSP}(G, H)$ is NP-hard. [Brakensiek, Guruswami; '18]

Rmk: It is enough to show it for $G = C_\ell$ (ℓ odd) and $H = K_k$.

Promise Graph Homomorphism

Let G, H graphs such that $G \to H$. The (decision) Promise Graph Homomorphism Problem $\mathrm{PCSP}(G, H)$ is the following problem:

Input: a graph I *Output:* **Yes** if $I \to G$; **No** if $I \nrightarrow H$

Conj: If G and H are non bipartite graphs with $G \to H$, then $\mathrm{PCSP}(G, H)$ is NP-hard. [Brakensiek, Guruswami; '18]

Rmk: It is enough to show it for $G = C_\ell$ (ℓ odd) and $H = K_k$.

Theorem (Krokhin, Opršal, Wrochna, Živný; '23)

For any 3-colourable non bipartite graph G, $\mathrm{PCSP}(G, K_3)$ is NP-hard.

Promise Graph Homomorphism

Let G, H graphs such that $G \to H$. The (decision) Promise Graph Homomorphism Problem $\mathrm{PCSP}(G, H)$ is the following problem:

Input: a graph I *Output:* **Yes** if $I \to G$; **No** if $I \nrightarrow H$

Conj: If G and H are non bipartite graphs with $G \to H$, then $\mathrm{PCSP}(G, H)$ is NP-hard. [Brakensiek, Guruswami; '18]

Rmk: It is enough to show it for $G = C_\ell$ (ℓ odd) and $H = K_k$.

Theorem (Krokhin, Opršal, Wrochna, Živný; '23)

For any 3-colourable non bipartite graph G, $\mathrm{PCSP}(G, K_3)$ *is NP-hard.*

Theorem (Avvakumov, Filakovský, Opršal, T., Wagner; '25+)

For any 4-colourable non bipartite graph G, $\mathrm{PCSP}(G, K_4)$ *is NP-hard.*

Proof Structure

- By a general algebraic theory of PCSPs[1], the complexity of PCSP(C_ℓ, K_4) is governed by its *polymorphisms*

$$f : C_\ell^n \to K_4.$$

[1] [Barto, Bulín, Krokhin, Opršal; '21]

Proof Structure

- By a general algebraic theory of PCSPs[1], the complexity of PCSP(C_ℓ, K_4) is governed by its *polymorphisms*

$$f : C_\ell^n \to K_4.$$

C_ℓ^n is categorical/tensor product: vertices of C_ℓ^n are n-tuples of vertices, edge between two tuples when each coordinate form an edge.

[1] [Barto, Bulín, Krokhin, Opršal; '21]

Proof Structure

- By a general algebraic theory of PCSPs[1], the complexity of $\mathrm{PCSP}(C_\ell, K_4)$ is governed by its *polymorphisms*

$$f : C_\ell^n \to K_4.$$

- Via topology to $f\colon C_\ell^n \to K_4$ we associate a map $\phi(f) : \mathbb{Z}_2^n \to \mathbb{Z}_2$ of the form:

$$\phi(f)(x_1, \ldots, x_n) = \sum_{i=1}^{n} \alpha_i x_i$$

with $\sum_i \alpha_i = 1 \mod 2$, respecting variable substitutions & permutations.

[1] [Barto, Bulín, Krokhin, Opršal; '21]

Proof Structure

- By a general algebraic theory of PCSPs[1], the complexity of $\mathrm{PCSP}(C_\ell, K_4)$ is governed by its *polymorphisms*

$$f : C_\ell^n \to K_4.$$

- Via topology to $f\colon C_\ell^n \to K_4$ we associate a map $\phi(f) : \mathbb{Z}_2^n \to \mathbb{Z}_2$ of the form:

$$\phi(f)(x_1, \ldots, x_n) = \sum_{i=1}^{n} \alpha_i x_i$$

with $\sum_i \alpha_i = 1 \mod 2$, respecting variable substitutions & permutations.

E.g. $f\colon C_\ell^2 \to K_4$ and $g\colon C_\ell^4 \to K_4$ such that:

$$f(x, y) = g(x, y, y, x)$$

[1] [Barto, Bulín, Krokhin, Opršal; '21]

Proof Structure

- By a general algebraic theory of PCSPs[1], the complexity of PCSP(C_ℓ, K_4) is governed by its *polymorphisms*

$$f : C_\ell^n \to K_4.$$

- Via topology to $f\colon C_\ell^n \to K_4$ we associate a map $\phi(f) : \mathbb{Z}_2^n \to \mathbb{Z}_2$ of the form:

$$\phi(f)(x_1, \dots, x_n) = \sum_{i=1}^{n} \alpha_i x_i$$

with $\sum_i \alpha_i = 1 \mod 2$, respecting variable substitutions & permutations.
E.g. $f\colon C_\ell^2 \to K_4$ and $g\colon C_\ell^4 \to K_4$ such that:

$$f(x, y) = g(x, y, y, x) \leftrightharpoons \phi(f)(x, y) = \phi(g)(x, y, y, x)$$

1 [Barto, Bulín, Krokhin, Opršal; '21]

Proof Structure

- By a general algebraic theory of PCSPs[1], the complexity of $\mathrm{PCSP}(C_\ell, K_4)$ is governed by its *polymorphisms*

$$f : C_\ell^n \to K_4.$$

- Via topology to $f\colon C_\ell^n \to K_4$ we associate a map $\phi(f) : \mathbb{Z}_2^n \to \mathbb{Z}_2$ of the form:

$$\phi(f)(x_1, \dots, x_n) = \sum_{i=1}^{n} \alpha_i x_i$$

with $\sum_i \alpha_i = 1 \mod 2$, respecting variable substitutions & permutations.
- Via a combinatorial argument, the number of non-zero α_i is $O(\ell^2)$, *independent of n* $\Rightarrow$ hardness follows from general algebraic theory.

[1] [Barto, Bulín, Krokhin, Opršal; '21]

Proof Structure

- By a general algebraic theory of PCSPs[1], the complexity of $\mathrm{PCSP}(C_\ell, K_4)$ is governed by its *polymorphisms*

$$f : C_\ell^n \to K_4.$$

- Via topology to $f : C_\ell^n \to K_4$ we associate a map $\phi(f) : \mathbb{Z}_2^n \to \mathbb{Z}_2$ of the form:

$$\phi(f)(x_1, \ldots, x_n) = \sum_{i=1}^{n} \alpha_i x_i$$

with $\sum_i \alpha_i = 1 \mod 2$, respecting variable substitutions & permutations.

- Via a combinatorial argument, the number of non-zero α_i is $O(\ell^2)$, *independent of n* $\Rightarrow$ hardness follows from general algebraic theory.

[1] [Barto, Bulín, Krokhin, Opršal; '21]

Hom Complex

Hom-complex is a well-studied construction in topological combinatorics. [Lovász; '78]

Hom Complex

Hom-complex is a well-studied construction in topological combinatorics. [Lovász; '78]

Hom$(K_2, -)$ is a "recipe" to combinatorially build a topological space (simplicial set).

Hom Complex

Hom-complex is a well-studied construction in topological combinatorics. [Lovász; '78]

Hom$(K_2, -)$ is a "recipe" to combinatorially build a topological space (simplicial set). The key properties we will use are:

Hom Complex

Hom-complex is a well-studied construction in topological combinatorics. [Lovász; '78]

$\text{Hom}(K_2, -)$ is a "recipe" to combinatorially build a topological space (simplicial set). The key properties we will use are:

1. For any G, $\text{Hom}(K_2, G)$ inherits the $\mathbb{Z}_2$-symmetry of K_2 ($\mathbb{Z}_2$-action).

Hom Complex

Hom-complex is a well-studied construction in topological combinatorics. [Lovász; '78]

$\mathrm{Hom}(K_2, -)$ is a "recipe" to combinatorially build a topological space (simplicial set). The key properties we will use are:

1. For any G, $\mathrm{Hom}(K_2, G)$ inherits the $\mathbb{Z}_2$-symmetry of K_2 ($\mathbb{Z}_2$-action).
2. For any $G \to H$, there is a corresponding continuous map $\mathrm{Hom}(K_2, G) \to \mathrm{Hom}(K_2, H)$ respecting the symmetry.

Hom Complex

Hom-complex is a well-studied construction in topological combinatorics. [Lovász; '78]

$\mathrm{Hom}(K_2, -)$ is a "recipe" to combinatorially build a topological space (simplicial set). The key properties we will use are:

1. For any G, $\mathrm{Hom}(K_2, G)$ inherits the $\mathbb{Z}_2$-symmetry of K_2 ($\mathbb{Z}_2$-action).
2. For any $G \to H$, there is a corresponding continuous map $\mathrm{Hom}(K_2, G) \to \mathrm{Hom}(K_2, H)$ respecting the symmetry.
3. Up to a (symmetry-preserving) topological equivalence,

$$\mathrm{Hom}(K_2, G^n) \simeq (\mathrm{Hom}(K_2, G))^n \,.$$

Hom Complex

Hom-complex is a well-studied construction in topological combinatorics. [Lovász; '78]

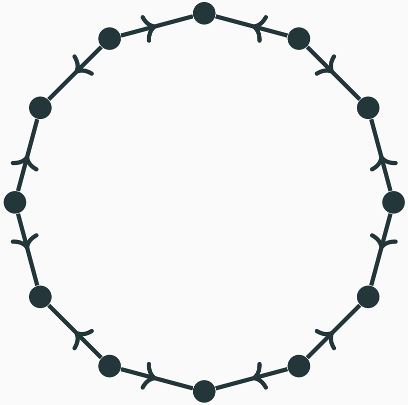
$\mathrm{Hom}(K_2, -)$ is a "recipe" to combinatorially build a topological space (simplicial set). The key properties we will use are:

1. For any G, $\mathrm{Hom}(K_2, G)$ inherits the $\mathbb{Z}_2$-symmetry of K_2 ($\mathbb{Z}_2$-action).
2. For any $G \to H$, there is a corresponding continuous map $\mathrm{Hom}(K_2, G) \to \mathrm{Hom}(K_2, H)$ respecting the symmetry.
3. Up to a (symmetry-preserving) topological equivalence,

$$\mathrm{Hom}(K_2, G^n) \simeq (\mathrm{Hom}(K_2, G))^n .$$

4. For $G = C_\ell$ and $G = K_4$, we can explicitly determine $\mathrm{Hom}(K_2, G)$.

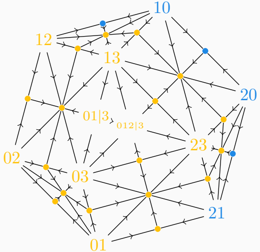
When $\ell \geq 3$ is odd, $\Gamma_{4\ell} := \mathrm{Hom}(K_2, C_\ell)$ is topologically the circle S^1.

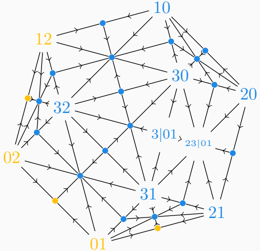


Hom(K_2, C_3)

$\mathrm{Hom}(K_2, K_4)$

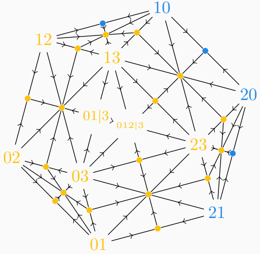
$\mathrm{Hom}(K_2, K_4)$ is topologically the sphere S^2,

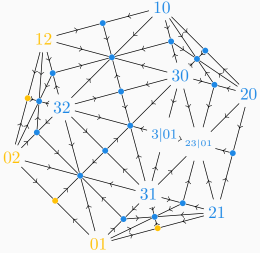


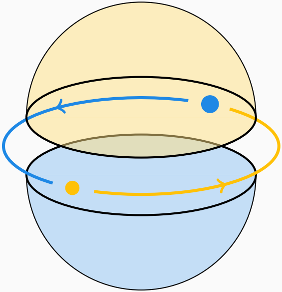


Hom(K_2, K_4)

Hom(K_2, K_4) is topologically the sphere S^2, triangulation can be simplified by $t : \text{Hom}(K_2, K_4) \to \Sigma^2$.







Building ϕ - Part I

Start with $f : C_\ell^n \to K_4$.

Building ϕ - Part I

Start with $f : C_\ell^n \to K_4$. $\mathrm{Hom}(K_2, -)$ induces symmetry-preserving map

$$f_* : \underbrace{\Gamma_{4\ell}^n = (\mathrm{Hom}(K_2, C_\ell))^n}_{T^n = (S^1)^n} \to \underbrace{\mathrm{Hom}(K_2, K_4) \to \Sigma^2}_{S^2}$$

Building ϕ - Part I

Start with $f : C_\ell^n \to K_4$. $\mathrm{Hom}(K_2, -)$ induces symmetry-preserving map

$$f_* : \underbrace{\Gamma_{4\ell}^n = (\mathrm{Hom}(K_2, C_\ell))^n}_{T^n=(S^1)^n} \to \underbrace{\mathrm{Hom}(K_2, K_4) \to \Sigma^2}_{S^2}$$

$\mathbb{Z}_2$-maps $T^n \to S^2$ are still too complicated; change S^2 to Y a "nicer" space (Eilenberg-MacLane space)

$$T^n = \Gamma_{4\ell}^n \xrightarrow{f_*} \Sigma^2 \longrightarrow Y, \qquad \mu(f) : T^n \to Y$$

Building ϕ - Part I

Start with $f : C_\ell^n \to K_4$. $\text{Hom}(K_2, -)$ induces symmetry-preserving map

$$f_* : \underbrace{\Gamma_{4\ell}^n = (\text{Hom}(K_2, C_\ell))^n}_{T^n = (S^1)^n} \to \underbrace{\text{Hom}(K_2, K_4) \to \Sigma^2}_{S^2}$$

$\mathbb{Z}_2$-maps $T^n \to S^2$ are still too complicated; change S^2 to Y a "nicer" space (Eilenberg-MacLane space)

$$T^n = \Gamma_{4\ell}^n \xrightarrow{f_*} \Sigma^2 \longrightarrow Y, \qquad \mu(f) : \Gamma_{4\ell}^n \to Y$$

We can classify $\mathbb{Z}_2$-maps $T^n \to Y$!

Monomial maps

Fix $S^1 = \{z \in \mathbb{C} \mid |z| = 1\}$ and $J \subseteq [n]$ with $|J|$ odd.

$[n] = \{1, \ldots, n\}$.

Monomial maps

Fix $S^1 = \{z \in \mathbb{C} \mid |z| = 1\}$ and $J \subseteq [n]$ with $|J|$ odd.
We have a $\mathbb{Z}_2$-map $m_J : T^n = (S^1)^n \to S^1 \hookrightarrow S^2 \subseteq Y$:

$$m_J(z_1, \ldots, z_n) \coloneqq \Pi_{j \in J} z_j$$

$[n] = \{1, \ldots, n\}$.

Monomial maps

Fix $S^1 = \{z \in \mathbb{C} \mid |z| = 1\}$ and $J \subseteq [n]$ with $|J|$ odd.
We have a $\mathbb{Z}_2$-map $m_J : T^n = (S^1)^n \to S^1 \hookrightarrow S^2 \subseteq Y$:

$$m_J(z_1, \ldots, z_n) := \Pi_{j \in J} z_j$$

Obs: Monomial maps correspond to linear maps $\mathbb{Z}_2^n \to \mathbb{Z}_2$.

$[n] = \{1, \ldots, n\}$.

Monomial maps

Fix $S^1 = \{z \in \mathbb{C} \mid |z| = 1\}$ and $J \subseteq [n]$ with $|J|$ odd.
We have a $\mathbb{Z}_2$-map $m_J : T^n = (S^1)^n \to S^1 \hookrightarrow S^2 \subseteq Y$:

$$m_J(z_1, \ldots, z_n) := \Pi_{j \in J} z_j$$

Obs: Monomial maps correspond to linear maps $\mathbb{Z}_2^n \to \mathbb{Z}_2$.
Prop 1: Different monomial maps are not equivalent.

$[n] = \{1, \ldots, n\}$.

Monomial maps

Fix $S^1 = \{z \in \mathbb{C} \mid |z| = 1\}$ and $J \subseteq [n]$ with $|J|$ odd.
We have a $\mathbb{Z}_2$-map $m_J : T^n = (S^1)^n \to S^1 \hookrightarrow S^2 \subseteq Y$:

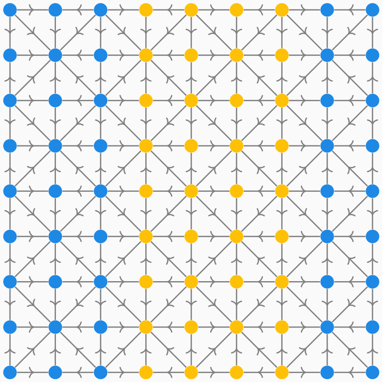
$$m_J(z_1, \ldots, z_n) := \Pi_{j \in J} z_j$$

Obs: Monomial maps correspond to linear maps $\mathbb{Z}_2^n \to \mathbb{Z}_2$.
Prop 1: Different monomial maps are not equivalent.
Prop 2: Any $\mathbb{Z}_2$-map $f : T^n \to Y$ is equivalent to a monomial map.

$[n] = \{1, \ldots, n\}$.

Building the minion homomorphism - Part II

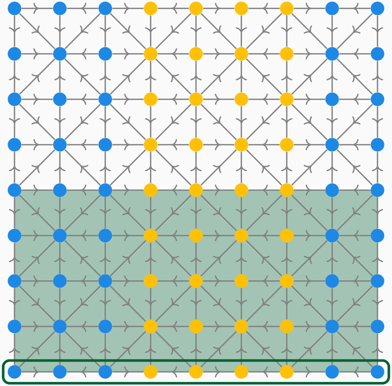
How do we associate to $f : T^n \to Y$ the right monomial map?



Building the minion homomorphism - Part II

Use *degree* in direction i. First, for binary $g : T^2 \to Y$:

$\deg_1(g) = \#\{[\bullet, \bullet] \text{ edges in } \square\} + \#\{[\bullet, \bullet, \bullet] \text{ triangles in } \blacksquare\} \pmod 2$

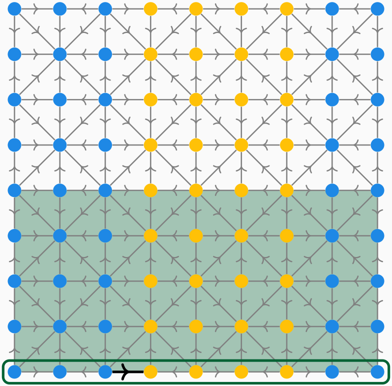


Building the minion homomorphism - Part II

Use *degree* in direction i. First, for binary $g : T^2 \to Y$:

$\deg_1(g) = \#\{[\bullet, \bullet]$ edges in $\square\} + \#\{[\bullet, \bullet, \bullet]$ triangles in $\blacksquare\}$ (mod 2)

Ex: $\deg_1(g_1) = 1 + 0$

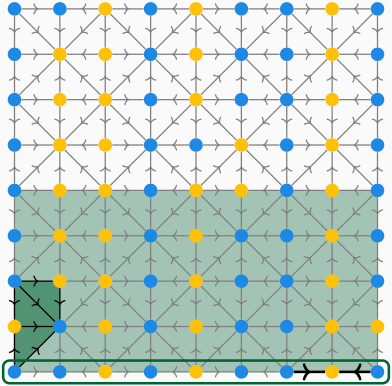


Building the minion homomorphism - Part II

Use *degree* in direction i. First, for binary $g : T^2 \to Y$:

$\deg_1(g) = \#\{[\bullet, \bullet]$ edges in $\square\} + \#\{[\bullet, \bullet, \bullet]$ triangles in $\blacksquare\}\pmod 2$

Ex: $\deg_1(g_2) = 2 + 3 \equiv 1$



Building the minion homomorphism - Part II

Use *degree* in direction i. First, for binary $g : T^2 \to Y$:

$\deg_1(g) = \#\{[\bullet, \bullet] \text{ edges in } \square\} + \#\{[\bullet, \bullet, \bullet] \text{ triangles in } \blacksquare\} \pmod 2$

For $f : T^n \to Y$, set $g(x, y) = f(y, \ldots, y, x, y, \ldots, y)$, then:

$$\deg_i(f) := \deg_1(g)$$

Building the minion homomorphism - Part II

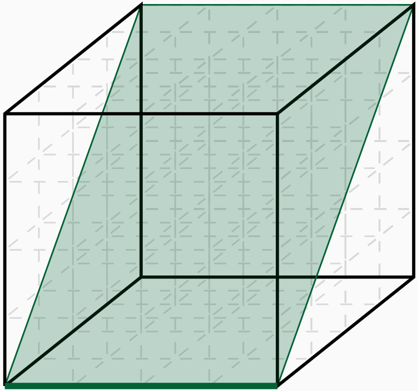
Use *degree* in direction i. First, for binary $g : T^2 \to Y$:

$\deg_1(g) = \#\{[\bullet, \bullet] \text{ edges in } \square\} + \#\{[\bullet, \bullet, \bullet] \text{ triangles in } \blacksquare\} \pmod 2$

For $f : T^n \to Y$, set $g(x, y) = f(y, \ldots, y, x, y, \ldots, y)$, then:

$$\deg_i(f) := \deg_1(g)$$

Ex: $\deg_1(f)$



Building the minion homomorphism - Part II

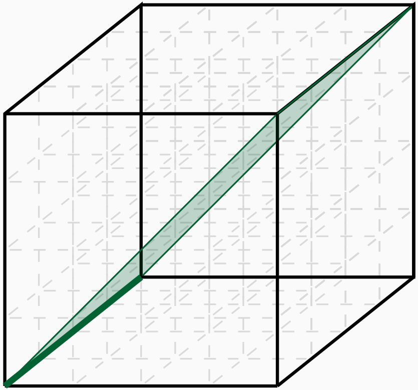
Use *degree* in direction i. First, for binary $g : T^2 \to Y$:

$\deg_1(g) = \#\{[\bullet, \bullet] \text{ edges in } \square\} + \#\{[\bullet, \bullet, \bullet] \text{ triangles in } \blacksquare\} \pmod 2$

For $f : T^n \to Y$, set $g(x, y) = f(y, \ldots, y, x, y, \ldots, y)$, then:

$$\deg_i(f) \coloneqq \deg_1(g)$$

Ex: $\deg_2(f)$



Building the minion homomorphism - Part II

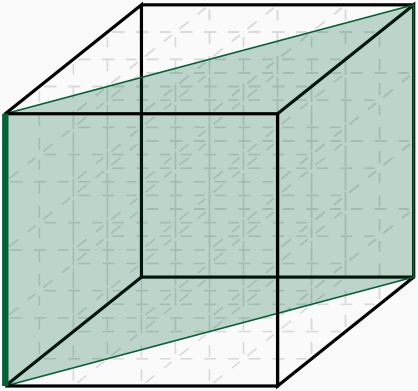
Use *degree* in direction i. First, for binary $g : T^2 \to Y$:

$\deg_1(g) = \#\{[\bullet, \bullet] \text{ edges in } \square\} + \#\{[\bullet, \bullet, \bullet] \text{ triangles in } \blacksquare\} \pmod 2$

For $f : T^n \to Y$, set $g(x, y) = f(y, \ldots, y, x, y, \ldots, y)$, then:

$$\deg_i(f) \coloneqq \deg_1(g)$$

Ex: $\deg_3(f)$

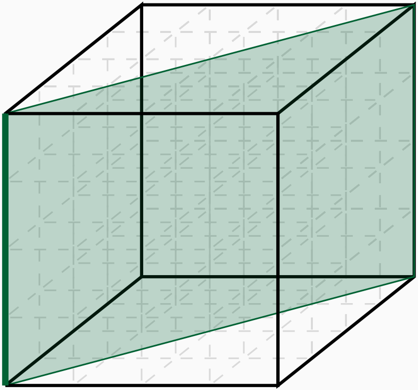


Building the minion homomorphism - Part II

Use *degree* in direction i. First, for binary $g : T^2 \to Y$:

$\deg_1(g) = \#\{[\bullet, \bullet] \text{ edges in } \square\} + \#\{[\bullet, \bullet, \bullet] \text{ triangles in } \blacksquare\} \pmod 2$

Finally, define $\phi : f \mapsto (\deg_1(f), \ldots, \deg_n(f))$.



What's next?

- What can we say $\mathrm{PCSP}(G, K_5)$? It requires new topological ideas.

What's next?

- What can we say $\text{PCSP}(G, K_5)$? It requires new topological ideas.
- Can topology work with non-symmetric templates (e.g. digraphs)?

What's next?

- What can we say $\mathrm{PCSP}(G, K_5)$? It requires new topological ideas.
- Can topology work with non-symmetric templates (e.g. digraphs)?
- Which templates can be studied topologically?

What's next?

- What can we say PCSP(G, K_5)? It requires new topological ideas.
- Can topology work with non-symmetric templates (e.g. digraphs)?
- Which templates can be studied topologically?
- More category theory (e.g. generalized nerve functors instead of Hom)?

Thank you!

Combinatorial Constraints

Obs 1: Simplicial maps $X \to \Sigma^2 \leftrightharpoons$ 2-colourings of the vertices such that every simplex has at most two alternations.

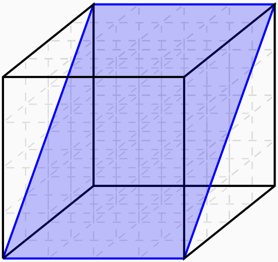
Combinatorial Constraints

Obs 1: Simplicial maps $X \to \Sigma^2 \leftrightharpoons$ 2-colourings of the vertices such that every simplex has at most two alternations.
Obs 2: Any simplicial map $f : \Gamma_L^2 \to \Sigma^2$ with $\deg_1(f) = 1$ has to have at least one horizontal color swapping edges; i.e. at least one horizontal edge in the torus is not monochrome.

Combinatorial Constraints

Obs 1: Simplicial maps $X \to \Sigma^2 \leftrightharpoons$ 2-colourings of the vertices such that every simplex has at most two alternations.
Obs 2: Any simplicial map $f : \Gamma_L^2 \to \Sigma^2$ with $\deg_1(f) = 1$ has to have at least one horizontal color swapping edges; i.e. at least one horizontal edge in the torus is not monochrome.

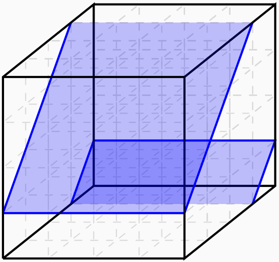
By slicing, we get that at least a $1/CL^2$ fraction of horizontal edges is color swapping.



Combinatorial Constraints

Obs 1: Simplicial maps $X \to \Sigma^2 \leftrightharpoons$ 2-colourings of the vertices such that every simplex has at most two alternations.
Obs 2: Any simplicial map $f : \Gamma_L^2 \to \Sigma^2$ with $\deg_1(f) = 1$ has to have at least one horizontal color swapping edges; i.e. at least one horizontal edge in the torus is not monochrome.

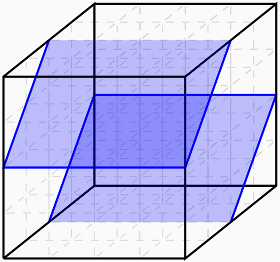
By slicing, we get that at least a $1/CL^2$ fraction of horizontal edges is color swapping.



Combinatorial Constraints

Obs 1: Simplicial maps $X \to \Sigma^2 \leftrightharpoons$ 2-colourings of the vertices such that every simplex has at most two alternations.
Obs 2: Any simplicial map $f : \Gamma_L^2 \to \Sigma^2$ with $\deg_1(f) = 1$ has to have at least one horizontal color swapping edges; i.e. at least one horizontal edge in the torus is not monochrome.

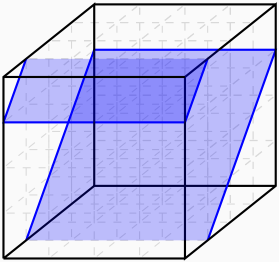
By slicing, we get that at least a $1/CL^2$ fraction of horizontal edges is color swapping.



Combinatorial Constraints

Obs 1: Simplicial maps $X \to \Sigma^2 \leftrightharpoons$ 2-colourings of the vertices such that every simplex has at most two alternations.
Obs 2: Any simplicial map $f : \Gamma_L^2 \to \Sigma^2$ with $\deg_1(f) = 1$ has to have at least one horizontal color swapping edges; i.e. at least one horizontal edge in the torus is not monochrome.

By slicing, we get that at least a $1/CL^2$ fraction of horizontal edges is color swapping.



Bounding Essential Arity

Goal: Show that any simplicial map $f : \Gamma_L^n \to \Sigma^2$ can have at most $O(L^2)$ essential coordinates (i.e. $\deg_i(f) \neq 0$).

Bounding Essential Arity

Goal: Show that any simplicial map $f : \Gamma_L^n \to \Sigma^2$ can have at most $O(L^2)$ essential coordinates (i.e. $\deg_i(f) \neq 0$).

Proof: We can assume (up to taking a suitable minor) that every coordinate of f is essential. Choose a non degenerate n-simplex $\sigma = [u_0, \ldots, u_n]$ uniformly at random. Let $X_i(\sigma)$ the random variable that is 1 if $[u_i, u_{i+1}]$ is colour swapping, 0 otherwise.

Bounding Essential Arity

Goal: Show that any simplicial map $f : \Gamma_L^n \to \Sigma^2$ can have at most $O(L^2)$ essential coordinates (i.e. $\deg_i(f) \neq 0$).

Proof: We can assume (up to taking a suitable minor) that every coordinate of f is essential. Choose a non degenerate n-simplex $\sigma = [u_0, \ldots, u_n]$ uniformly at random. Let $X_i(\sigma)$ the random variable that is 1 if $[u_i, u_{i+1}]$ is colour swapping, 0 otherwise.

$$E\left[\sum_i X_i(\sigma)\right] = \sum_i E\left[X_i(\sigma)\right] \geq n\frac{1}{CL^2}.$$

Bounding Essential Arity

Goal: Show that any simplicial map $f : \Gamma_L^n \to \Sigma^2$ can have at most $O(L^2)$ essential coordinates (i.e. $\deg_i(f) \neq 0$).

Proof: We can assume (up to taking a suitable minor) that every coordinate of f is essential. Choose a non degenerate n-simplex $\sigma = [u_0, \ldots, u_n]$ uniformly at random. Let $X_i(\sigma)$ the random variable that is 1 if $[u_i, u_{i+1}]$ is colour swapping, 0 otherwise.

$$E\left[\sum_i X_i(\sigma)\right] = \sum_i E\left[X_i(\sigma)\right] \geq n\frac{1}{CL^2}.$$

However, $\sum_i X_i(\sigma)$ is the total number of colour swaps in σ, thus $\sum_i X_i(\sigma) \leq 2$ since f is simplicial.

Bounding Essential Arity

Goal: Show that any simplicial map $f : \Gamma_L^n \to \Sigma^2$ can have at most $O(L^2)$ essential coordinates (i.e. $\deg_i(f) \neq 0$).

Proof: We can assume (up to taking a suitable minor) that every coordinate of f is essential. Choose a non degenerate n-simplex $\sigma = [u_0, \ldots, u_n]$ uniformly at random. Let $X_i(\sigma)$ the random variable that is 1 if $[u_i, u_{i+1}]$ is colour swapping, 0 otherwise.

$$E\left[\sum_i X_i(\sigma)\right] = \sum_i E\left[X_i(\sigma)\right] \geq n\frac{1}{CL^2}.$$

However, $\sum_i X_i(\sigma)$ is the total number of colour swaps in σ, thus $\sum_i X_i(\sigma) \leq 2$ since f is simplicial.

Therefore,

$$n \leq 2CL^2$$

that is, any function in the image of ϕ has essential arity at most $O(L^2) \Rightarrow \mathrm{Im}\phi$ has bounded essential arity.