The complexity of 3-colouring **H**-colourable graphs

Andrei Krokhin and **Jakub Opršal**

60th Annual Symposium on Foundations of Computer Science, 2019

Durham University

Is there a polynomial time algorithm that colours a given 3-colourable graph by 3 colours?

Is there a polynomial time algorithm that colours a given 3-colourable graph by 3 colours?

No! (Unless P $=$ NP) [Karp, "72]

Is there a polynomial time algorithm that colours a given 3-colourable graph by 4 colours?

No! (Unless P $=$ NP) [Khanna, Linian, Safra, '00]

Is there a polynomial time algorithm that colours a given 3-colourable graph by 5 colours?

No! (Unless P $=$ NP) [Bulín, Krokhin, __, ‘19]

Is there a polynomial time algorithm that colours a given 3-colourable graph by 6 colours?

(We don’t know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by 7 colours?

(We don't know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by 1729 colours?

(We don't know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by 2^{259} colours?

(We don't know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by $O(\log n)$ colours?

(We don't know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by $O(n^{<1/5})$ colours?

Yes! [Kawarabayashi, Thorup, '17]

Is there a polynomial time algorithm that colours a given 3-colourable graph by $O(\log n)$ colours?

(We don't know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by 2^{259} colours?

(We don't know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by 1729 colours?

(We don't know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by 7 colours?

(We don't know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by 6 colours?

(We don’t know.)

Is there a polynomial time algorithm that colours a given 3-colourable graph by 5 colours?

No! (Unless P $=$ NP) [Bulín, Krokhin, __, '19]

Is there a polynomial time algorithm that colours a given 3-colourable graph by 4 colours?

No! (Unless P $=$ NP) [Khanna, Linian, Safra, '00]

Is there a polynomial time algorithm that colours a given 3-colourable graph by 3 colours?

No! (Unless P $=$ NP) [Karp, “72]

Is there a polynomial time algorithm that colours a given 2-colourable graph by 3 colours?

Yes. (easy)

Is there a polynomial time algorithm that colours a given $(2+\epsilon)$-colourable graph by 3 colours?

REAL GOAL

Create new tools for showing lower bounds approximate graph colouring.

Graph homomorphism problem

Given two graphs G and H, decide whether there is a graph homomorphism (edge-preserving map) from G to H.

$f: V(G) \to V(H)$ is a graph homomorphism from G to H if $(f(u), f(v)) \in E(H)$ for each $(u, v) \in E(G)$.

- If we fix H, we get so-called H-colouring problem.
- K_c-colouring is just c-colouring.

Theorem [Hell, Nešetřil, "90].

For any non-bipartite loop-less graph H, H-colouring is NP-complete.

Promise graph homomorphism

Question.

How hard is to find a homomorphism to G given an H-colourable graph?

Conjecture [Brakensiek, Guruswami, '16].

This problem is NP-hard for any G, H such that $H \to G$ and both are loopless and bipartite.

Theorem.

Let $\mathbf{H}$ be a 3-colourable non-bipartite graph. Then finding a colouring of a given H-colourable graph with 3 colours is NP-hard.

Promise graph homomorphism

Question.

How hard is to colour a given $(2+\epsilon)$-colourable graph by c colours?

Conjecture [Brakensiek, Guruswami, '16].

This problem is NP-hard for any $\epsilon > 0$ and $c \geq 2+\epsilon$.

Theorem.

This problem is NP-hard for $c = 3$ and any $\epsilon > 0$.

Formalized by so-called circular chromatic number.

(Promise) constraint satisfaction

- (non-uniform) CSP can be viewed as H-colouring generalized to arbitrary relational structures.
- H-colouring was one of the base cases for the CSP dichotomy conjecture [Feder, Vardi,“98] (a.k.a. Bulatov-Zhuk theorem [Bulatov, ‘17; Zhuk, ‘17]).
- Promise CSP includes e.g. so-called $(2+\epsilon)$-SAT [Austrin, Guruswami, Håstad, ‘17].
- A (special) theory of gadget reductions for promise CSP (using universal algebra) is described in [Barto, Bulín, Krokhin, __, ‘19].

Methods

Step 1. Label Cover

A label cover instance is a tuple $(U, V, l, r, E \subseteq U \times V, \{\pi_e : [l] \to [r] \mid e \in E\})$.

m-List Label Cover

- Assuming there is an assignment $s\colon U \to [l], V \to [r]$ such that $\pi_{(u,v)}(s(u)) = s(v)$ for each $(u, v) \in E$,
- find $S\colon U \to \binom{[l]}{m}, V \to \binom{[r]}{m}$ such that $\pi_{(u,v)}(S(u)) \cap S(v) \neq \emptyset$ for each $(u, v) \in E$.

Theorem [Arora, Safra, et al., "98; Raz, '01 + Folklore].

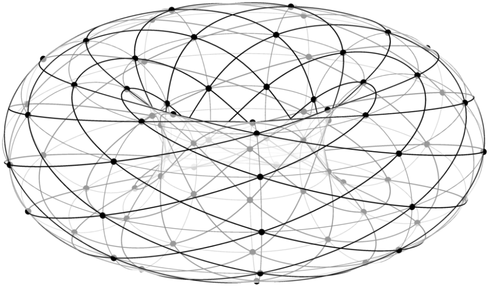
For each $m \geq 1$, m-List Label Cover is NP-hard.

Step 2. Gadgets

The n-th (tensor) power of a graph G is the graph G^n with $V(G^n) = V(G)^n$ and

$$((u_1, \dots, u_n), (v_1, \dots, v_n)) \in E(G^n) \Leftrightarrow (u_i, v_i) \in E(G) \text{ for all } i.$$

$C_9^2 =$



Step 2. Gadgets

Assuming H contains an odd cycle C_{2k+1}. From an LC instance construct a graph G

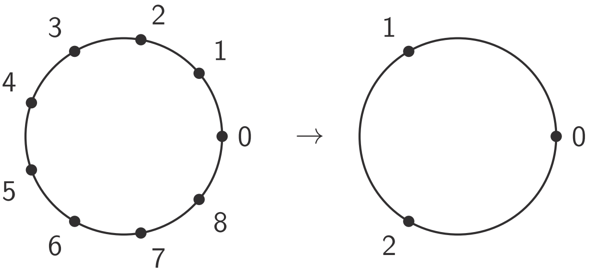
- Replace each $u \in U$ with C^{l}_{2k+1} and $v \in V$ with C^{r}_{2k+1},
- collapse some vertices according to π_e's.

soundness. Given a solvable instance of LC, find a homomorphism from G to H.

completeness. Given a 3-colouring of G, find a list of m candidates for each variable of the LC instance.

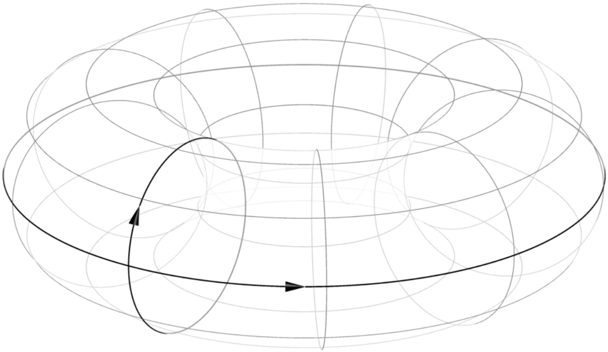
Identify at most m important (influential, essential, ...) coordinates of a given $C^{n}_{2k+1} \to K_3$!

Step 3. Topology

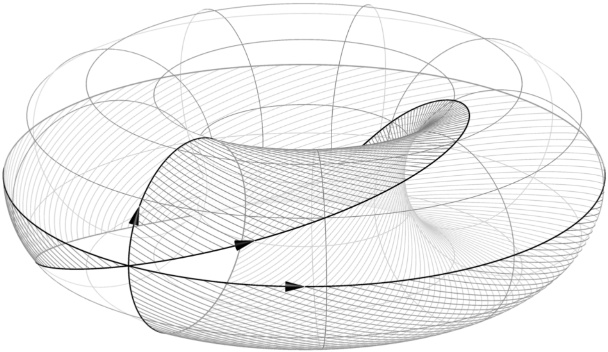


- Count how many times you 'walk around'.
- We are interested in continuous mappings from T^n to S^1.

Step 3. Topology



Step 3. Topology



Conclusions

Theorem.
*Let **H** be a 3-colourable non-bipartite graph. Then finding a colouring of a given H-colourable graph with 3 colours is NP-hard.*

Theorem.
It is NP-hard to colour a given $(2+\epsilon)$-colourable graph with 3 colours.

Theorem [Wrochna, Živný, SODA'20].
It is NP-hard to $(4-\epsilon)$-colour a given $(2+\epsilon)$-colourable graph.

■