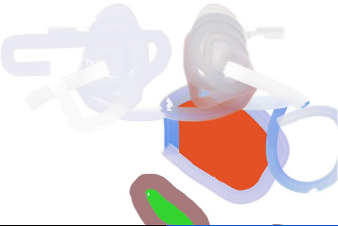
WL50 * Pilsen * July 7, 2018

Permutation groups and Graph Isomorphism: Local certificates

László Babai
University of Chicago



Graph Isomorphism: the most recent bounds

Input: graphs X, Y with n vertices
Question: $X \cong Y$?

trivial bound $n!$

$\exp(O(\sqrt{n \log n}))$ Luks 1983

moderately exponential

Graph Isomorphism: the most recent bounds

Input: graphs X, Y with n vertices
Question: $X \cong Y$?

trivial bound $n!$

$\exp(O(\sqrt{n \log n}))$ Luks 1983

moderately exponential

$\exp((\log n)^{O(1)})$ this talk

quasipolynomial

Graph Isomorphism is equivalent to
finding orbits of automorphism group

Graph Isomorphism is equivalent to
finding orbits of automorphism group

QUESTION Is there an efficiently computable relaxation of symmetry that is complete – implies symmetry?

The job

Close the gap between

SYMMETRY and **REGULARITY**

Combinatorial relaxations do not suffice

Cai, Furer, Immerman 1992

Ingredients: symmetry AND regularity

- PART 1 — **group theory** (symmetry)
  - finite permutation groups
  - "local to global tool"
- PART 2 — **coherent configurations**

  (regularity)

  "divide-and-conquer tool"

  (efficient recurrence)

Let $G \leq \mathrm{Sym}(\Omega)$ be a **permutation group**, $|\Omega| = n$
stabilizer of $x \in \Omega$: $G_x = \{\sigma \in G \mid x^\sigma = x\}$ (fixes x)

DEF: Let $\varphi : G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ be a homomorphism **onto** the alternating group (even permutations) of a set Γ, $|\Gamma| = m$
$x \in \Omega$ is **affected** by φ if $\varphi(G_x) \neq \mathrm{Alt}(\Gamma)$

Let $G \leq \mathrm{Sym}(\Omega)$ be a **permutation group**, $|\Omega| = n$
stabilizer of $x \in \Omega$: $G_x = \{\sigma \in G \mid x^\sigma = x\}$ (fixes x)

Theorem (Unaffected stabilizers lemma)

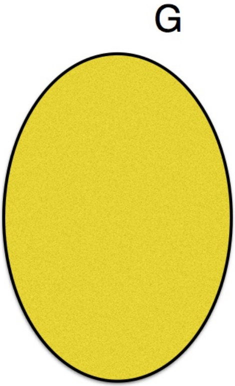
Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

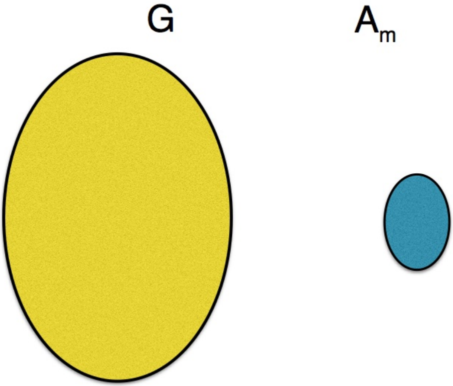
If $m > \max\{8, 2 + \log_2 n\}$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

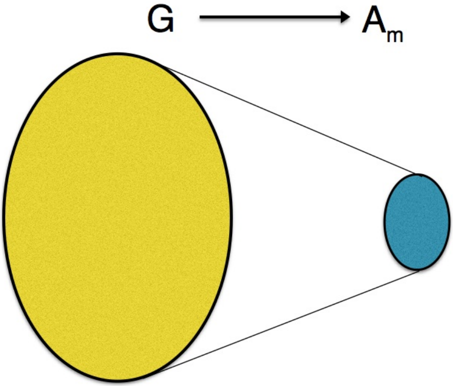
Unaffected stabilizers lemma



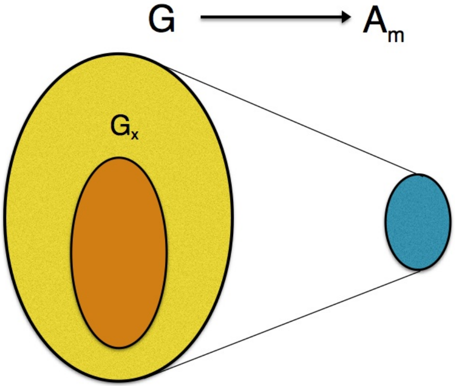
Unaffected stabilizers lemma



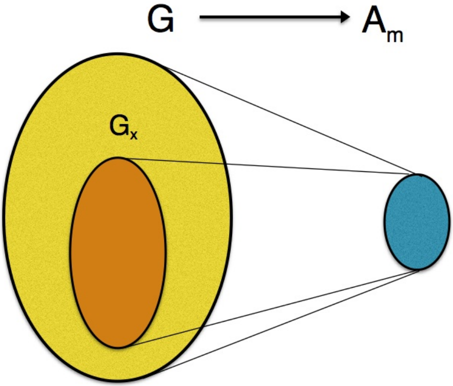
Unaffected stabilizers lemma



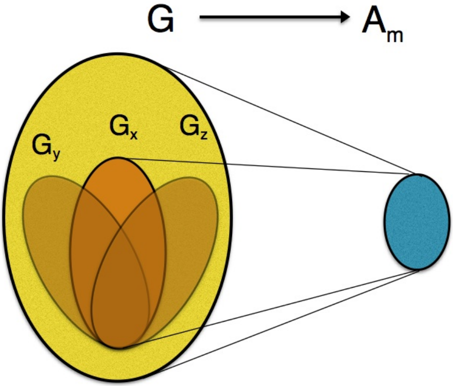
Unaffected stabilizers lemma



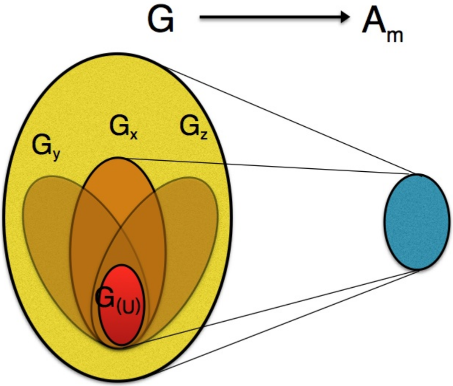
Unaffected stabilizers lemma



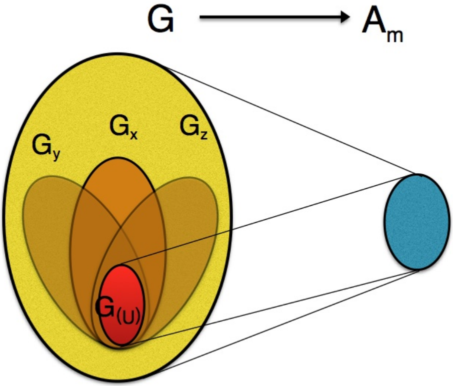
Unaffected stabilizers lemma



Unaffected stabilizers lemma



Unaffected stabilizers lemma



Unaffected stabilizers lemma

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

Unaffected stabilizers lemma

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

Corollary. At least one point is affected.

Unaffected stabilizers lemma

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

Corollary. At least one point is affected.

Result **tight:** $\exists$ infinitely many examples with $m = 2 + \log_2 n$ with **no affected points.**

Unaffected stabilizers lemma: tight

Corollary. At least one point is affected.

Result **tight:** $\exists$ infinitely many examples with $m = 2 + \log_2 n$ with **no affected points.**

Unaffected stabilizers lemma: tight

Corollary. At least one point is affected.

Result **tight:** $\exists$ infinitely many examples with
$m = 2 + \log_2 n$ with **no affected points.**

Example

$A_m \leq \mathrm{GL}(m, 2)$ – permute coordinates

$G := \mathbb{F}_2^m \rtimes A_m \leq \mathrm{AGL}(m, 2)$

acting on $\mathbb{F}_2^m$ so $n = 2^m$, i.e., $m = \log_2 n$

$G \twoheadrightarrow A_m$ and $G_0 \cong A_m$ and $(\forall x)(G_x \cong A_m)$

reduce dim by 2

take $A_m \leq \mathrm{GL}(m, 2)$ restict to $\sum x_i = 0$

quotient by $x_1 = \cdots = x_m = 1$ if m even

$\Longrightarrow A_m \leq \mathrm{AGL}(m-2, 2) \Longrightarrow m = 2 + \log_2 n$

Unaffected stabilizers lemma

DEF: Let $\varphi : G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ be a homomorphism **onto** the alternating group (even permutations) of a set Γ, $|\Gamma| = m$

$x \in \Omega$ is **affected** by φ if $\varphi(G_x) \neq \mathrm{Alt}(\Gamma)$

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

Unaffected stabilizers lemma

DEF: Let $\varphi : G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ be a homomorphism **onto** the alternating group (even permutations) of a set Γ, $|\Gamma| = m$

$x \in \Omega$ is **affected** by φ if $\quad \varphi(G_x) \neq \mathrm{Alt}(\Gamma)$

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

Uses Classification of Finite Simple Groups (CFSG) through *Schreier's Hypothesis:* "The outer automorphism group of every finite simple group is solvable."

Unaffected stabilizers lemma

DEF: Let $\varphi : G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ be a homomorphism **onto** the alternating group (even permutations) of a set Γ, $|\Gamma| = m$

$x \in \Omega$ is **affected** by φ if $\quad \varphi(G_x) \neq \mathrm{Alt}(\Gamma)$

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

Uses Classification of Finite Simple Groups (CFSG) through *Schreier's Hypothesis:* "The outer automorphism group of every finite simple group is solvable."

Pyber recently removed CFSG assuming $m > (\log n)^c$.

Unaffected stabilizers lemma

DEF: Let $\varphi : G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ be a homomorphism **onto** the alternating group (even permutations) of a set Γ, $|\Gamma| = m$

$x \in \Omega$ is **affected** by φ if $\quad \varphi(G_x) \neq \mathrm{Alt}(\Gamma)$

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

Proof: Induction. Base case: G – primitive group.

Unaffected Stabilizers: primitive case

Lemma (Primitive case)

Let $G \le S_n$ *be a primitive group and* $\varphi : G \twoheadrightarrow A_m$ *for some* $m > \max\{8, 2 + \log_2 n\}$*. Then* φ *is an isomorphism.*

Proof of lemma depends on *Schreier's Hypothesis*

Tool: O'Nan–Scott–Aschbacher Structure Thm
for primitive groups

Unaffected stabilizers lemma

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

Unaffected stabilizers lemma

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) = \mathrm{Alt}(\Gamma).$$

HOW IS THIS RELATED TO

Graph Isomorphism ?

String Isomorphism: anagrams via a permutation group

string: $\mathbf{x} : \Omega \to ABC$ (Ω: set of positions)

$\sigma \in \mathrm{Sym}(\Omega)$ transforms strings $\mathbf{x} \mapsto \mathbf{x}^\sigma$

permutation group $G \le \mathrm{Sym}(\Omega)$

subgroup of the symmetric group acting on Ω

G given by a list of generators

strings $\mathbf{x}, \mathbf{y}$ are **G-isomorphic**: $\mathbf{x} \cong_G \mathbf{y}$ if $(\exists \sigma \in G)(\mathbf{x}^\sigma = \mathbf{y})$

String Isomorphism: anagrams via a permutation group

string: $\mathbf{x} : \Omega \to ABC$ (Ω: set of positions)

$\sigma \in \mathrm{Sym}(\Omega)$ transforms strings $\mathbf{x} \mapsto \mathbf{x}^\sigma$

permutation group $G \leq \mathrm{Sym}(\Omega)$

subgroup of the symmetric group acting on Ω

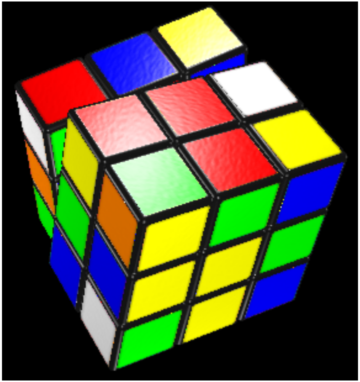
G given by a list of generators

strings $\mathbf{x}, \mathbf{y}$ are **G-isomorphic:** $\mathbf{x} \cong_G \mathbf{y}$ if $(\exists \sigma \in G)(\mathbf{x}^\sigma = \mathbf{y})$

String Isomorphism problem (Luks 1980/82):

Given G, $\mathbf{x}, \mathbf{y}$ decide: $\mathbf{x} \cong_G \mathbf{y}$?

String Isomorphism



Can a given coloring of Rubik's cube be transformed into another one via legal moves?

strings $\mathbf{x}, \mathbf{y}$ are *G*-isomorphic: $\mathbf{x} \cong_G \mathbf{y}$ if $(\exists \sigma \in G)(\mathbf{x}^\sigma = \mathbf{y})$

String Isomorphism problem (Luks 1980/82):

Given G, $\mathbf{x}, \mathbf{y}$ decide: $\mathbf{x} \cong_G \mathbf{y}$?

Theorem

String Isomorphism decidable in quasipolynomial time.

Previous best: $\exp(\widetilde{O}(\sqrt{n}))$ (B 1983)

dead end

Reducing GI to SI

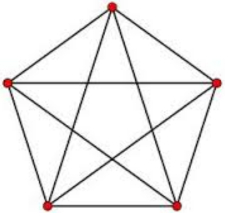
Graph X with n vertices encoded as
(0, 1)-string $\mathbf{x}(X)$ of length $\binom{n}{2}$
edge-subset of the complete graph K_n

$$G = S_n^{(2)} \leq S_{\binom{n}{2}}$$

$S_n^{(2)}$ — **induced symmetric group on pairs**

$S_n^{(2)}$ — action of S_n on $E(K_n)$

$$S_n^{(2)} \cong S_n$$



Reducing GI to SI

Graph X with n vertices encoded as

$(0,1)$-string $\mathbf{x}(X)$ of length $\binom{n}{2}$

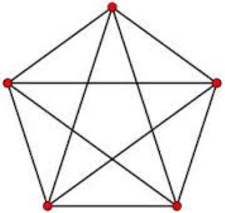
edge-subset of the complete graph K_n

$$G = S_n^{(2)} \leq S_{\binom{n}{2}}$$

$S_n^{(2)}$ — **induced symmetric group on pairs**

$S_n^{(2)}$ — action of S_n on $E(K_n)$

$$S_n^{(2)} \cong S_n$$



$$X \cong Y \iff \mathbf{x}(X) \cong_G \mathbf{x}(Y)$$

Hypergraphs to strings

t-uniform hypergraph X with n vertices encoded as
$(0,1)$-string $\mathbf{x}(X)$ of length $\binom{n}{t}$

$G = S_n^{(t)} \le S_{\binom{n}{t}}$ **induced symmetric group on t-tuples**

$S_n^{(t)}$ — **"Johnson groups"**

$S_n^{(t)} \cong S_n$ acts on $\binom{n}{t}$ t-tuples

X, Y t-uniform hypergraphs

$$X \cong Y \iff \mathbf{x}(X) \cong_G \mathbf{x}(Y)$$

The Luks equivalence class

The following decision problems are **equivalent** under Karp reductions (polynomial-time many-one reductions) to **String Isomorphism:**

INPUT: $G, H \leq \mathrm{Sym}(\Omega)$ and $\sigma, \tau \in \mathrm{Sym}(\Omega)$

Questions:

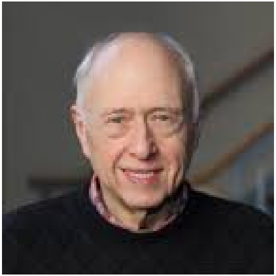
- **Coset intersection:** is $G\sigma \cap H\tau \neq \emptyset$?
- **Centralizer in coset:**
  Is the centralizer of τ in $G\sigma$ not empty?
- **Double coset membership:** is $\tau \in G\sigma H$?

In-depth use of group theory

Eugene M. Luks

Isomorphism of graphs of bouded valence can be tested in polynomial time

FOCS 1980, JCSS 1982



single most important paper ever on GI

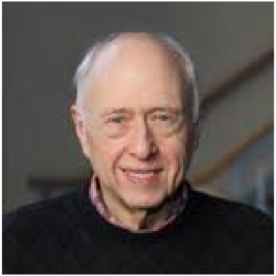
group theoretic divide-and-conquer method

In-depth use of group theory

Eugene M. Luks

Isomorphism of graphs of bouded valence can be tested in polynomial time

FOCS 1980, JCSS 1982



single most important paper ever on GI

group theoretic divide-and-conquer method

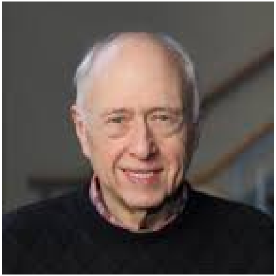
What we don't use: Luks's **result**

In-depth use of group theory

Eugene M. Luks

Isomorphism of graphs of bouded valence can be tested in polynomial time

FOCS 1980, JCSS 1982



single most important paper ever on GI

group theoretic divide-and-conquer method

What we don't use: Luks's **result**

What we do use: Luks's **method**

wired into the genes of the new algorithm

New ingredients

1. Group theory: “Unaffected stabilizers lemma”
local-to-global tool ♡

2. Group theoretic algorithms

2a. “local certificates” ♡

2b. aggregation of the “fullness certificates”

3. Combinatorial algorithms

canonical partitioning/reduction procedures :
combinatorial divide-and-conquer tools

3a. Design Lemma

3b. “Split-or-Johnson”

15-second overview of Luks's method

$$\mathrm{ISO}_G(\mathbf{x},\mathbf{y}) := \{\sigma \in G \mid \mathbf{x}^\sigma = \mathbf{y}\}$$

Divide-and-Conquer strategy: recursion on G

- Reduce to orbits — superfast recurrence
- Descend to subgroup
  — multiplicative cost: index of subgroup
- typically, descend to kernel of action on
  blocks of imprimitivity

Luks barrier

Fact **Either** efficient Luks reduction found
or epimorphism $G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ found
for some large $m = |\Gamma|$

Luks barrier

Fact **Either** efficient Luks reduction found

or epimorphism $G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ found

for some large $m = |\Gamma|$

complexity: $m = |\Gamma|$ **goes into exponent**

barrier case: $m >$ polylog(n)

Proof by **Cameron's** classification of

large primitive groups: $|G| > n^{1+\log_2 n}$ (1981)

(heavily depends on CFSG)

Luks barrier

Socle Soc(G): product of minimal normal subgroups

Theorem (Cameron (1981) as sharpened by Attila Maróti (2002))

If $G \le S_n$ primitive, $|G| > n^{1+\log_2 n}$, then $\mathrm{Soc}(G) \cong A_m^\ell$ *acting as Johnson groups in the product action on* $\binom{m}{t}^\ell$ *and*

$$(A_m^{(t)})^\ell \le G \le S_m^{(t)} \wr S_\ell$$

All we need from this is that

- $\mathrm{Soc}(G) \cong A_m^\ell$ and $|G : \mathrm{Soc}(G)| \le 2^\ell \ell!$;
- $n \ge m^\ell$ and $(m!)^\ell \ell! > n^{1+\log_2 n}$

Corollary (calculation): $|G : \mathrm{Soc}(G)| \le n$
Luks descends to socle: multipl cost $\le n$
now $G \cong A_m^\ell \twoheadrightarrow A_m$

Luks strategy, barrier

Fact **Either** efficient Luks reduction found

or epimorphism $G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ found

for some large $m = |\Gamma|$

complexity: $m = |\Gamma|$ **goes into exponent**

barrier case: $m > \mathrm{polylog}(n)$

Ω actual domain Γ **"ideal domain"**

Luks strategy, barrier

Fact **Either** efficient Luks reduction found

or epimorphism $G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ found

for some large $m = |\Gamma|$

complexity: $m = |\Gamma|$ **goes into exponent**

barrier case: $m > \mathrm{polylog}(n)$

Ω actual domain Γ **"ideal domain"**

Luks continues to CONQUER

Luks strategy, barrier

Fact **Either** efficient Luks reduction found

or epimorphism $G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ found

for some large $m = |\Gamma|$

complexity: $m = |\Gamma|$ **goes into exponent**

barrier case: $m > \mathrm{polylog}(n)$

Ω actual domain Γ **"ideal domain"**

Luks continues to CONQUER

as long as *somebody* DIVIDES

Luks strategy, barrier

Fact **Either** efficient Luks reduction found

or epimorphism $G \twoheadrightarrow \mathrm{Alt}(\Gamma)$ found

for some large $m = |\Gamma|$

complexity: $m = |\Gamma|$ **goes into exponent**

barrier case: $m > \mathrm{polylog}(n)$

Ω actual domain Γ **"ideal domain"**

Luks continues to CONQUER

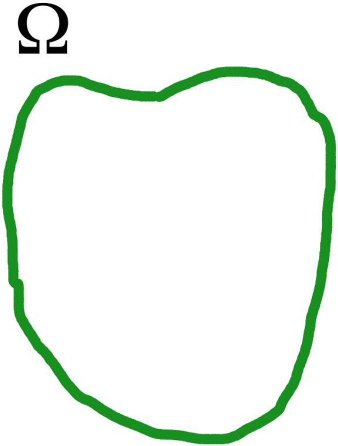
as long as *somebody* DIVIDES

somebody $\leftarrow$ new group theory + combinatorics

algorithmic
GROUP THEORY (Luks+new)
and
COMBINATORICS (new)
DIVIDE

algorithmic
GROUP THEORY (Luks)
CONQUERS

Strings



Ω set of positions

Strings

Ω

$\mathbf{x} : \Omega \to ABC$ string

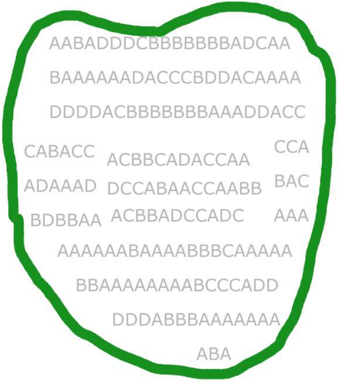
Strings

string ignored, focus on $G \leq \mathrm{Sym}(\Omega)$

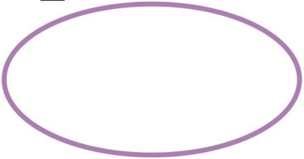
Luks barrier

$$\varphi : G \twoheadrightarrow \mathrm{Alt}(\Gamma)$$

Ω



Γ

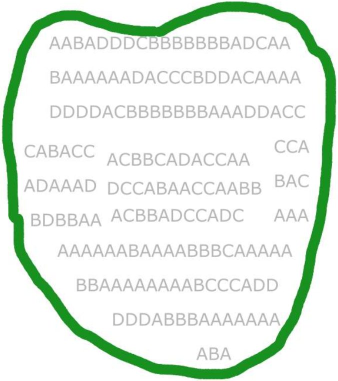


Γ: **ideal domain**

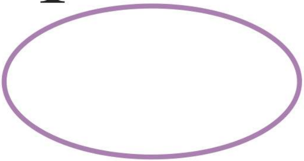
Luks barrier

$$\varphi : G \twoheadrightarrow \mathrm{Alt}(\Gamma)$$

Ω



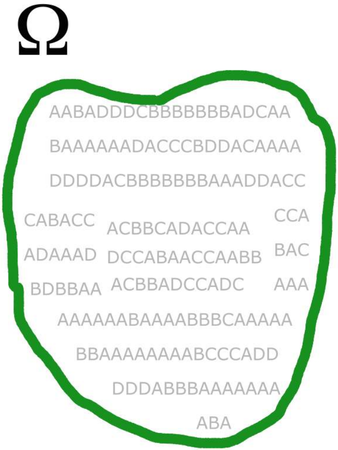
Γ

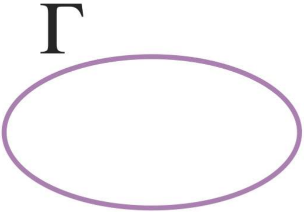


Γ: **ideal domain**

Luks barrier

$$\varphi : G \twoheadrightarrow \mathrm{Alt}(\Gamma)$$





Γ: **ideal domain**

Plato

Automorphism action on ideal domain

Barrier situation: $G \overset{\varphi}{\twoheadrightarrow} \mathrm{Alt}(\Gamma)$

Goal: either

(i) confirm: $\mathrm{Aut}_G(\mathbf{x}) \overset{\varphi}{\twoheadrightarrow} \mathrm{Alt}(\Gamma)$, or

(ii) break symmetry of Γ:

Automorphism action on ideal domain

Barrier situation: $G \overset{\varphi}{\twoheadrightarrow} \mathrm{Alt}(\Gamma)$

Goal: either

(i) confirm: $\mathrm{Aut}_G(\mathbf{x}) \overset{\varphi}{\twoheadrightarrow} \mathrm{Alt}(\Gamma)$, or

(ii) break symmetry of Γ:

find $M \leq \mathrm{Sym}(\Gamma)$

M much smaller than $\mathrm{Sym}(\Gamma)$

s.t. $\varphi(\mathrm{Aut}_G(\mathbf{x})) \leq M$ (**"encasing group"**)

reduce G to $\varphi^{-1}(M)$, **recurse**

Automorphism action on ideal domain

Def: Giant(Γ) $=$ Sym(Γ) or Alt(Γ)

Recall **Goal:** (i) confirm $\mathrm{Aut}_G(\mathbf{x}) \stackrel{\varphi}{\twoheadrightarrow} \mathrm{Alt}(\Gamma)$

or at least $\mathrm{Aut}_G(\mathbf{x}) \stackrel{\varphi'}{\twoheadrightarrow} \mathrm{Giant}(\Gamma')$
for some $\Gamma' \subset \Gamma$, $|\Gamma'| \geq 0.9|\Gamma|$

if "yes," all of $\mathrm{ISO}_G(\mathbf{x}, \mathbf{y})$ is found

by efficient Luks reduction 

---

idea: condition verifiable: lift 3-cycles on Γ to $\mathrm{Aut}_G(\mathbf{x})$
(efficient Luks reduction)
once verified, $\approx$ every bijection
$\mathrm{supp}(\varphi(F_{\mathbf{x}})) \to \mathrm{supp}(\varphi(F_{\mathbf{y}}))$
lifts to $\mathbf{x} \to \mathbf{y}$ isomorphism
(again, efficient Luks reduction)

Automorphism action on ideal domain

else **Goal** (ii): break symmetry of Γ

find "encasing group" M

$\mathrm{Aut}_G(\mathbf{x}) \xrightarrow{\varphi} M << \mathrm{Sym}(\Gamma)$

Automorphism action on ideal domain

else **Goal** (ii): break symmetry of Γ

find "encasing group" M

$$\mathrm{Aut}_G(\mathbf{x}) \xrightarrow{\varphi} M << \mathrm{Sym}(\Gamma)$$

HOW ?

canonical coloring of Γ:

preserved under G-isomorphisms

---

canonical partition of Γ defined analogously

and any other canonical structure on Γ

CANONICITY of assignment of objects: a **functor**

from the category of G-isomorphisms

Canonical assignment

Assignment $\mathbf{x} \mapsto F(\mathbf{x})$ structures

E.g., $\mathbf{x}$ – graph, $F(\mathbf{x})$ – coloring of vertices

F **canonical** if it also assigns
isomorphism $\mapsto$ isomorphism

$$\begin{array}{ccc} \mathbf{x} & \xrightarrow{\sigma} & \mathbf{y} \\ \Downarrow & & \Downarrow \\ F(\mathbf{x}) & & F(\mathbf{y}) \end{array}$$

Canonical assignment

Assignment $\quad \mathbf{x} \mapsto F(\mathbf{x})$ structures

E.g., $\quad \mathbf{x}$ – graph, $\quad F(\mathbf{x})$ – coloring of vertices

F **canonical** if it also assigns
isomorphism $\mapsto$ isomorphism

$$\begin{tikzcd} \mathbf{x} \arrow[r, "\sigma"] \arrow[d, Rightarrow] & \mathbf{y} \arrow[d, Rightarrow] \\ F(\mathbf{x}) \arrow[r, "F(\sigma)"'] & F(\mathbf{y}) \end{tikzcd}$$

FUNCTOR between categories of isomorphisms

$$F(\sigma\tau) = F(\sigma)F(\tau)$$

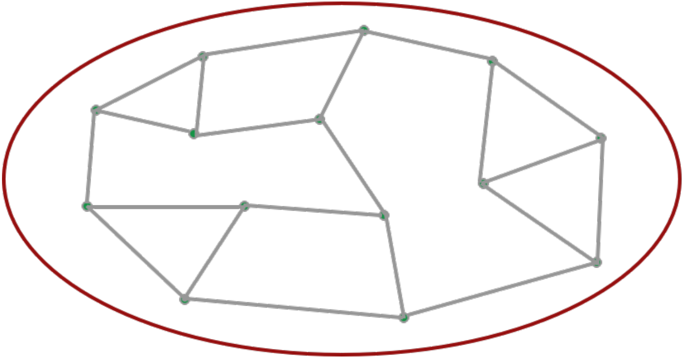
e.g., $\quad F : \text{Graphs} \to \text{ColoredSets}$

Canonical coloring

Given a graph X, vertex-coloring by degree is **canonical**

What if X is *regular?*

Example: color each vertex by number of triangles attached

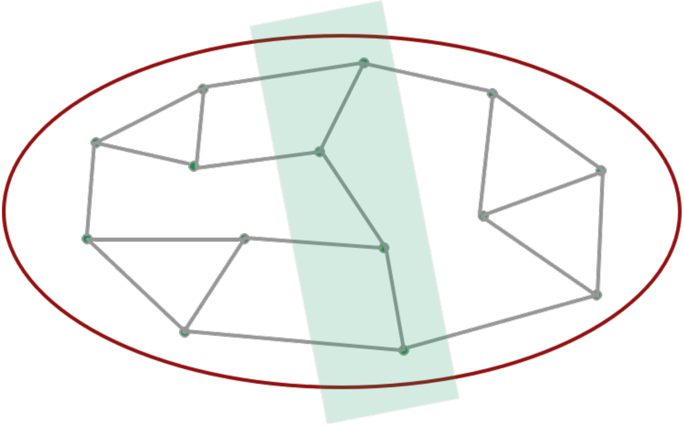


Canonical coloring

Given a graph X, vertex-coloring by degree is **canonical**

What if X is *regular?*

Example: color each vertex by number of triangles attached

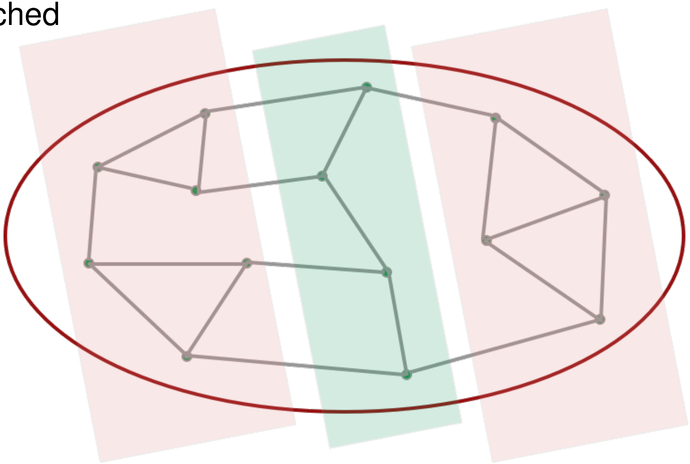


Canonical coloring

Given a graph X, vertex-coloring by degree is **canonical**

What if X is *regular?*

Example: color each vertex by number of triangles attached



Updated recurrence

Break symmetry? How?

- find **good canonical coloring**: every color class ≤ 0.9 fraction of Γ,   or
- find canonical coloring with nontrivial **canonical equipartition** of dominant color class

Complexity: $g(n, m)$ $n = |\Omega|$ $m = |\Gamma| \le n$

$$f(n) = g(n, n)$$
$$g(n, m) \le q(n) g(n, 0.9m)$$
$$g(n, m_0) \le f(0.9n)$$

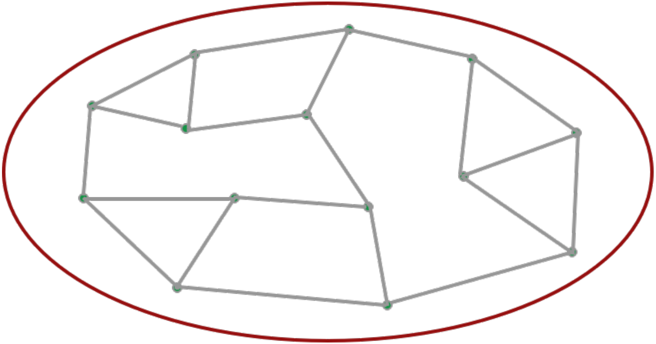
m_0 cutoff point (polylog(n))

Solution: $f(n) = q(n)^{O(\log^2 n)}$

Breaking symmetry on ideal domain

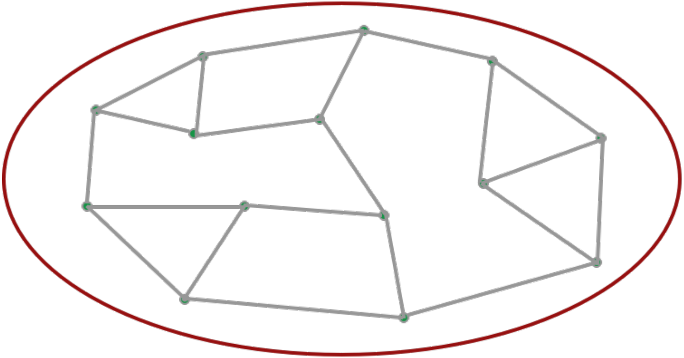
Goal break symmetry

Intermediate goal Find canonically embedded nontrivial regular graph on ≥ 0.9 fraction of Γ



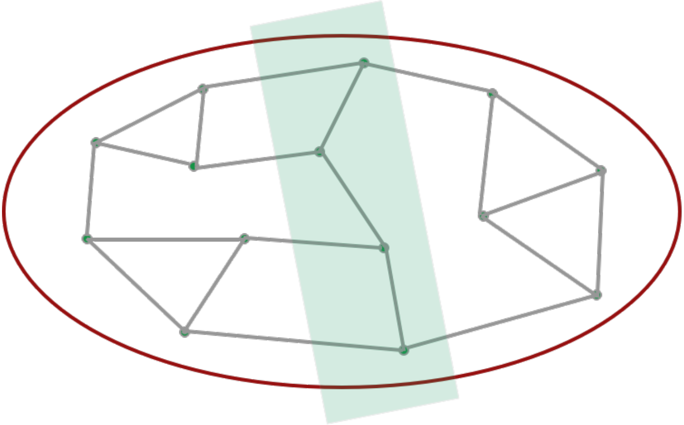
Breaking symmetry of regular graph?

Given a nontrivial regular graph X on Γ
can we find an X-canonical good coloring or equipartition at modest multiplicative cost?



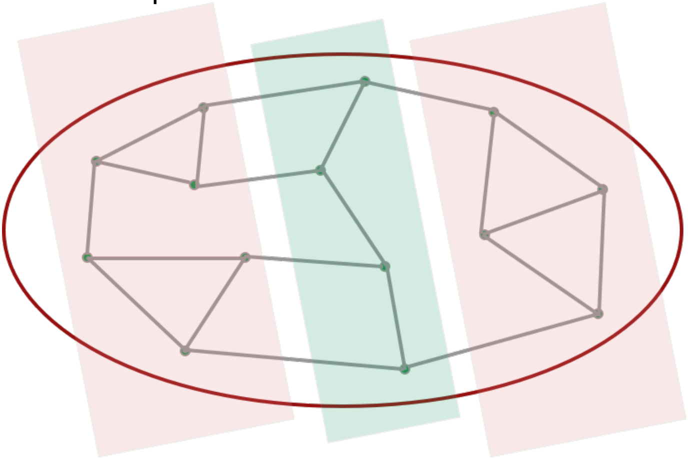
Breaking symmetry of regular graph?

Given a nontrivial regular graph X on Γ
can we find an X-canonical good coloring or equipartition at modest multiplicative cost?



Breaking symmetry of regular graph?

Given a nontrivial regular graph X on Γ
can we find an X-canonical good coloring or equipartition
at modest multiplicative cost?



Breaking symmetry of regular graph?

Given a nontrivial regular graph X on Γ
can we find an X-canonical good coloring or equipartition at modest multiplicative cost?

NO: **Johnson graphs**
resilient to good coloring/partition

DEF: $J(k,t)$ **Johnson graph** $t \geq 1$ $k \geq 2t+1$

vertex set $V = \{v_T \mid T \subseteq \Delta, |T| = t\}$ where $|\Delta| = k$

$$|V| = \binom{k}{t}$$

adjacency: $v_T \sim v_S \iff |T \setminus S| = 1$

---

multiplicative cost of good coloring/partition $\exp(\Omega(k/t))$

Split-or-Johnson

Johnson graphs are the *only* obstructions to good partitioning

Theorem (Split-or-Johnson)

Given a nontrivial regular graph on Γ, one can individualize a polylog number of vertices and find

(a) *a good canonical coloring ($\forall$ color class ≤ 0.9), or*

(b) *a canonical equipartition of the dominant color class (> 0.9), or*

(c) *a canonically embedded Johnson graph on the dominant color class*

Canonicity: relative to the choice of the polylog vertices

Johnson graph: friend or foe?

Johnson graph $J(m', t)$ on Γ

$\Gamma = \binom{\Gamma'}{t} \quad t \geq 2$

$m = \binom{m'}{t}$ so $m' < 1 + \sqrt{2m}$

Reduce $\varphi(G)$ from $\mathrm{Sym}(\Gamma)$ to $\mathrm{Aut}(J(m', t)) \cong S_{m'}$

m dramatically reduced
recurrence bottoms out in $\log \log n$ rounds

We don't immediately find a canonical regular graph

First, a **canonical k-ary relation** for $k = O(\log n)$

$\mathfrak{X} = (\Gamma, R)$ where $R \subseteq \Gamma^k$ $\quad$ k-ary relation

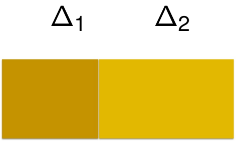
Twins, symmetry defect

$\mathfrak{X} = (\Gamma, \mathcal{R})$ — structure

DEF: $x \neq y \in \Gamma$ **twins** if transposition $(x, y) \in \mathrm{Aut}(\mathfrak{X})$
Fact: "twin or equal" — equivalence relation

DEF: $\Delta \subseteq \Gamma$ **set of twins**: subset of equivalence class
Fact: $\Delta \subseteq \Gamma$ **set of twins** $\iff$ $\mathrm{Sym}(\Delta) \leq \mathrm{Aut}(\mathfrak{X})$
DEF: Symmetricity of $\mathfrak{X}$:
relative size of largest twin equivalence class
DEF: Symmetry defect of $\mathfrak{X}$: $1-$ symmetricity$(\mathfrak{X})$

Example: if $\mathrm{Aut}(\mathfrak{X}) = \mathrm{Sym}(\Delta_1) \times \mathrm{Sym}(\Delta_2)$ where
$\Gamma = \Delta_1 \cup \Delta_2$ then the defect of $\mathfrak{X}$ is $\min\{|\Delta_1|, |\Delta_2|\}/|\Gamma|$



Design Lemma: k-ary to binary

Theorem (Design Lemma)

Given a k-ary relation on Γ with symmetry defect $\geq 1/10$, one can individualize $k-1$ vertices and find

(a) *a good canonical coloring ($\forall$ color class ≤ 0.9), or*

(b) *a canonical equipartition of the dominant color class (> 0.9), or*

(c) *a canonically embedded nontrivial regular graph on the dominant color class*

– in time $m^{O(k)}$.

Canonicity: relative to the choice of the $k-1$ vertices

Design Lemma: k-ary to binary

Theorem (Design Lemma)

Given a k-ary relation on Γ with symmetry defect $\geq 1/10$, one can individualize $k-1$ vertices and find

(a) *a good canonical coloring ($\forall$ color class ≤ 0.9), or*
(b) *a canonical equipartition of the dominant color class (> 0.9), or*
(c) *a canonically embedded nontrivial regular graph on the dominant color class*

– in time $m^{O(k)}$.

Canonicity: relative to the choice of the $k-1$ vertices

How do we obtain canonical k-ary relation?

Design Lemma: k-ary to binary

Theorem (Design Lemma)

Given a k-ary relation on Γ with symmetry defect $\geq 1/10$, one can individualize $k-1$ vertices and find

(a) *a good canonical coloring ($\forall$ color class ≤ 0.9), or*

(b) *a canonical equipartition of the dominant color class (> 0.9), or*

(c) *a canonically embedded nontrivial regular graph on the dominant color class*

– in time $m^{O(k)}$.

Canonicity: relative to the choice of the $k-1$ vertices

How do we obtain canonical k-ary relation?

GROUP THEORY $\qquad k = O(\log n)$

Overall plan

- Luks works until barrier encountered
- Luks $+$ Cameron construct ideal domain Γ
  giant homomorphism $G \overset{\varphi}{\twoheadrightarrow} \mathrm{Alt}(\Gamma)$
- construct "Local certificates" $\heartsuit$
- aggregation of Local certificates: constructs
  canonical k-ary relation on Γ $\quad k = O(\log n)$
  with large symmetry defect
- Design Lemma reduces k-ary to binary
  $\rightarrow$ regular graph $\qquad$ method: k-ary WL
- Split-or-Johnson significantly reduces $|\Gamma|$
  method: (classical) coherent configurations ($k = 2$)
- return "divided" domain to Luks to "conquer"

Role of “Unaffected Stabilizers Lemma”

To construct **“local certificates”** on Γ
from which **canonical k-ary relation** is derived

Key difficulty:

“Global automorphisms

from local information”

Unaffected stabilizers lemma

Let $G \leq \mathrm{Sym}(\Omega)$ and $\varphi : G \twoheadrightarrow \mathrm{Giant}(\Gamma)$.
G_x: stabilizer of $x \in \Omega$ in G (subgroup that fixes x).

DEF: $x \in \Omega$ is **affected** by φ if
$\varphi(G_x)$ is NOT a giant in $\mathrm{Sym}(\Gamma)$

Theorem (Unaffected stabilizers lemma)

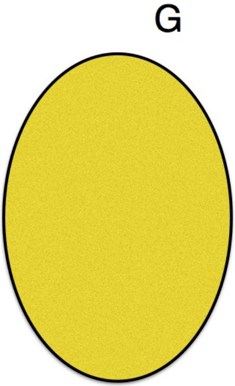
Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

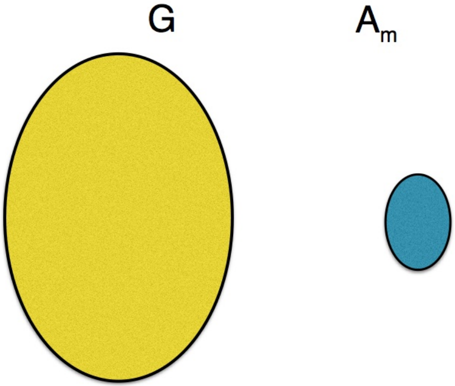
If $m > 2 + \log_2 n$ then

$$\varphi(G_{(U)}) \text{ is a giant in } \mathrm{Sym}(\Gamma).$$

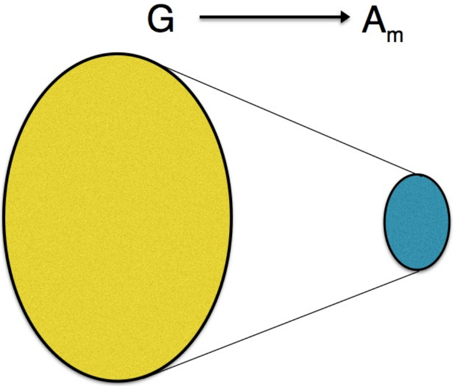
Unaffected stabilizers lemma



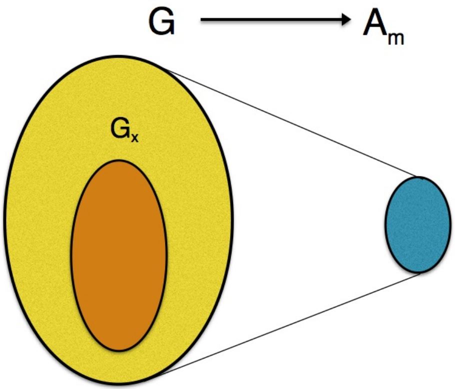
Unaffected stabilizers lemma



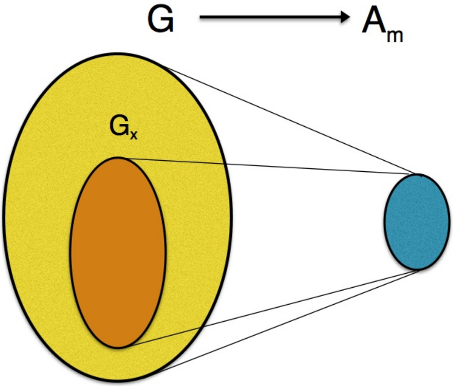
Unaffected stabilizers lemma



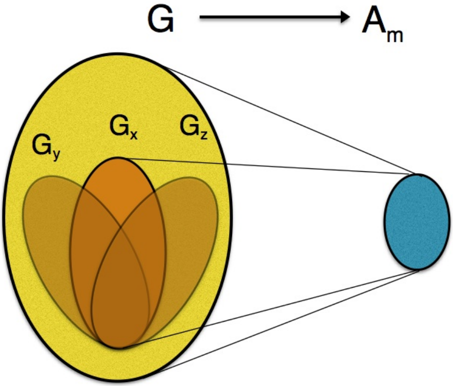
Unaffected stabilizers lemma



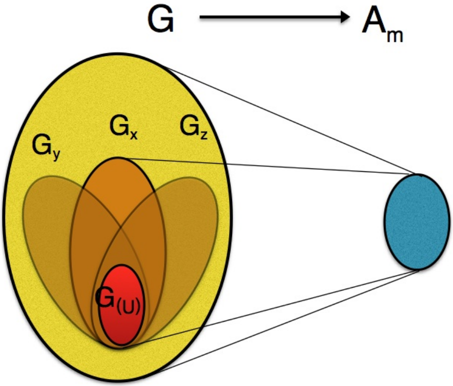
Unaffected stabilizers lemma



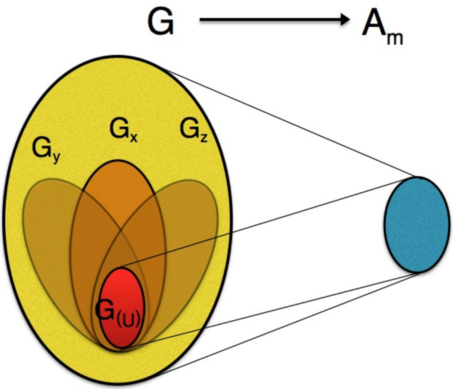
Unaffected stabilizers lemma



Unaffected stabilizers lemma



Unaffected stabilizers lemma



Unaffected stabilizers lemma

Let $G \leq \mathrm{Sym}(\Omega)$ and $\varphi : G \twoheadrightarrow \mathrm{Giant}(\Gamma)$.
G_x: stabilizer of $x \in \Omega$ in G (subgroup that fixes x).

DEF: $x \in \Omega$ is **affected** by φ if
$\varphi(G_x)$ is NOT a giant in $\mathrm{Sym}(\Gamma)$.

Theorem (Unaffected stabilizers lemma)

Let U be the set of unaffected elements of Ω and $G_{(U)}$ the pointwise stabilizer of U, i.e.,

$$G_{(U)} = \bigcap_{x \in U} G_x$$

If $m > 2 + \log_2 n$ then $\varphi(G_{(U)})$ is a giant in $\mathrm{Sym}(\Gamma)$.

Affected orbit lemma

Let $K \to G \overset{\varphi}{\twoheadrightarrow} \mathrm{Giant}(\Gamma)$. Let Δ be an affected orbit.

Proposition (Affected orbit lemma)

If $m \geq 5$ then K is not transitive on Δ; each K-orbit on Δ has length $\leq |\Delta|/m$.

This will allow efficient Luks-recurrence

Local certificates

$|\Omega| = n \quad |\Gamma| = m$
$G \leq \mathrm{Sym}(\Omega) \quad \varphi : G \twoheadrightarrow \mathrm{Giant}(\Gamma)$

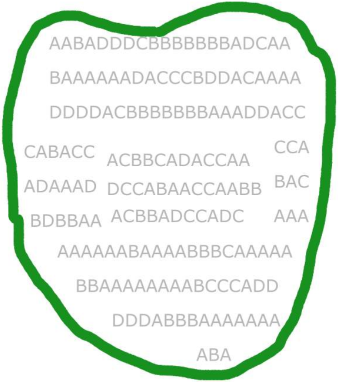
Test set: $T \subset \Gamma \quad |T| = t \quad t > 2 + \log_2 n$

G_T setwise stabilizer of T in G

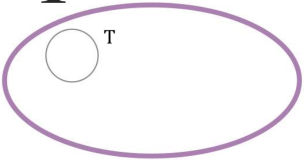
ψ_T composition of $G_T \xrightarrow{\varphi} \mathrm{Sym}(\Gamma)_T \to \mathrm{Sym}(T)$

Local certificates

Ω

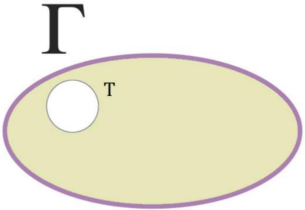


Γ



Selecting a test set $T \subset \Gamma$

Local certificates



Restricting G to G_T and φ to $\psi_T : G_T \to \mathrm{Sym}(T)$

Local certificates

T test set: $T \subset \Gamma \quad |T| = t > 2 + \log_2 n$

DEF: ***"T is full"*** if $\mathrm{Aut}_{G_T}(\mathbf{x}) \overset{\psi_T}{\twoheadrightarrow} \mathrm{Giant}(T)$

Fullness certificate: $K(T) \leq \mathrm{Aut}_{G_T}(\mathbf{x})$ such that

$K(T) \overset{\psi_T}{\twoheadrightarrow} \mathrm{Giant}(T)$ $K(T)$ global object

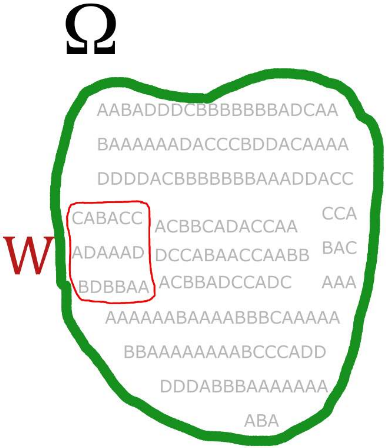
Non-fullness certificate: $M(T) \leq \mathrm{Sym}(T)$, not giant, s.t.

$\psi_T(\mathrm{Aut}_{G_T}(\mathbf{x})) \leq M(T)$ $M(T)$ local object

Theorem

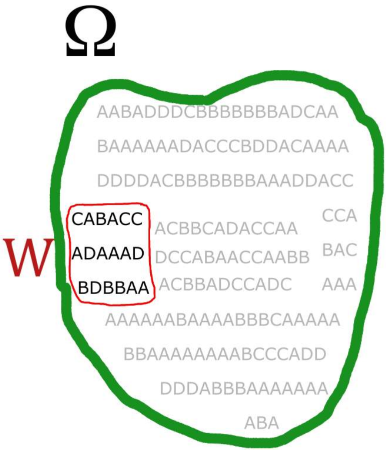
We can decide by efficient recursion whether or not T is full, and find certificates for each outcome.

Windows, partial strings



$W \subseteq \Omega$ window

Windows, partial strings



$\mathbf{x}^W : W \to ABC$ partial string: restriction of $\mathbf{x}$ to window

$A(G, W) := \mathrm{Aut}_G^W(\mathbf{x})$ aut group of partial string $\mathbf{x}^W$
(fixes W setwise)

Local certificates algorithm

W: current window

$A(G_T, W) := \mathrm{Aut}_{G_T}^{W}(\mathbf{x})$ aut group of partial string $\mathbf{x}^W$

Procedure *Local Certificates*

initialize: $W \leftarrow \emptyset$ (: $A(G_T, \emptyset) = G_T$:)

while (condition)

$W \leftarrow \mathrm{Aff}(A(G_T, W))$ points affected by current $A(G_T, W)$

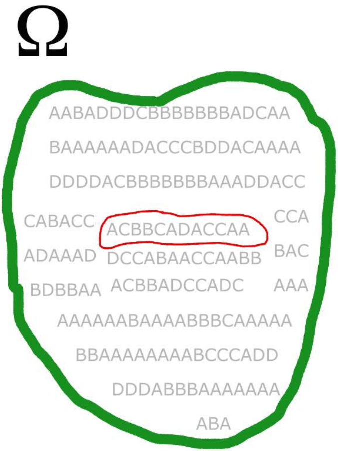
update $A(G_T, W)$

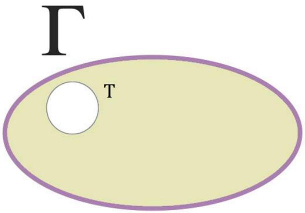
end(while)

produce certificate

Note: $H \leq G \implies \mathrm{Aff}(H) \supseteq \mathrm{Aff}(G)$ so the window W keeps growing

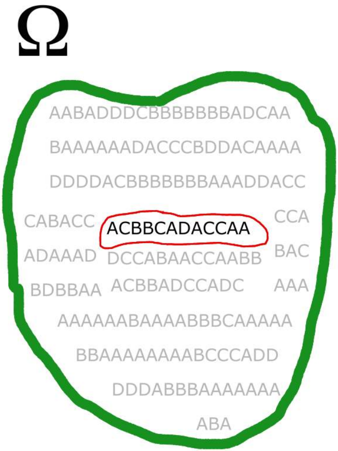
Local certificates algorithm

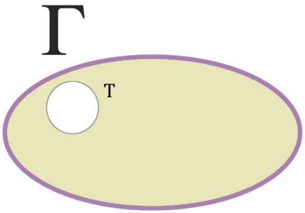




W: points affected by G_T

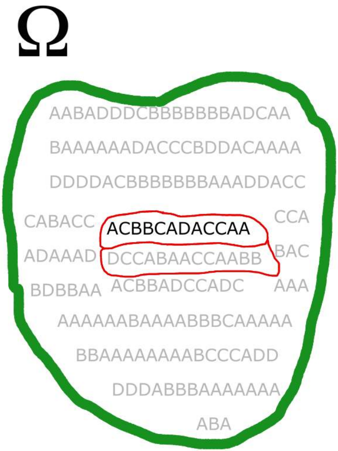
Local certificates algorithm

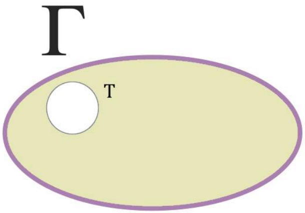




partial string $\mathbf{x}^W$ uncovered, its aut group $A(G_T, W)$ updated

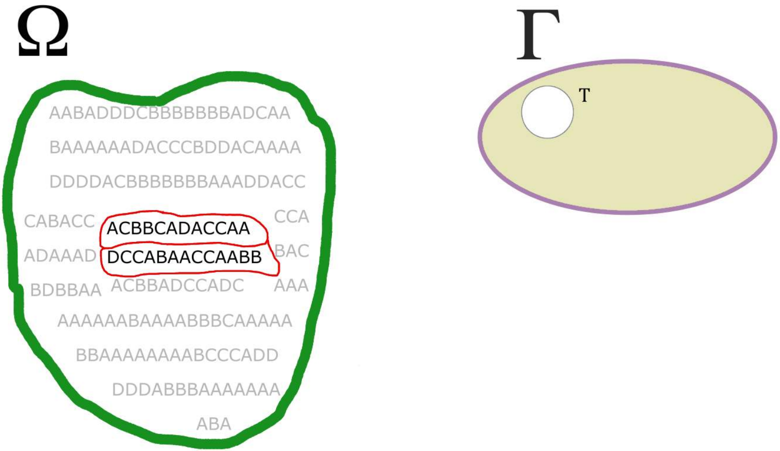
Local certificates algorithm





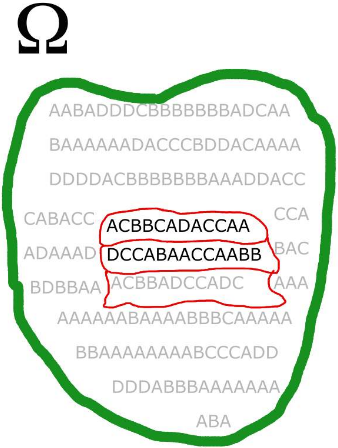
W updated: new layer added to the window

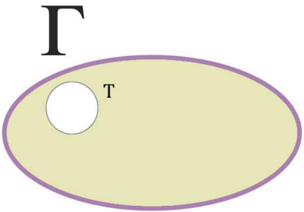
Local certificates algorithm



partial string $\mathbf{x}^W$ and its aut group $A(G_T, W)$ updated

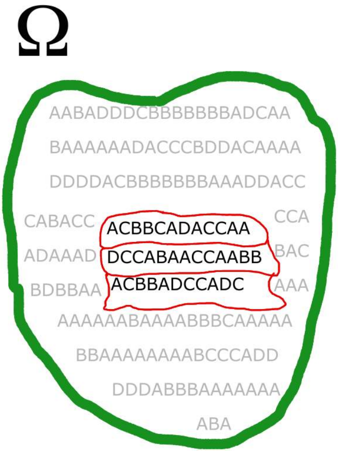
Local certificates algorithm

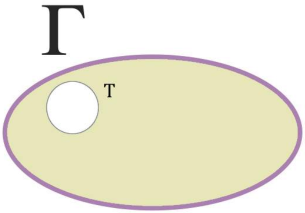




W updated; another layer added to the window

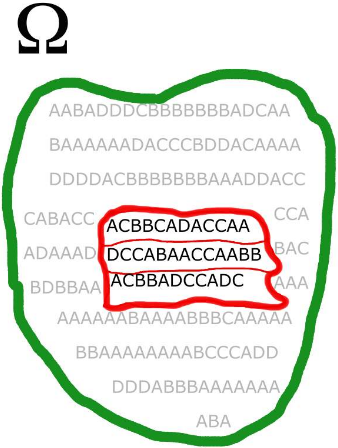
Local certificates algorithm

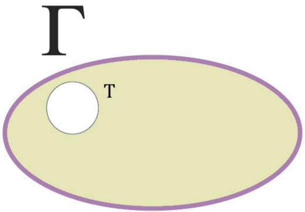




partial string $\mathbf{x}^W$ and its aut group $A(G_T, W)$ updated

Local certificates





while loop ended

Local certificates algorithm

How does the **while** loop end?

(A) $A(G_T, W)$ became too small, it does not map onto Giant(T), or

(B) window stopped growing

Which case corresponds to what type of certificate?

Local certificates algorithm

How does the **while** loop end?

(A) $A(G_T, W)$ became too small, it does not map onto Giant(T), or

(B) window stopped growing

Which case corresponds to what type of certificate?

(A): $M(T) := \psi_T(A(G_T, W))$ non-fullness

(B): need $K(T) \le \mathrm{Aut}_{G_T}(\mathbf{x})$ that maps onto Giant(T)
$A(G_T, W)$ does map onto Giant(T)
but only respects $\mathbf{x}^W$

Local certificates algorithm

How does the **while** loop end?

(A) $A(G_T, W)$ became too small, it does not map onto $\mathrm{Giant}(T)$, or

(B) window stopped growing

Which case corresponds to what type of certificate?

(A): $M(T) := \psi_T(A(G_T, W))$ non-fullness

(B): need $K(T) \le \mathrm{Aut}_{G_T}(\mathbf{x})$ that maps onto $\mathrm{Giant}(T)$
$A(G_T, W)$ does map onto $\mathrm{Giant}(T)$
but only respects $\mathbf{x}^W$

$K(T) := (A(G_T, W))_{(U)}$ where $U = \Omega \setminus W$

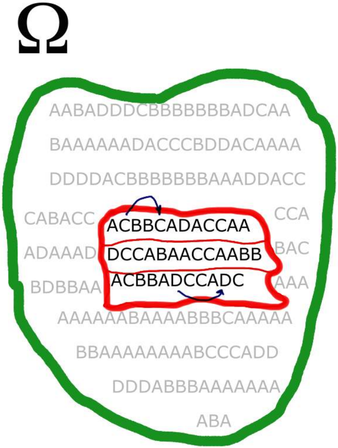
Local certificates

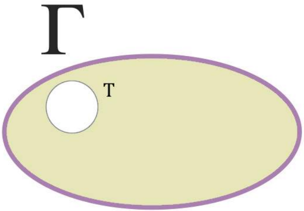
$K(T) := (A(G_T, W))_{(U)}$ where $U = \Omega \setminus W$

maps onto Giant(T) by “Unaffected stabilizers lemma”

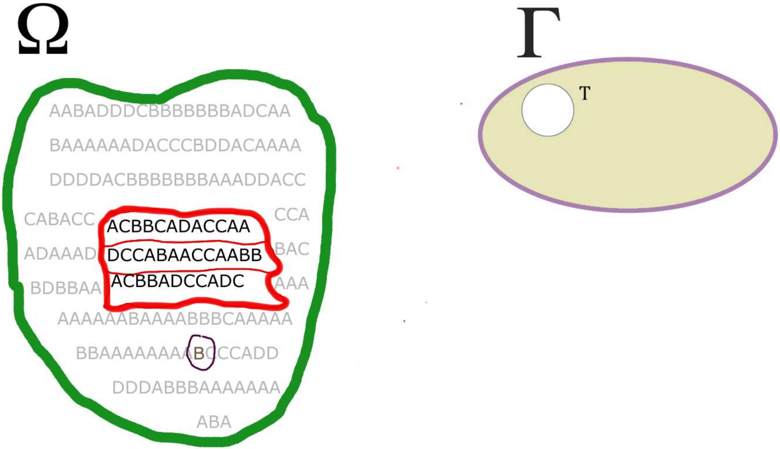
Why does it consist of (global) automorphisms?

Local certificates



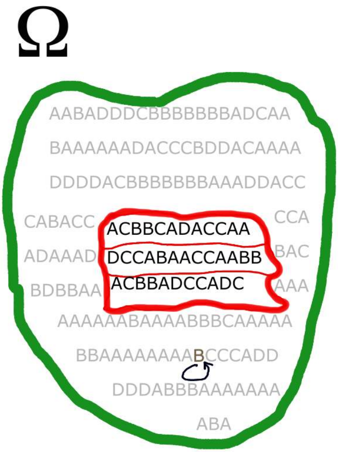


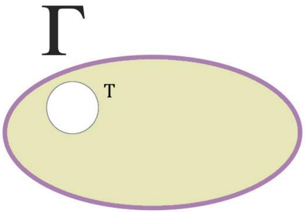
Local certificates



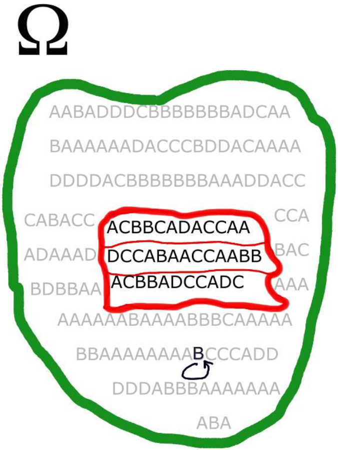
Why is this letter B respected?

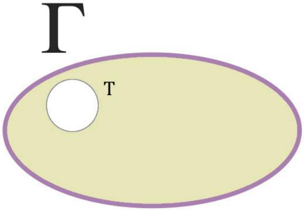
Local certificates





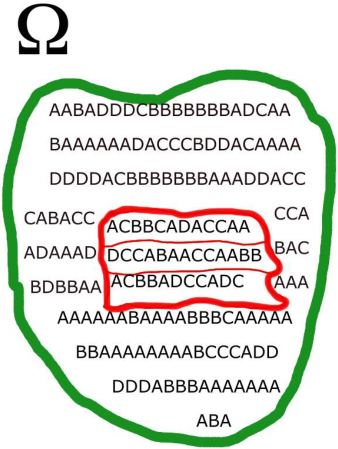
Why is this letter *B* respected?
Because it is fixed

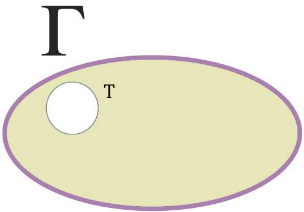




Why is this letter *B* respected?
Because it is fixed

Local certificates





So all letters are respected:

global automorphisms from local information

Aggregation of certificates: sketch

- If fullness certificates dominate:
  rich set of (global!) G-automorphisms found

  $\to$ use group theory to **split Γ**

  (orbits, bounds on multiple transitivity)

  or reduce to Johnson group

- If non-fullness certificates dominate: large set of (local!) obstacles to equivalence of ordered t-tuples found

  $\to$ they define a **canonical t-ary relation on Γ**

  $t =$ size of test sets $\approx \log_2 n$

Aggregation of certificates: details

$F := \langle K(T) \mid T \text{ full } \rangle \leq \mathrm{Aut}_G(\mathbf{x})$

(group generated by fullness certificates)

$s := |\mathrm{supp}(\varphi(F))|$ number of points in Γ not fixed by F

Case A: $m/10 \leq s \leq 9m/10 \quad \rightarrow$

$(\mathrm{supp}, \Gamma \setminus \mathrm{supp})$ good coloring

(end Case A)

Case B: $s > 9m/10$ group acts on 90% of Γ

if no $\varphi(F)$-orbit dominant ($> 9m/10$) $\rightarrow$ good partition

else $\Gamma \leftarrow$ dominant orbit (efficient Luks-reduction)

Aggregation of certificates: details

$F := \langle K(T) \mid T \text{ full } \rangle \leq \mathrm{Aut}_G(\mathbf{x})$

(group generated by fullness certificates)

Case B continued: $\varphi(F)$ transitive on Γ

if $\varphi(F)$-action on Γ giant

easy case, already dealt with:

ISO$(\mathbf{x}, \mathbf{y})$ via efficient Luks reduction

Aggregation of certificates: details

$F := \langle K(T) \mid T \text{ full} \rangle \leq \mathrm{Aut}_G(\mathbf{x})$
(group generated by fullness certificates)

Case B continued: $\varphi(F)$ transitive but not giant on Γ

$t :=$ degree of transitivity of $\varphi(F)$: $t \geq 1$
(: $t \leq 5$ (CFSG) or $t < \log^2 n$ (Bochert 1896) :)
individualize $t - 1$ points
(: $\varphi(F)_{(T)}$ transitive but not doubly transitive
on $\Gamma \setminus T$ where $|T| = t - 1$:)
individualize one of the orbitals (orbits on pairs)
(: multipl cost $=$ # orbitals $\leq n - 1$:)
$\therefore$ **canonical biregular digraph** found
$\rightarrow$ Split-or-Johnson
(end Case B)

Aggregation of certificates: details

$F := \langle K(T) \mid T \text{ full} \rangle \leq \mathrm{Aut}_G(\mathbf{x})$

(group generated by fullness certificates)

Case C: $|\mathrm{supp}(\varphi(F))| < m/10$:

(: 90% of Γ has non-fullness certificates only :)

infer **canonical *t*-ary relation**
with **large symmetry defect**

$\to$ Design Lemma $\to$ Split-or-Johnson

Aggregation of non-fullness certificates

$\Gamma' := \Gamma \setminus \mathrm{supp}(\varphi(F))$

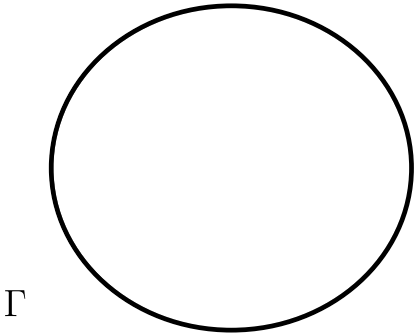
$$|\Gamma'| \geq 0.9|\Gamma|$$

GOAL:

canonical t-ary relational structure on Γ'

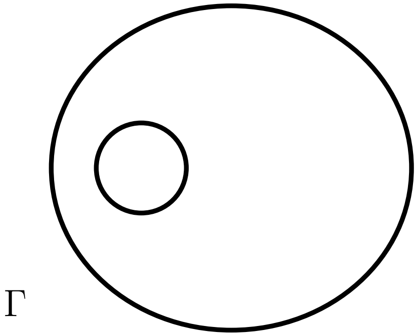
with large symmetry defect

Aggregation of non-fullness certificates



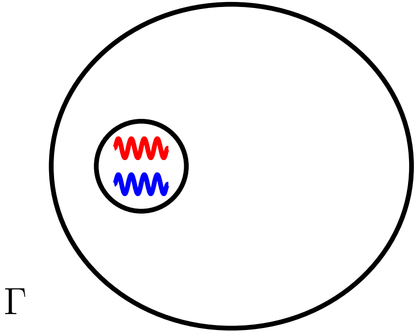
Ideal domain Γ

Aggregation of non-fullness certificates



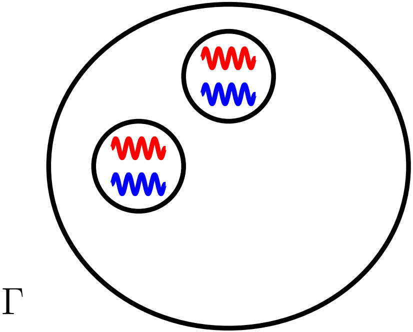
Test set

Aggregation of non-fullness certificates



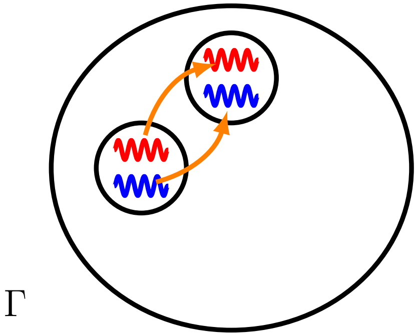
Inequivalent orderings of test set

Aggregation of non-fullness certificates



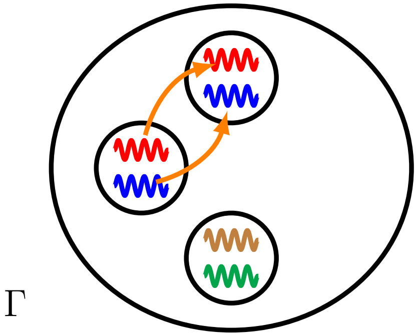
Another test set

Aggregation of non-fullness certificates



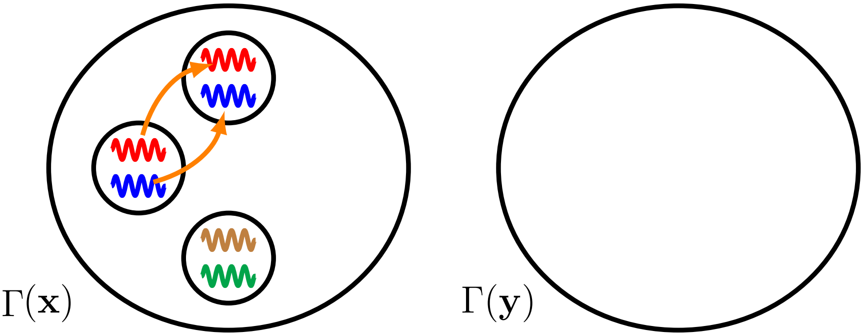
Equivalent orderings of test sets

Aggregation of non-fullness certificates



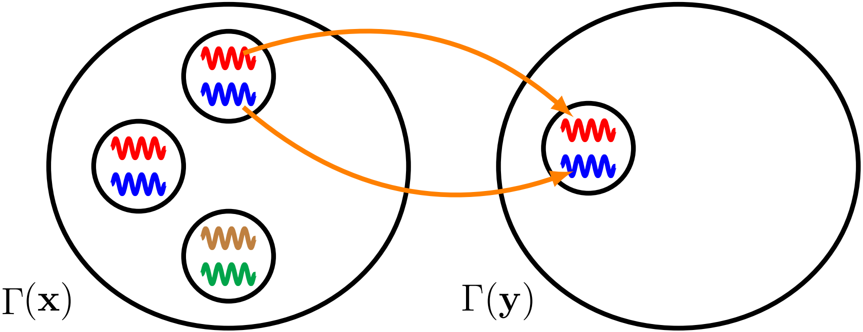
Third test set: not equivalent with first two

Aggregation of non-fullness certificates



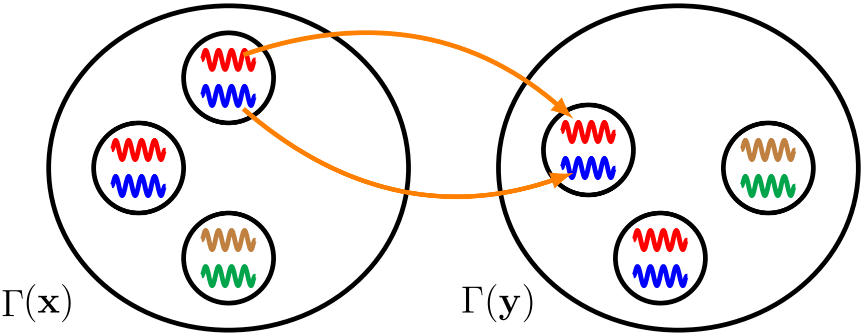
Ideal domains for two input strings $\mathbf{x}, \mathbf{y}$

Aggregation of non-fullness certificates



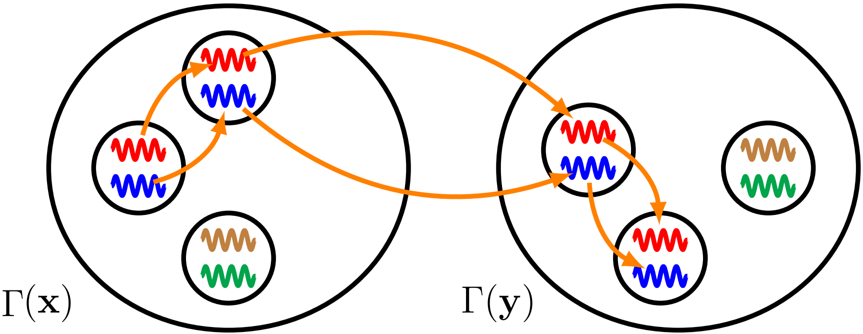
Identification of ordered t-tuples across inputs

Aggregation of non-fullness certificates



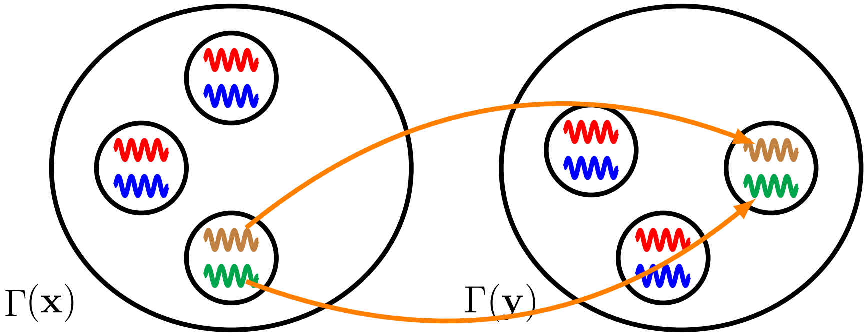
More test sets

Aggregation of non-fullness certificates



Equivalent ordered t-tuples

Aggregation of non-fullness certificates



Equivalent ordered t-tuples

Aggregation of non-fullness certificates

OBTAINED:

canonical t-ary relational structure on Γ'

with large symmetry defect:

symmetricity $<$ relative size of test set
because $\mathrm{Sym}(T)$ cannot act on
non-full test set T

Overall algorithm

apply Luks's group theoretic divide-and-conquer

when Luks barrier encountered:

find **local certificates** using
affected/unaffected dichotomy

aggregate local certificates

split Γ by group theory or by combinatorial partitioning
or reduce $\mathrm{Sym}(\Gamma)$ to $\mathrm{Sym}(\Gamma') = \mathrm{Aut}(\mathrm{Johnson})$

recurse (Γ significantly reduced)

Paradoxes of Graph Isomorphism

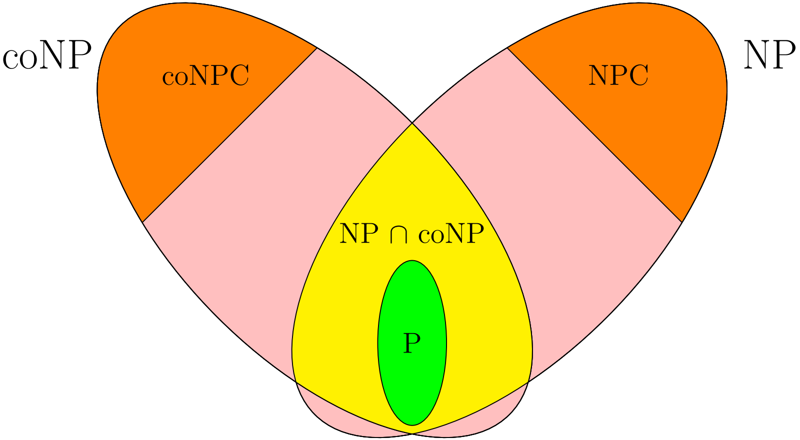
	GraphIso	**factoring**
in practice	easy	hard
hard instances	?	abound
average case	easy	presumed hard
worst case	quasipoly	moderately exp

Paradoxes of Graph Isomorphism

	GraphIso	**factoring**
in practice	easy	hard
hard instances	?	abound
average case	easy	presumed hard
worst case	quasipoly	moderately exp
in coNP?	?	yes
quantum	?	BQP (q-poly-time)

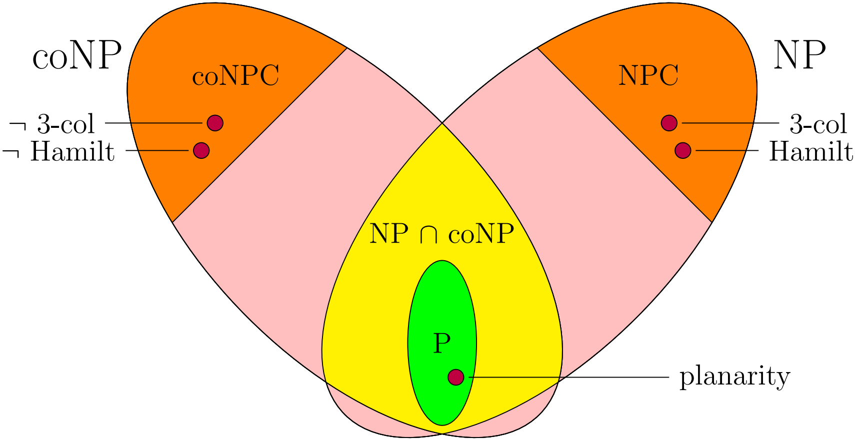
no quantum advantage

Paradoxes of Graph Isomorphism



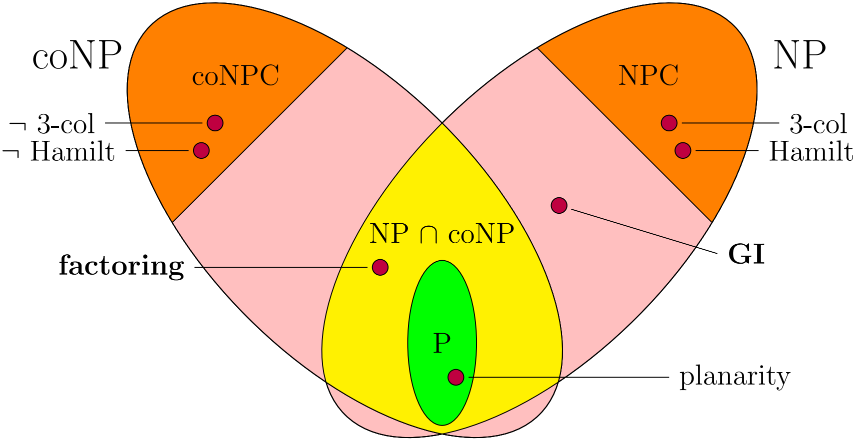
Help with most pictures: *Bernard Lidicky*

Paradoxes of Graph Isomorphism



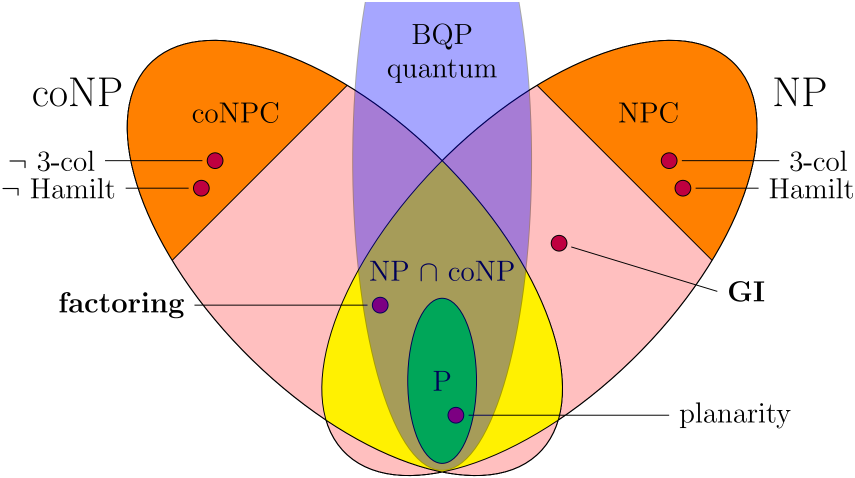
Help with most pictures: *Bernard Lidicky*

Paradoxes of Graph Isomorphism



Help with most pictures: *Bernard Lidicky*

Paradoxes of Graph Isomorphism



Help with most pictures: *Bernard Lidicky*

Paradoxes of Graph Isomorphism

	GraphIso	factoring
in practice	easy	hard
hard instances	?	abound
average case	easy	presumed hard
worst case	quasipoly	moderately exp
in coNP?	?	yes
quantum	?	BQP (q-poly-time)

no quantum advantage

Paradoxes of Graph Isomorphism

	GraphIso	**factoring**
in practice	easy	hard
hard instances	?	abound
average case	easy	presumed hard
worst case	quasipoly	moderately exp
in coNP?	?	yes
quantum	?	BQP (q-poly-time)

no quantum advantage

provable hardness	hard for semi-algebraic proof systems	?

Paradoxes of Graph Isomorphism

GI algorithmically easier

structurally harder than **factoring**