2023.3.7 STACS'23 @ Hamburg, Germany

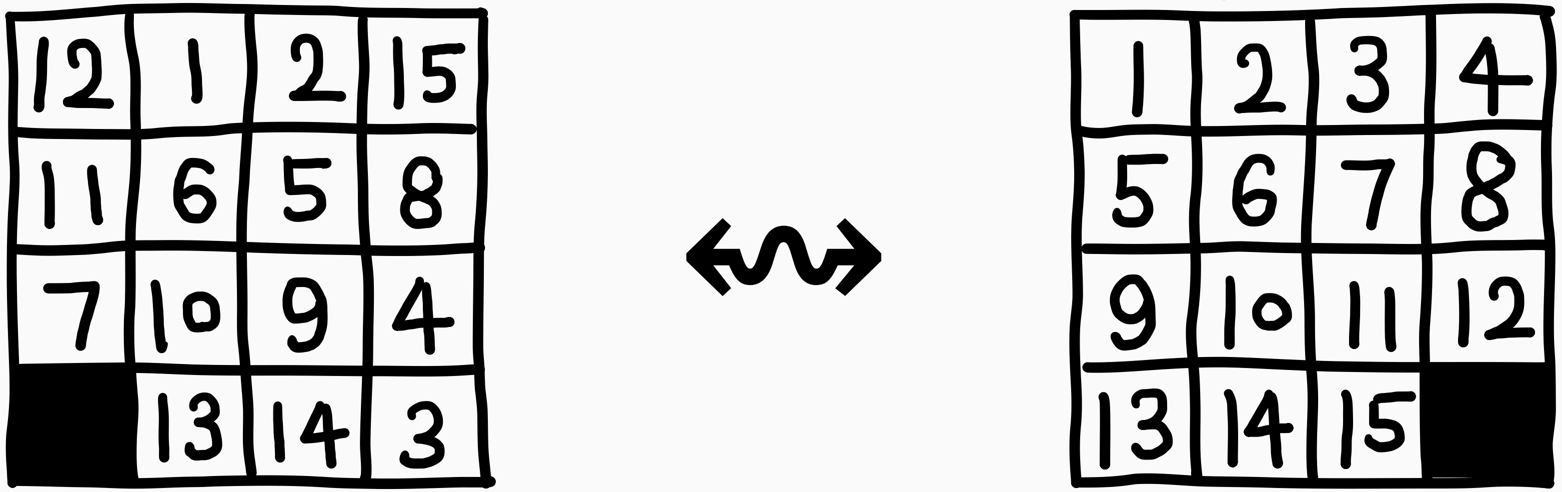
Gap Preserving Reductions Between Reconfiguration Problems

Naoto Ohsaka

CyberAgent AI Lab

What are reconfiguration problems?

Transform initial state into target state by repeating small changes



- Classical puzzles: 15-puzzles, Rubik's cube, sliding block puzzles
- Understand the structure of solution space
  applications in dynamic environments
- Unified framework: defined w.r.t. feasibility & adjacency
  [Ito-Demaine-Harvey-Papadimitriou-Sideri-Uehara-Uno. *Theor. Comput. Sci.*'11]
- Excellent surveys [Nishimura. *Algorithms*'18] [van den Heuvel. '13]

Example

3-SAT Reconfiguration

- **Input:** 3-CNF formula φ & satisfying σ_s, σ_t
- **Output:** $\boldsymbol{\sigma} = \langle \sigma^{(0)}=\sigma_s, \ldots, \sigma^{(\ell)}=\sigma_t \rangle$ s.t.

  $\sigma^{(i)}$ satisfies φ (feasibility)

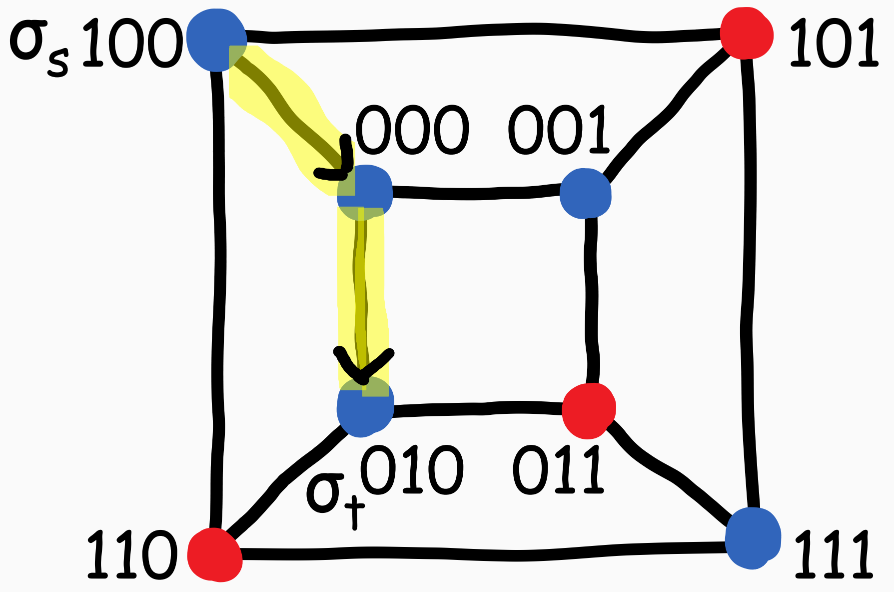
  $\mathrm{Ham}(\sigma^{(i-1)}, \sigma^{(i)}) = 1$ (adjacency on hypercube)

YES case

$\varphi = (\bar{x} \vee \bar{y} \vee z) \wedge (\bar{x} \vee y \vee \bar{z}) \wedge (x \vee \bar{y} \vee \bar{z})$

$\sigma_s = (1,0,0)$

$\sigma_t = (0,1,0)$



Example

3-SAT Reconfiguration

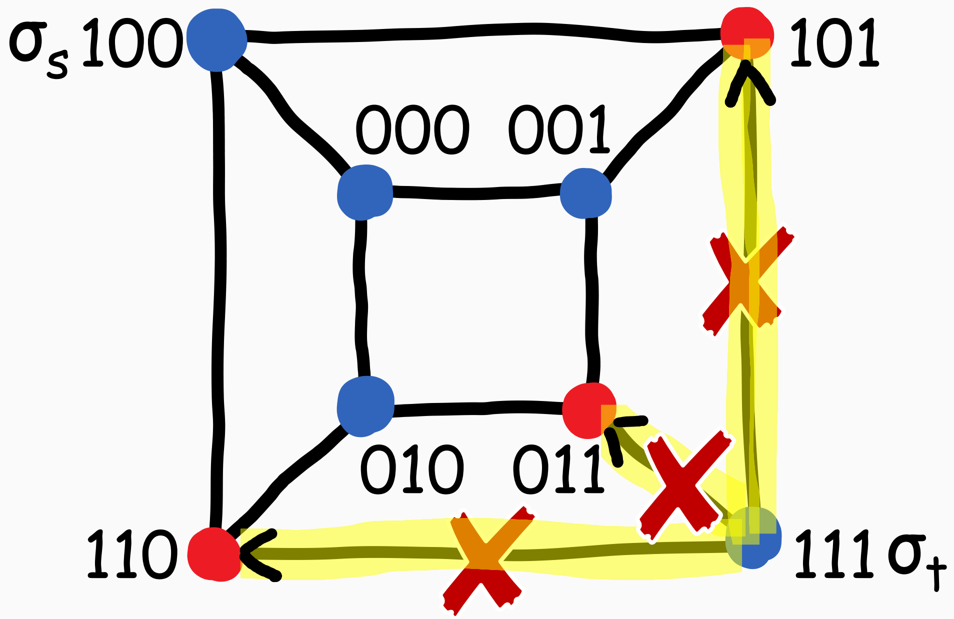
- **Input:** 3-CNF formula φ & satisfying σ_s, σ_t
- **Output:** $\boldsymbol{\sigma} = \langle \sigma^{(0)}=\sigma_s, \ldots, \sigma^{(\ell)}=\sigma_t \rangle$ s.t.
  $\sigma^{(i)}$ satisfies φ (feasibility)
  $\mathrm{Ham}(\sigma^{(i-1)}, \sigma^{(i)}) = 1$ (adjacency on hypercube)

NO case

$\varphi = (\bar{x} \vee \bar{y} \vee z) \wedge (\bar{x} \vee y \vee \bar{z}) \wedge (x \vee \bar{y} \vee \bar{z})$

$\sigma_s = (1,0,0)$

$\sigma_t = (1,1,1)$



Our focus & motivation: Approximate reconfigurability

Even if…

- 😖NOT reconfigurable! and/or
- 😖many problems are **PSPACE**-complete!

Still want a "reasonable" sequence (quickly)

(e.g.) made up of almost-satisfying assignments

⬇

Relax feasibility to obtain *optimization variants*

Example'

Maxmin 3-SAT Reconfiguration

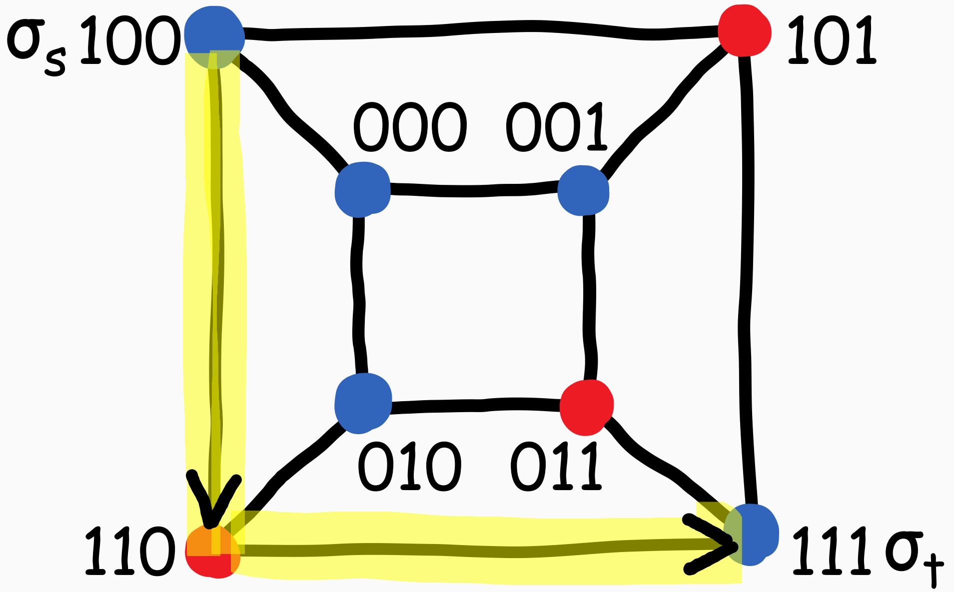
[Ito-Demaine-Harvey-Papadimitriou-Sideri-Uehara-Uno. *Theor. Comput. Sci.*'11]

- **Input:** 3-CNF formula φ & satisfying σ_s, σ_t
- **Output:** $\boldsymbol{\sigma} = \langle \sigma^{(0)}=\sigma_s, \ldots, \sigma^{(\ell)}=\sigma_t \rangle$ s.t.
  ~~$\sigma^{(i)}$ satisfies φ~~ (feasibility)
  $\mathrm{Ham}(\sigma^{(i-1)}, \sigma^{(i)}) = 1$ (adjacency on hypercube)
- **Goal:** $\max_{\boldsymbol{\sigma}} \mathrm{val}_\varphi(\boldsymbol{\sigma}) \stackrel{\text{def}}{=} \min_i$ (frac. of satisfied clauses by $\sigma^{(i)}$)

$\varphi = (\bar{x} \vee \bar{y} \vee z) \wedge (\bar{x} \vee y \vee \bar{z}) \wedge (x \vee \bar{y} \vee \bar{z})$

- $\sigma_s = (1,0,0)$
- $\sigma_t = (1,1,1)$

➜ $\mathrm{val}_\varphi(\boldsymbol{\sigma}) = 2/3$



Known results on optimization variants

How computationally hard?

Approximability

[Ito+. *Theor. Comput. Sci.*'11] [Ito-Demaine. *J. Comb. Optim.*'14] [O.-Matsuoka. *WSDM*'22]

- Set Cover Reconf., Subset Sum Reconf., Submodular Reconf.

Are the problems in Section 4 PSPACE-hard to approximate (not just NP-hard)?

NP-hardness of approximation

[Ito-Demaine-Harvey-Papadimitriou-Sideri-Uehara-Uno. *Theor. Comput. Sci.*'11]

- SAT Reconfiguration & Clique Reconfiguration

  ⬆Rely on **NP**-hardness of combinatorial optimization problems…

Significance of showing **PSPACE**-hardness…

- no poly-time algorithm (**P ≠ PSPACE**)
- no poly-length sequence (**NP ≠ PSPACE**)

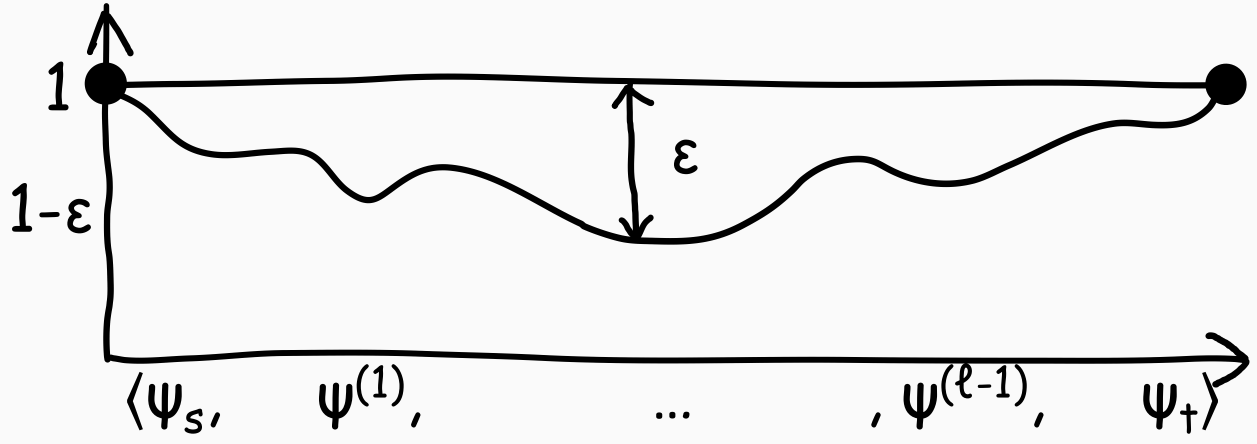
Our working hypothesis & question

Reconfiguration Inapproximability Hypothesis (**RIH**)

q-ary CSP G & satisfying ψ_s, ψ_t, **PSPACE**-hard to distinguish between

- $\exists\psi\ \mathrm{val}_G(\psi) = 1$ (some sequence violates no constraint)
- $\forall\psi\ \mathrm{val}_G(\psi) < 1-\varepsilon$ (any sequence violates >ε-frac. of constraints)

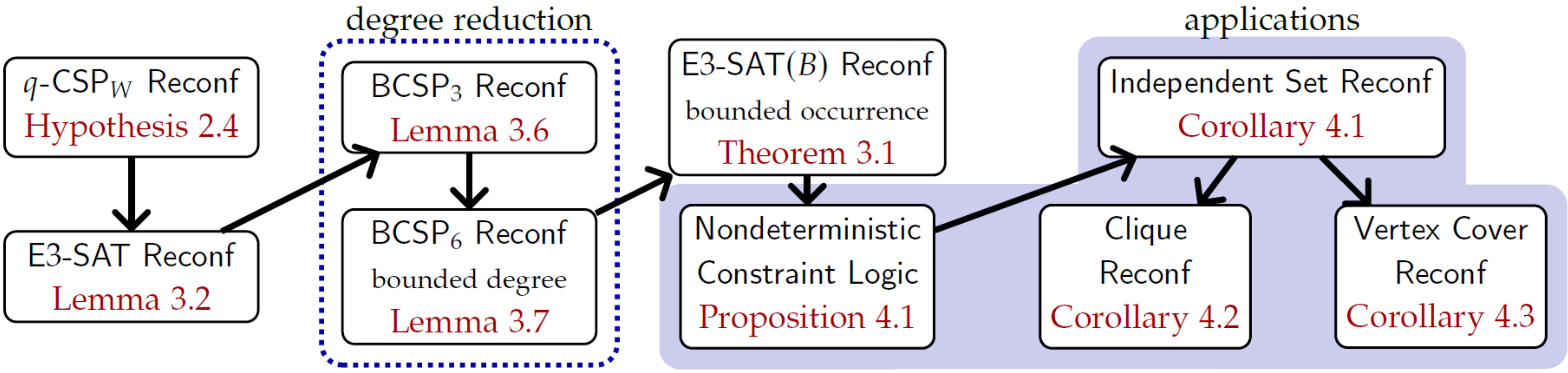
- True if "**NP**-hard" is used [Ito+. *Theor. Comput. Sci.*'11]
- Reconfiguration analogue of the **PCP** theorem (?)



Q. Which reconfiguration problems are **PSPACE**-hard to approximate under (seemingly) plausible **RIH**?

Our results

- Under **RIH**, many problems are **PSPACE**-hard to approximate!!



Sequence of gap-preserving reductions

😋Our reductions preserve perfect completeness

➜ YES instance have a solution for (original) decision version

Main result

Maxmin 3-SAT Reconfiguration of bounded occurrence

- **Input**: 3-CNF formula φ of bounded occurrence & satisfying σ_s, σ_t
- **Output**: $\boldsymbol{\sigma} = \langle \sigma^{(0)}=\sigma_s, \ldots, \sigma^{(\ell)}=\sigma_t \rangle$ s.t.
  $\mathrm{Ham}(\sigma^{(i-1)}, \sigma^{(i)}) = 1$ (adjacency on hypercube)

Define the value of best sequence:

$$\mathrm{OPT}_\varphi(\sigma_s \leftrightsquigarrow \sigma_t) \stackrel{\text{def}}{=} \max_{\boldsymbol{\sigma}} \min_i \text{(frac. of clauses satisfied by } \sigma^{(i)})$$

Under **RIH**, **PSPACE**-hard to distinguish between

- $\mathrm{OPT}_\varphi(\sigma_s \leftrightsquigarrow \sigma_t) = 1$ ($\exists\boldsymbol{\sigma}$ every $\sigma^{(i)}$ satisfies φ), or
- $\mathrm{OPT}_\varphi(\sigma_s \leftrightsquigarrow \sigma_t) < 1\text{-}\varepsilon$ ($\forall\boldsymbol{\sigma}$ some $\sigma^{(i)}$ violates ε-frac. of clauses)

Most technical step in this paper

Degree reduction of reconfiguration problems

- $\text{Gap}_{1,1-\varepsilon}$ Binary CSP Reconfiguration
  Promise: $\text{OPT}_G(\psi_s \leftrightsquigarrow \psi_t) = 1$ or $\text{OPT}_G(\psi_s \leftrightsquigarrow \psi_t) < 1\text{-}\varepsilon$

⬇ ⚠ Reduction preserving gap & reconfigurability

- $\text{Gap}_{1,1-\varepsilon'}$ Binary CSP Reconfiguration of max. degree Δ
  Promise: $\text{OPT}_{G'}(\psi'_s \leftrightsquigarrow \psi'_t) = 1$ or $\text{OPT}_{G'}(\psi'_s \leftrightsquigarrow \psi'_t) < 1\text{-}\varepsilon'$
  😋 Δ & ε' depend only on ε

Why important?

Can reduce to Maxmin 3-SAT Reconfiguration of bounded occurrence

In the remainder of this talk…

Proof sketch of degree reduction

In the remainder of this talk...

Proof sketch of degree reduction

- Recap of degree reduction of Binary CSP
  [Papadimitriou-Yannakakis. *J. Comput. Syst. Sci.*'91] also used by [Dinur. *J. ACM*'07]

- Simple application of ⤴ to Binary CSP Reconfiguration
  looses perfect completeness

- TRICK: **Alphabet squaring**
  - 😋 Preserves perfect completeness
  - 😖 But, NOT a Karp reduction

- Sketching soundness proof
  - Why we use expander mixing lemma & near-Ramanujan graphs

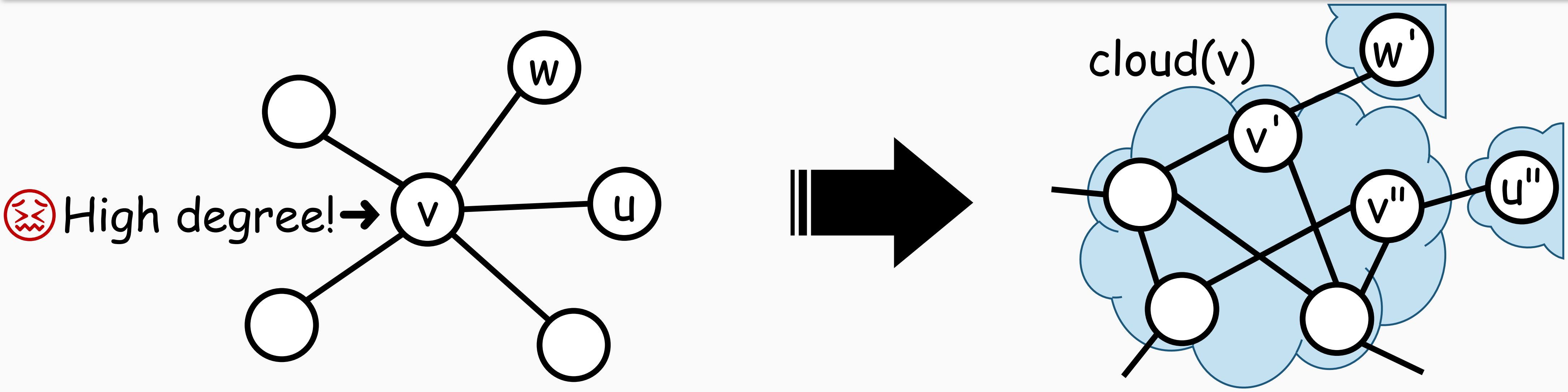
Recap of degree reduction of Binary CSP

[Papadimitriou-Yannakakis. *J. Comput. Syst. Sci.*'91]

- **Input:** Binary CSP $G=(V,E,\Sigma,\Pi=(\pi_e)_{e\in E})$
- **Output:** $\psi: V\to\Sigma$
- **Goal:** $\max_\psi$ (frac. of edges satisfied by ψ)

  $OPT(G) \stackrel{\text{def}}{=}$ value of ⤴

  ψ satisfies (v,w) if $(\psi(v), \psi(w)) \in \pi_{(v,w)}$

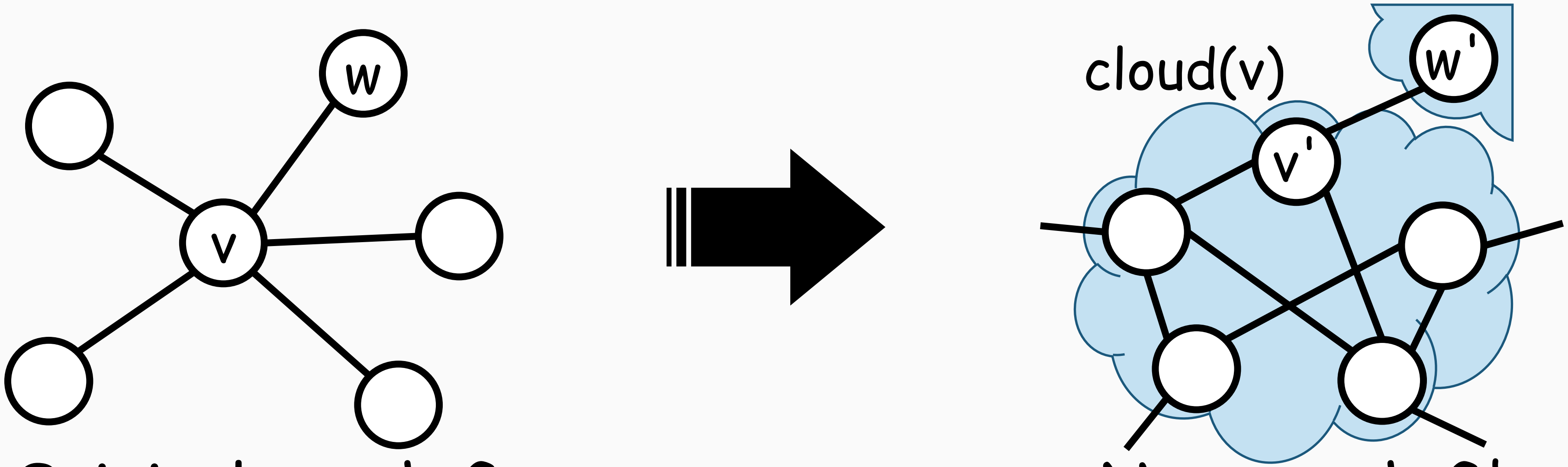


Original $G=(V,E,\Sigma,\Pi)$

New $G'=(V',E',\Sigma,\Pi')$

Recap of degree reduction of Binary CSP

[Papadimitriou-Yannakakis. *J. Comput. Syst. Sci.*'91]



Original graph G — New graph G'

- Completeness: $\mathrm{OPT}(G) = 1 \implies \mathrm{OPT}(G') = 1$
- Soundness: $\mathrm{OPT}(G) < 1\text{-}\varepsilon \implies \mathrm{OPT}(G') < 1\text{-}\varepsilon'$

cloud(v) SHOULD behave like a single assignment

- Equality constraints on intra-cloud edges
- cloud(v) should be sparse yet well-connected ➜ 😋expander graphs

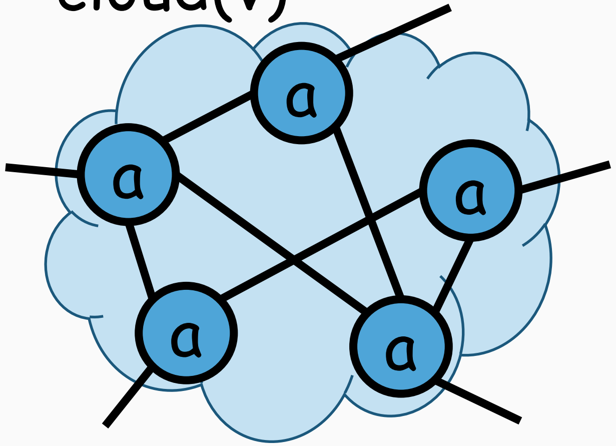
Loosing perfect completeness on Maxmin Binary CSP Reconfiguration

- **Input:** Binary CSP $G=(V,E,\Sigma,\Pi=(\pi_e)_{e\in E})$, satisfying ψ_s, ψ_t: $V\to\Sigma$
- **Output:** $\psi = \langle \psi^{(0)}=\psi_s, \ldots, \psi^{(\ell)}=\psi_t \rangle$ s.t. $\mathrm{Ham}(\psi^{(i-1)}, \psi^{(i)}) = 1$
- **Goal:** $\max_\psi \min_i$ (frac. of edges satisfied by $\psi^{(i)}$)

  $\mathrm{OPT}_G(\psi_s \leftrightsquigarrow \psi_t) \stackrel{\text{def}}{=}$ value of ⤴

⚠ $\mathrm{OPT}_G(\psi_s \leftrightsquigarrow \psi_t) = 1 \not\Rightarrow \mathrm{OPT}_{G'}(\psi'_s \leftrightsquigarrow \psi'_t) = 1$

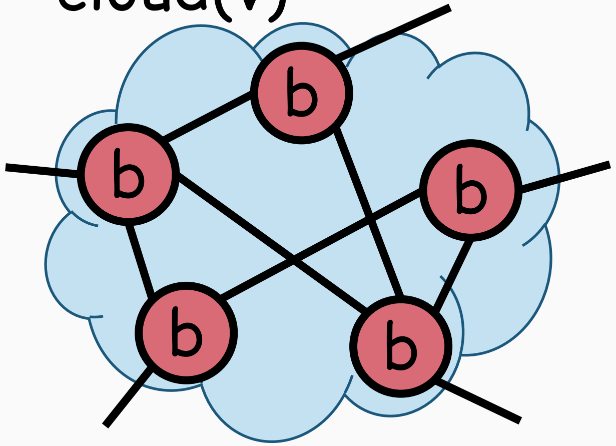
cloud(v)



?

$\leftrightsquigarrow$

cloud(v)

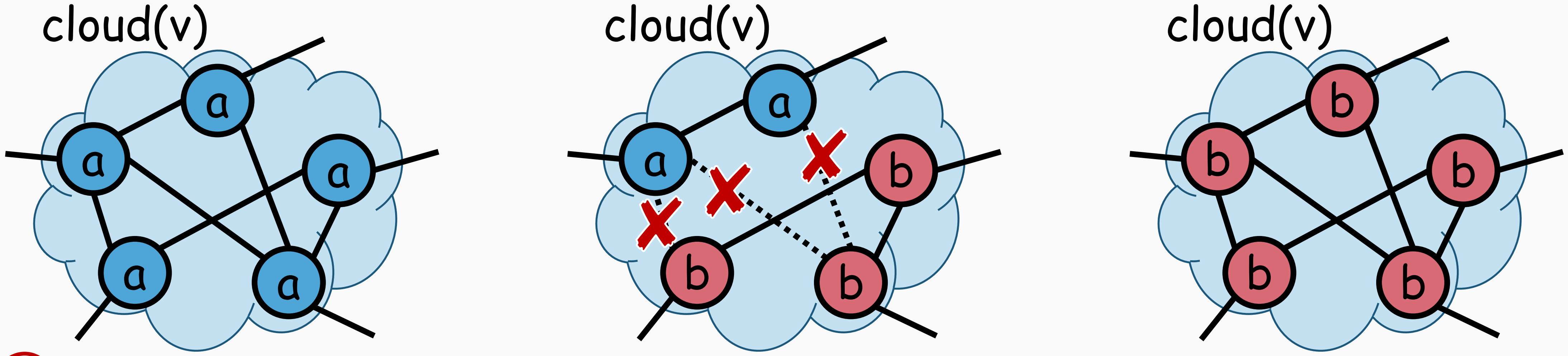


Loosing perfect completeness on Maxmin Binary CSP Reconfiguration

- **Input:** Binary CSP $G=(V,E,\Sigma,\Pi=(\pi_e)_{e\in E})$, satisfying $\psi_s, \psi_t: V\to\Sigma$
- **Output:** $\psi = \langle\psi^{(0)}=\psi_s, \ldots, \psi^{(\ell)}=\psi_t\rangle$ s.t. $\mathrm{Ham}(\psi^{(i-1)}, \psi^{(i)}) = 1$
- **Goal:** $\max_\psi \min_i$ (frac. of edges satisfied by $\psi^{(i)}$)

  $\mathrm{OPT}_G(\psi_s \leftrightsquigarrow \psi_t) \stackrel{\text{def}}{=}$ value of ⤴

⚠ $\mathrm{OPT}_G(\psi_s \leftrightsquigarrow \psi_t) = 1 \not\Rightarrow \mathrm{OPT}_{G'}(\psi'_s \leftrightsquigarrow \psi'_t) = 1$

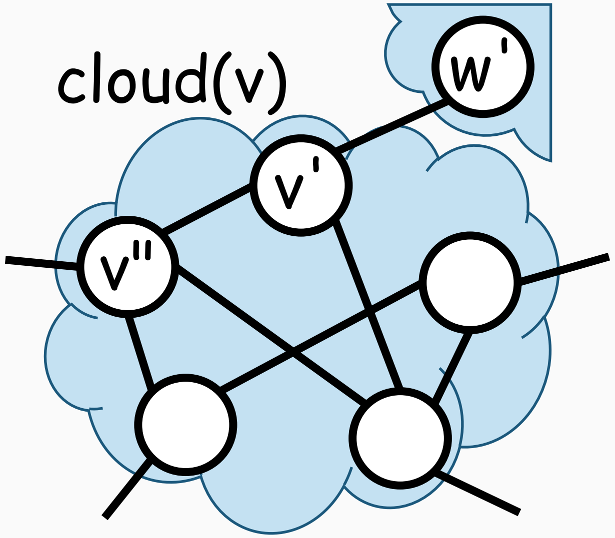


Cannot reconfigure without violating any equality constraints

TRICK: Alphabet squaring

Think as if vertex could take a pair of values!

- Original $\Sigma = \{a, b, c\}$
- New $\Sigma' = \{a, b, c, \mathbf{ab}, \mathbf{bc}, \mathbf{ca}\}$



Constraint for inter-cloud edge e'=(v',w')

- Original $\pi_e = \{(a,b), (a,c)\}$
- New $\pi'_{e'} = \{(a,b), (a,c), \mathbf{(a,bc)}\}$

Equality-LIKE constraint for intra-cloud edge e'=(v',v")

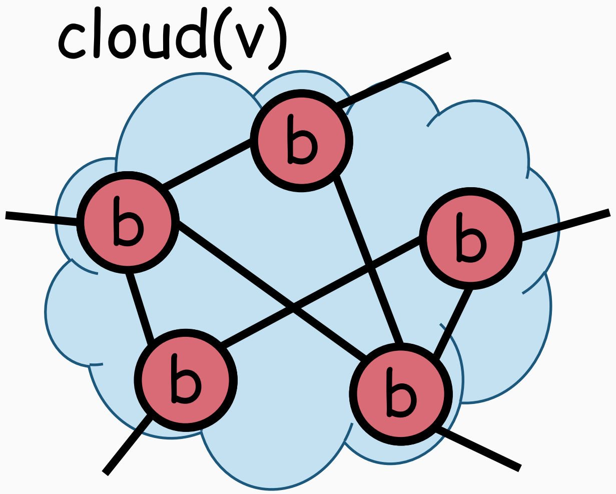
- $\pi'_{e'} = \{(\alpha, \beta) : \alpha \subseteq \beta \text{ or } \beta \subseteq \alpha\}$

$= \{(a,a), (b,b), (c,c), (ab,a), (ab,b), (bc,b), (bc,c), (ca,c), (ca,a), (a,ab), (b,ab), (b,bc), (c,bc), (c,ca), (a,ca), (ab,ab), (bc,bc), (ca,ca)\}$

Alphabet squaring preserves perfect completeness

Think as if vertex could take a pair of values!

- Original $\Sigma = \{a, b, c\}$
- New $\Sigma' = \{a, b, c, \mathbf{ab}, \mathbf{bc}, \mathbf{ca}\}$



Constraint for inter-cloud edge e'=(v',w')

- Original $\pi_e = \{(a,b), (a,c)\}$
- New $\pi'_{e'} = \{(a,b), (a,c), \mathbf{(a,bc)}\}$

Equality-LIKE constraint for intra-cloud edge e'=(v',v")

- $\pi'_{e'} = \{(\alpha, \beta) : \alpha \subseteq \beta \text{ or } \beta \subseteq \alpha\}$

$= \{(a,a), (b,b), (c,c), (ab,a), (ab,b), (bc,b), (bc,c), (ca,c), (ca,a), (a,ab), (b,ab), (b,bc), (c,bc), (c,ca), (a,ca), (ab,ab), (bc,bc), (ca,ca)\}$

😱 Alphabet squaring is NOT a Karp reduction

- **Input**: Binary CSP $G=(V,E,\Sigma,\Pi=(\pi_e)_{e\in E})$, satisfying $\psi_s, \psi_t: V \to \Sigma$
- **Output**: $\psi = \langle \psi^{(0)}=\psi_s, \ldots, \psi^{(\ell)}=\psi_t \rangle$ s.t. $\mathrm{Ham}(\sigma^{(i-1)}, \sigma^{(i)}) = 1$
- **Goal**: $\max_\psi \min_i$ (frac. of edges satisfied by $\psi^{(i)}$)

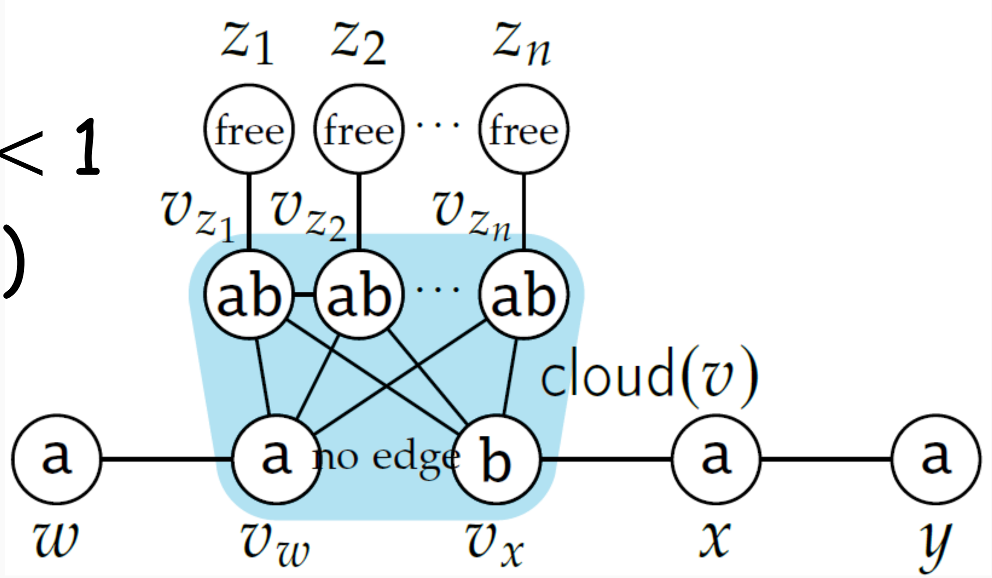
  $\mathrm{OPT}_G(\psi_s \leftrightsquigarrow \psi_t) \stackrel{\text{def}}{=}$ value of ↗

Apply degree reduction with AS

⚠ $\mathrm{OPT}_G(\psi_s \leftrightsquigarrow \psi_t) < 1 \not\Rightarrow \mathrm{OPT}_{G'}(\psi'_s \leftrightsquigarrow \psi'_t) < 1$

- ∵ Can assign conflicting values to cloud(v) without sacrificing any constraint

  (See my paper for concrete example ➜)



Sketch of soundness proof

Observation

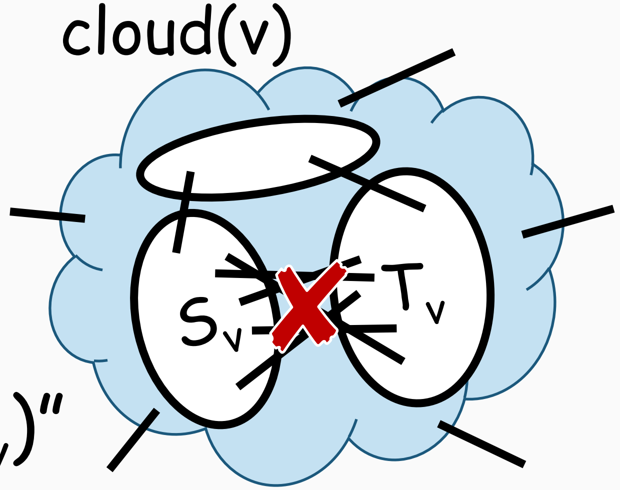
Goal: $\mathrm{OPT}_G(\psi_s \leftrightsquigarrow \psi_t) < 1-\varepsilon \implies \mathrm{OPT}_{G'}(\psi'_s \leftrightsquigarrow \psi'_t) < 1-\varepsilon'$

$\psi = \langle \psi^{(0)}, \ldots, \psi^{(\ell)} \rangle \xleftarrow{\text{plurality vote}}$ Optimal $\psi' = \langle \psi'^{(0)}, \ldots, \psi'^{(\ell)} \rangle$

- Let $\psi \stackrel{\text{def}}{=} \psi^{(i)}$ & $\psi' \stackrel{\text{def}}{=} \psi'^{(i)}$ s.t. "$\psi^{(i)}$ is the WORST assignment"

  $D_v \stackrel{\text{def}}{=}$ {vertices of cloud(v) disagreeing $\psi(v)$}
- Can assume $\Sigma_v |D_v| = O(\varepsilon|E|)$

  similarly to [Papadimitriou-Yannakakis. *J. Comput. Syst. Sci.* '91]



😋 $\exists S_v$ & T_v of size $O(|D_v|)$ s.t. "ψ' violates all $E(S_v,T_v)$"

Sketch of soundness proof

Bounding # of (violated) edges

- How large is $E(S_v,T_v)$? ➜ **Expander mixing lemma!** [Alon-Chung. *Discret. Math.* '88]

degree of cloud(v)

$$|E(S_v,T_v)| \geq \frac{d|S_v|\cdot|T_v|}{|\text{cloud}(v)|} - \boxed{\lambda\sqrt{|S_v|\cdot|T_v|}}$$

2nd eigval of cloud(v)

⚠ Should be small

- Get $\lambda/d \ll \varepsilon$ by setting $\lambda\approx\varepsilon^{-1}$ & $d\approx\varepsilon^{-2}$ using **near-Ramanujan graphs** [Alon. *Comb.* '21] [Mohanty-O'Donnell-Paredes. *SIAM J. Comput.* '21]
- If $|D_v| = O(\varepsilon\cdot\text{cloud}(v))$ (hold for many v)

😋 $|E(S_v,T_v)| = O(|D_v|)$

Taking sum over v (of large D_v) derives

$$(\text{total frac. of violated edges in } G') > \varepsilon' > 0$$

Conclusion and future work

Combinatorial reconfiguration × Hardness of approximation

🤪 *"Is Reconfiguration Inapproximability Hypothesis true...?*

- Use gap amplification? [Dinur. *J. ACM* '07]
- Reduce from **PSPACE**-hard inapproximable problems? [Condon-Feigenbaum-Lund-Shor. *Chic. J. Theor. Comput. Sci.* '95]
- Adapt a Karp reduction from TQBF to Nondeterministic Constraint Logic? [Hearn-Demaine. *Theor. Comput. Sci.* '05]
- Even if false, **NP**-hardness of approximation