Michael Forbes
miforbes@illinois.edu

2019-04-02.4 → 2019-04-04.1
2019-04-04.2 ← CS57

CS579 Computational Complexity : Lecture 22

admin = ps 5 due

ps6 at tonight

last time - #SAT ∈ IP ⇒ P^#P ⊆ IP ⊆ PSPACE

↖ decision problem

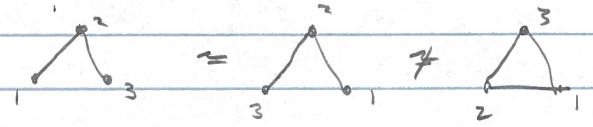
today: - graph non isomorphism ∈ IP

- TQBF ∈ IP ⇒ PSPACE ⊆ IP ⊆ PSPACE

def: $G_i = ([n], E_i)$ $i = 1, 2$ undirected graphs

G_1, G_2 are isomorphic if permutation $\pi : [n] \leftrightarrow [n]$ s.t.

$(i,j) \in E_1$ iff $(\pi(i), \pi(j)) \in E_2$

eg - 

[I relabel/invert]

def - $GI = \{ \langle G_1, G_2 \rangle : G_1 \cong G_2 \} \in NP$

$GNI = \overline{GI} \notin coNP$

perspective: GI not known in P : GI NP-complete ?

GI easy in practice : GI in P ?

[very hard that NP intermediate ?]

[I not NP-complete not in P ?]
[I long-open problem]
[I unlikely to be NP-complete]
[I probably in P]

Thm [Babai 16] : $GI \in TIME(n^{polylg(n)})$

thm [PPS05] : GI NP-complete ⇒ PH collapses

$GI \in NP \cap IP$

$GNI \in coNP \cap \cdots \subseteq IP$

Q GNI ∈ IP directly ? [first evidence IP stronger than NP]

prop: GNI ∈ IP [2] [2 rounds] [public coins]

pf:

P		G_1, G_2	V
	$\leftarrow$	$\pi(G_b)$	randomly choose - $b \in \{0,1\}$
		$(\pi(i), \pi(j)) \leftarrow (i,j)$	- $\pi : [n] \to [n]$
	$\rightarrow$	$\hat{b}$	accept iff $b = \hat{b}$

complexity: - two rounds

- V efficient

Michael Forbes
mforbes@illinois.edu
2019-04-04-2 ← 2019-04-04.1
→ 2019-04-04.3
CS 579

key fact: permutations form a 'group' — $\pi, \sigma' : [n] \leftrightarrow [n]$ permutation

- composability ⇒ $\pi \circ \sigma' : [n] \leftrightarrow [n]$ permutation, $x \mapsto \pi(\sigma'(x))$
- inverse : $\forall \pi \; \exists \pi^{-1}$ $\pi \circ \pi^{-1} = \pi^{-1} \circ \pi = id$
- associativity : $\forall \pi, \sigma', \tau'$ $\pi \circ (\sigma' \circ \tau') = (\pi \circ \sigma') \circ \tau'$

completeness : $G_1 \not\cong G_2$

$G_1 \cong \pi(G_1)$. if $\pi(G_1) \cong G_2$ then $\exists \sigma : \sigma(\pi(G_1)) = G_2$

$\Rightarrow (\sigma \circ \pi)(G_1) = G_2$

$\Rightarrow G_1 \cong G_2$

$\Rightarrow \pi(G_1) \not\cong G_2$

$\Rightarrow$ prover receives $\pi(G_b)$, $\exists! \hat{b}$ s.t. $G_{\hat{b}} \cong \pi(G_b)$, $\hat{b} = b$

↖ prover sends $\hat{b} = b \Rightarrow V$ accepts

soundness : $G_1 \cong G_2$: $\tau(G_1) = G_2$

. consider $H_1 = \{\pi(G_1)\}_{\pi \text{ rand}}$ π rand var

$H_2 =$ " G_2

Claim : H_1 and H_2 are rand vars with identical distribution

Pf : any σ, $\Pr_{H_1}[H_1 = \sigma(G_1)] = \frac{1}{n!}$

any τ $\Pr_{H_2}[H_2 = \underbrace{\tau(G_2)}_{(\tau \circ \pi)(G_1)}] = \frac{1}{n!}$ =

any σ, $\exists! \tau$ s.t. $\tau \circ \pi = \sigma$, i.e. $\tau = \sigma \circ \pi^{-1}$

$\Rightarrow$ any σ $\Pr_{H_2}[H_2 = \sigma(G_1)] = \frac{1}{n!}$

Sum : prover action — in PSPACE

- deterministic ← best response function

prover receives $H (= \pi(G_b))$ returns $\hat{b} = f(H)$ ← deterministic

$\Pr_{b,\pi}[b = \hat{b}(H)] = \Pr_{b,\pi}[b = f(H) = f(\underbrace{\pi(G_b)}_{\text{indep of } b \text{ in distribution}})]$

$= \Pr_{b,\pi}[b = f(H)]$

$= \Pr[0 = f(H) \wedge b = 0] + \Pr[f(H) = 1 \wedge b = 1]$

↖ indep ↗ ↖ indep ↗

$= \frac{1}{2}\Pr[f(H) = 0] + \frac{1}{2}\Pr[f(H) = 1]$

$= \frac{1}{2}$

Michael Forbes
mforbes@illinois.edu

$\Rightarrow \langle G_1, G_2\rangle \in GNI$: $\Pr[(V \leftrightarrow P)(\langle G_1, G_2\rangle) = 1] = 1$

$\notin$ $= 1/2$ [exact]

Rmk: - crucial that randomness of verifier is private from prover
- alternate protocols exist when verifier's randomness is public
eg sumcheck, as $GNI \in coNP \subseteq P^{\#P}$

P —$f_1(x_1)$→
←α_1— α_1 random, public
—$f_2(\alpha_1, x_2)$→
⋮

— Questions?

Thm [Shamir 1991]: $TQBF \in IP$

Cor: $IP = PSPACE$

Pf: idea: algebraize TQBF

$\Psi = \exists x_1 \forall x_2 \exists x_3 \cdots Q_n x_n\ \varphi(x_1, \ldots, x_n)$

use new functional identity to check — boolean formula

Saw: polynomial f st: - f has polynomial size algebraic circ [random f]
- $f(x) = \varphi(x)$ $\forall x \in \{0,1\}^n$

$\Rightarrow \exists x_1 \forall x_2 \cdots \varphi(x) = \exists x_1 \forall x_2 \cdots Q_n x_n\ f(x)$
($\{0,1\}$ under the quantifiers on both sides) [algebraize]

define $\Psi_i(x_1, \ldots, x_i) = Q_{i+1} x_{i+1} \cdots Q_n x_n\ \varphi(x_{\le i}, x_{>i})$

$\Rightarrow \Psi_i(x_{\le i}) = Q_{i+1} x_{i+1}\ \Psi_{i+1}(x_{\le i+1}) = \begin{cases} \Psi_{i+1}(x_{\le i}, 0) \vee \Psi_{i+1}(x_{\le i}, 1) & Q_{i+1} = \exists \\ \quad \wedge & = \forall \end{cases}$

$\Psi_0(x) = \Psi$

$\Psi_n(x) = \varphi(x)$

algebraize: $f_n(x_{\le n}) = f$

$f_i(x_{\le i}) = \begin{cases} 1 - (1 - f_{i+1}(x_{\le i}, 0))(1 - f_{i+1}(x_{\le i}, 1)) & Q_{i+1} = \exists \\ f_{i+1}(x_{\le i}, 0) \cdot f_{i+1}(x_{\le i}, 1) & = \forall \end{cases}$

P $f_0 = \Psi_0 \in \{0,1\}$

idea: P —$\hat{f}_0$ (Ψ)→ V check $\hat{f}_0 = 1$
—$\hat{f}_1(x_1)$→ check $\hat{f}_0 = \hat{f}_1(0) * \hat{f}_1(1)$ ← coherence needed
←α_1— random
—$\hat{f}_2(\alpha_1, x_2)$→ check $\hat{f}_1(\alpha_1) = \hat{f}_2(\alpha_2, 0) * \hat{f}_2(\alpha_2, 1)$
←α_2—
⋮
←α_n—
—$\hat{f}_n(\alpha_1, \ldots, \alpha_n)$→ check $\hat{f}_n(\alpha_1, \ldots, \alpha_n) = f(\alpha_1, \ldots, \alpha_n)$

Michael Forbes
MForbes@illinois.edu
2019-04-04.4 ← 2019-04-04.3
2019-04-09.1
CS 579

Q: What goes wrong? $\deg f_i \leq ?$

A: $\geq 2^n$

i.e. $f_i(x_{\leq i}) = f_{i+1}(x_{\leq i}, 0) \cdot f_{i+1}(x_{\leq i}, 1)$

↑ degree doubles!

idea: add degree reduction operators

i.e.: $h(x)$ univariate poly $\mapsto (1-x)\,h(0) + x\,h(1) = R_x\,h(x)$

↳ $h(0) = (R_x\,h(x))(0)$ ↑ linear

$h(1) = (R_x\,h(x))(1)$

define $f_i(x_{\leq i}) = R_{x_1} R_{x_2} \cdots R_{x_i}\, f_{i+1}(x_{\leq i}, 0) \cdot f_{i+1}(x_{\leq i}, 1)$

⇒ - f_i is multilinear

- f_i is still correct on $\{x_{\leq i} \in \{0,1\}^i\}$

idea: verify in i rounds

P V

$g_i(y_i)$

β

$g_{i+1}(y_{i+1})$

check $g_i(\beta_i) = \begin{cases} g_{i+1}(0) \cdot g_{i+1}(1) \\ 1 - (1-g_{i+1}(0))(1-g_{i+1}(1)) \\ (1-\beta_i)\,g_{i+1}(0) + \beta_i g_{i+1}(1) \end{cases}$

...

Analysis is similar to sumcheck

see ~~Ecc~~ Sipser

check $g(\bar\beta) = f(\bar\beta)$

we have

Rmk: - #GTQBF ⇒ $P[(V \Leftrightarrow P)(\varphi) = 1] = 1$ ∎

$\Leftarrow$ $\leq$ small

⇒ all languages in IP have interactive proof of perfect completeness