Parameterized Inapproximability Hypothesis under ETH

Venkatesan Guruswami
UC Berkeley

Bingkai Lin
Nanjing University

Xuandi Ren
UC Berkeley

Yican Sun
Peking University

Kewen Wu
UC Berkeley

ETH ⇒ PIH

- ETH (Exponential Time Hypothesis):
  - “3SAT requires $2^{\Omega(n)}$ time”.
- PIH (Parameterized Inapproximability Hypothesis):
  - Hardness of approximating constraint satisfaction problems

Parameterized Complexity

How to cope with an NP-hard problem?

- Associate each instance x with a parameter $k \in \mathbb{N}$
  - $k \ll |x|$
  - Measure complexity over $n = |x|$ and k
- **FPT** (**F**ixed-**P**arameter **T**ractable, Analogue of **P**):
  - Problems that admit $f(k) \cdot n^{O(1)}$ time algorithms for some computable function f

k-Vertex Cover

- Input:
  - $G = (V, E)$ and parameter k
- Output:
  - $\exists v_1, \ldots, v_k \in V$ covering all the edges?

has an $O(2^k \cdot n^{O(1)})$ enumeration algorithm

Efficient for small k!

$\in$ FPT

Parameterized Complexity

How to cope with an NP-hard problem?

- Associate each instance x with a parameter $k \in \mathbb{N}$
  - $k \ll |x|$
  - Measure complexity over $n = |x|$ and k
- **W[1]** (Analogue of **NP**): widely believed **W[1]** $\neq$ **FPT**

k-Clique
- Input:
  - $G = (V, E)$ and parameter k
- Output:
  - $\exists v_1, \dots, v_k \in V$ forming a clique?

Unlikely to have an $f(k) \cdot n^{O(1)}$ time algorithm

No algorithm known with runtime $n^{o(k)}$

W[1] -complete

Parameterized Approximation

Can we find a $\boldsymbol{g(k)}$-approximation in $\boldsymbol{f(k) \cdot n^{O(1)}}$ time, for some computable functions f, g?

E.g., Can we find a $\frac{k}{2}$-clique in a graph with a k-clique?

Parameterized Approximation

Can we find a $\boldsymbol{g(k)}$-approximation in $\boldsymbol{f(k) \cdot n^{O(1)}}$ time, for some computable functions f, g?

- **Optimal** ratio in FPT
- Beat polytime algorithms: $2.611 + \varepsilon$ for $\boldsymbol{k}$-Median, $9 + \varepsilon$ for $\boldsymbol{k}$-Means

Example [Cohen-Addad, Gupta, Kumar, Lee, Li'19]:

- $\left(1 + \frac{2}{e} + \varepsilon\right)$-approximation algorithm for $\boldsymbol{k}$-Median
- $\left(1 + \frac{8}{e} + \varepsilon\right)$-approximation algorithm for $\boldsymbol{k}$-Means

with runtime $\left(\frac{k \log k}{\varepsilon^2}\right)^k \cdot n^{O(1)}$

Parameterized Hardness of Approximation

- $\boldsymbol{k}$-SetCover
  - [Chen-Lin'18, Lin'19, Lin-Ren-Sun-Wang'23a] via threshold graph composition
  - [Karthik-Laekhanukit-Manurangsi'19] via distributed PCP framework
- $\boldsymbol{k}$-Clique
  - [Lin'21, Karthik-Khot'22, Lin-Ren-Sun-Wang'23b] via locally decodable codes
  - [Chen-Feng-Laekhanukit-Liu'23] via Sidon sets
- Max $\boldsymbol{k}$-Coverage
  - [Manurangsi'20] via k-wise agreement testing
- …

Ad-hoc reductions,

tailored to the specific problems!

Parameterized Hardness of Approximation

Unified and powerful machinery for parameterized inapproximability?

Parameterized PCP-type theorem!

Recall: PCP Theorem

Constraint Satisfaction Problem

Input: $\Pi = (X, \Sigma, \Phi)$

- X: variables
- Σ: alphabet
- Φ: constraints

Output:

- $\exists \sigma : X \rightarrow \Sigma$ satisfying all constraints?

val(Π):=max. fraction of constraints satisfied by some assignment

(1 vs δ) gap CSP

Input: a CSP instance $\Pi = (X, \Sigma, \Phi)$

Goal: distinguish val(Π) $= 1$ vs val(Π)$\leq \delta$

- PCP Theorem:
  - For any constant Σ and let $n = |X|$, there is no $n^{O(1)}$ time algorithm for (1 vs 0.9) gap CSP assuming P≠NP.

Parameterized Inapproximability Hypothesis

Constraint Satisfaction Problem

Input: $\Pi = (X, \Sigma, \Phi)$

- X: variables
- Σ: alphabet
- Φ: constraints

Output:

- $\exists \sigma: X \to \Sigma$ satisfying all constraints?

val(Π):=max. fraction of constraints satisfied by some assignment

$(1 \text{ vs } \delta)$ gap CSP

Input: a CSP instance $\Pi = (X, \Sigma, \Phi)$

Goal: distinguish val(Π) $= 1$ vs val(Π)$\leq \delta$

- Parameterized CSP:
  - $k = |X|$ and $n = |\Sigma|$, is there an $f(k) \cdot n^{O(1)}$ time algorithm?
  - Example: **Multi-colored k-Clique**

<u>PIH (Parameterized Inapproximability Hypothesis)</u> [Lokshtanov-Ramanujan-Saurabh-Zehavi'20]:
Let $k = |X|$ and $n = |\Sigma|$, there is no $f(k) \cdot n^{O(1)}$ time algorithm for (1 vs 0.9) gap parameterized CSP.

Parameterized Inapproximability Hypothesis

- The analogue of PCP theorem here is W[1]≠FPT ⇒ PIH
- It was known [Dinur-Manurangsi'18] that Gap-ETH ⇒ PIH
  - **Gap-ETH**: "Constant approximating Max3SAT requires $2^{\Omega(n)}$ time"

- **Open Question:** Can we prove **PIH** under some **gap-free** hypothesis?
- This work: ETH ⇒ PIH
  - **ETH**: "3SAT requires $2^{\Omega(n)}$ time"

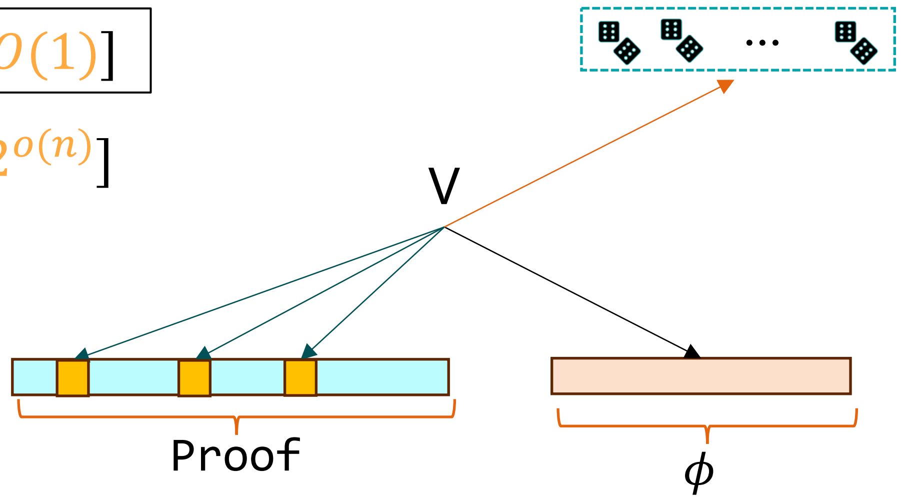
ETH ⇒ PIH

Goal: Reduce 3SAT to a (1 vs 0.9) parameterized CSP with

- $f(k) \ll n$ variables
- Alphabet size $\Sigma = 2^{o(n)}$

Classical PCP: 3SAT ∈ PCP $[1, 0.9, O(\log n), O(1), O(1)]$

- Equivalent to prove: 3SAT ∈ PCP $[1, 0.9, \log f(k), O(1), 2^{o(n)}]$
  - 1: completeness
  - 0.9: soundness
  - $\log f(k)$: randomness
  - $O(1)$: query complexity
  - $2^{o(n)}$: proof alphabet



If ϕ ∈3SAT then ∃ proof $\Pr[V \text{ accepts}] = 1$

If ϕ ∉3SAT then ∀ proof $\Pr[V \text{ accepts}] \leq 0.9$

ETH ⇒ PIH

Goal: Reduce 3SAT to a (1 vs 0.9) parameterized CSP with

- $f(k) \ll n$ variables
- Alphabet size $\Sigma = 2^{o(n)}$

- In particular, we prove:

$$3SAT \in PCP\left[1, 0.9, k^4, O(1), 2^{O\left(\frac{n}{k}\right)}\right]$$

$3SAT \in PCP[1, 0.9, k^4, O(1), 2^{O\left(\frac{n}{k}\right)}]$

Given a 3CNF ϕ, Typical PCP Proof:

Encoding of a 3CNF solution x	Auxiliary proof

Verify the proof:

- Test if the first part is a codeword
- Test the second part to see if x satisfies ϕ

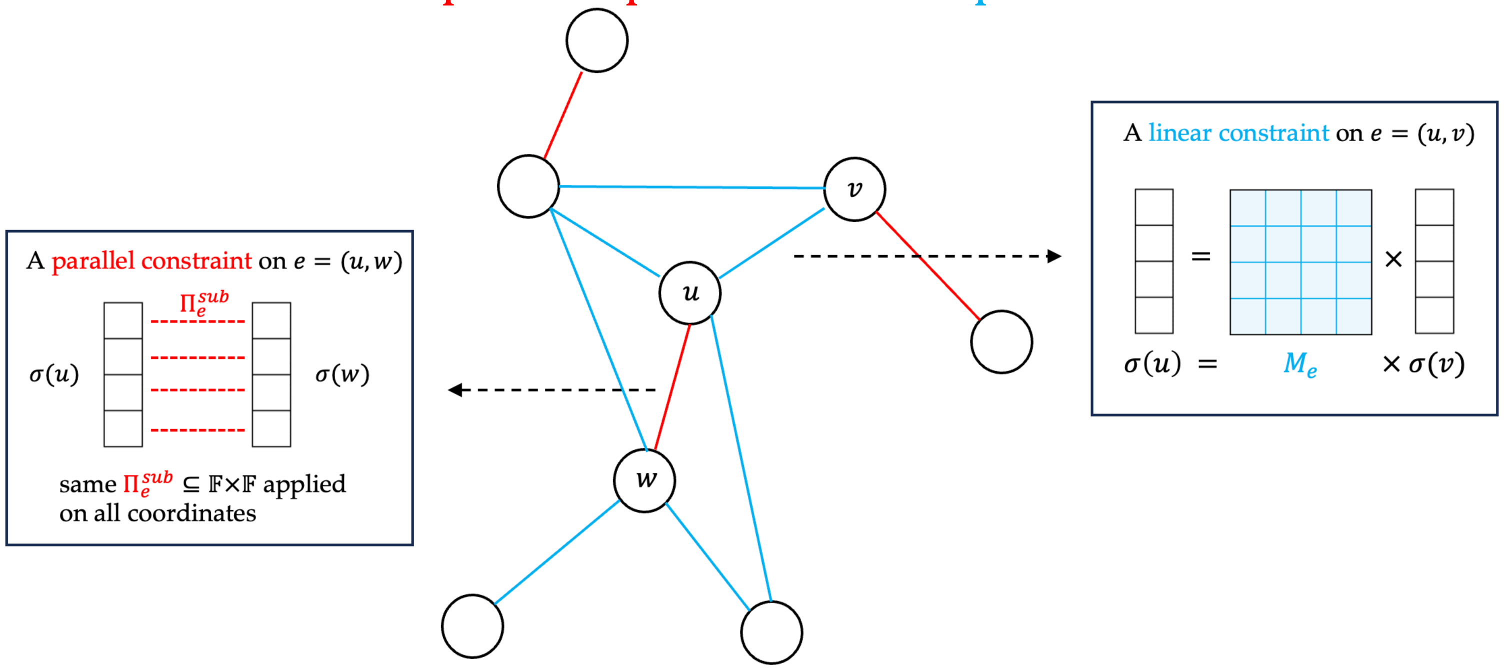
Hadamard encoding: randomness $= \text{poly}(n)$
Reed-Muller encoding: randomness $= O(\log n)$

Far beyond k^4

Vectorization

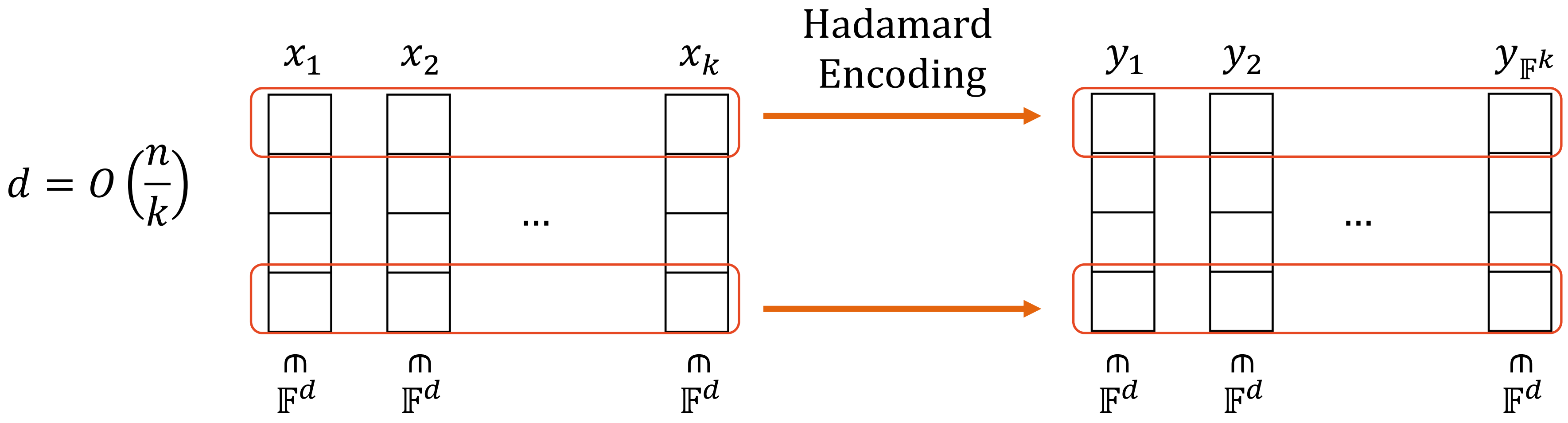
Reduce from 3SAT to Vector-valued CSP:

- $|V| = O(k)$,
- $\Sigma = \mathbb{F}^d$ - d-dimensional vectors over a finite field $\mathbb{F}$
  - $|\mathbb{F}| = O(1),\ d = O\left(\frac{n}{k}\right)$
- Constraint are divided into parallel part and linear part



Parallel Encoding

Given a vector-valued CSP with variables $\{x_1, \dots, x_k\}$:



Admit parallel codeword testing and verification!

Randomness only depends on k!

Recent Improvement

Theorem: Assuming ETH, (1 vs. 1-ε) parameterized CSP requires $\Sigma^{k^{1-o(1)}}$ time.

- A more compact reduction from 3-Coloring, with only a linear blow-up in k
- Reed-Muller encoding of the solution
- Succinct PCPs [Ben-Sasson, Goldreich, Harsha, Sudan, Vadhan'06], used in a black-box way

Open Questions

- Prove **PIH** under **W[1]≠FPT**?
  - Barrier: Vector-valued CSP seems to lie in **M[1]**, a subclass of **W[1]**

- More inapproximability results from **PIH**

Thanks for listening!