Parameterized Inapproximability for Min-CSP: From Baby PIH to Average Baby PIH

Xuandi Ren
UC Berkeley

Workshop on Hardness of Approximation in P
@DIMACS, Rutgers

Outline

- Background
  - Parameterized Complexity
  - Constraint Satisfaction Problem (CSP)
  - Parameterized Inapproximability Hypothesis (PIH)
- From Baby PIH to Average Baby PIH
- Proof Overview
  - Baby PIH
  - Average Baby PIH

Outline

- Background
  - Parameterized Complexity
  - Constraint Satisfaction Problem (CSP)
  - Parameterized Inapproximability Hypothesis (PIH)
- From Baby PIH to Average Baby PIH
- Proof Overview
  - Baby PIH
  - Average Baby PIH

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:
  - $\exists v_1, \dots, v_k \in V$ 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:
  - $\exists v_1, \ldots, v_k \in V$ 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, 6.357 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

- ***k***-SetCover
  - [Chen-Lin’18, Lin’19, Lin-R-Sun-Wang’23a] via threshold graph composition
  - [Karthik-Laekhanukit-Manurangsi’19] via distributed PCP framework
- ***k***-Clique
  - [Lin’21, Karthik-Khot’22, Lin-R-Sun-Wang’23b] via locally decodable codes
  - [Chen-Feng-Laekhanukit-Liu’23] via Sidon sets
- Max ***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:

- $\exists \sigma: X \to \Sigma$ 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 $\boxed{\text{val}(\Pi)} = 1$ 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 vs δ) gap CSP

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

Goal: distinguish $\boxed{\text{val}(\Pi)} = 1$ vs $\text{val}(\Pi) \leq \delta$

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

PIH (Parameterized Inapproximability Hypothesis) [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"
- Recently, it was proven [Guruswami-Lin-R-Sun-Wu'24] that ETH ⇒ PIH
  - **ETH**: "3SAT requires $2^{\Omega(n)}$ time"
  - Further improved by [Bafna-Minzer-Karthik'25, Guruswami-Lin-R-Sun-Wu'25] in terms of smaller parameter blow up

Outline

- Background
  - Parameterized Complexity
  - Constraint Satisfaction Problem (CSP)
  - Parameterized Inapproximability Hypothesis (PIH)
- From Baby PIH to Average Baby PIH
- Proof Overview
  - Baby PIH
  - Average Baby PIH

List Satisfiability of CSP

2-CSP

- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma: X \to 2^{\Sigma}$ list-satisfying all constraints?

List Value

max. list size of a list-satisfying multi-assignment $\sigma: X \to 2^{\Sigma}$

- A constraint on (x, y) is list-satisfied iff $\exists u \in \sigma(x), v \in \sigma(y)$, s.t. (u, v) satisfies this constraint.

List Satisfiability of CSP

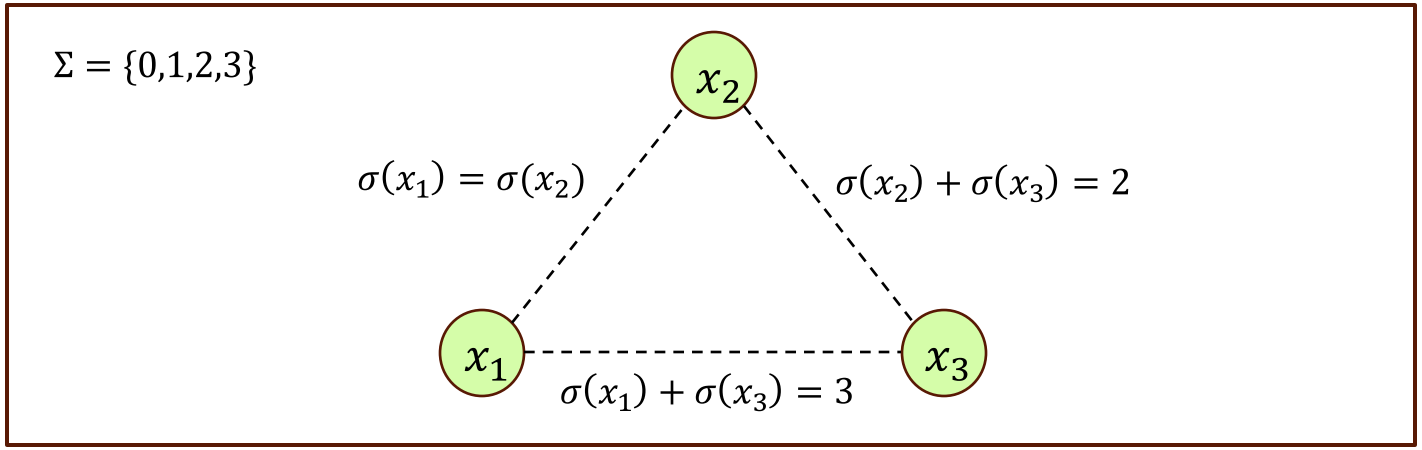
2-CSP

- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma : X \to 2^{\Sigma}$ list-satisfying all constraints?

List Value

max. list size of a list-satisfying multi-assignment $\sigma : X \to 2^{\Sigma}$

- A constraint on (x, y) is list-satisfied iff $\exists u \in \sigma(x), v \in \sigma(y)$, s.t. (u, v) satisfies this constraint.



List Satisfiability of CSP

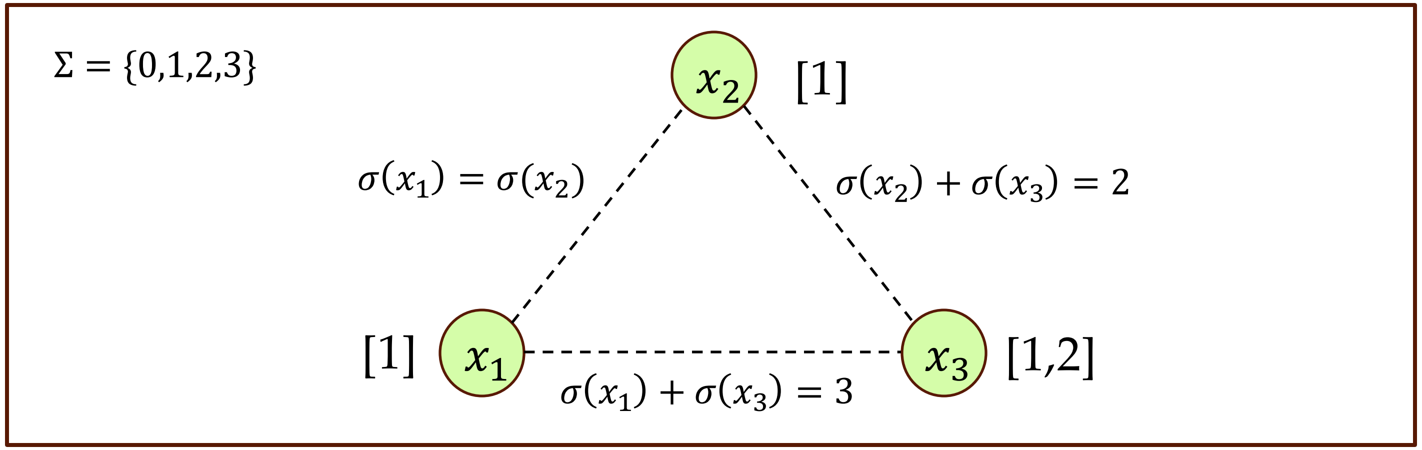
2-CSP

- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma: X \to 2^{\Sigma}$ list-satisfying all constraints?

List Value

max. list size of a list-satisfying multi-assignment $\sigma: X \to 2^{\Sigma}$

- A constraint on (x, y) is list-satisfied iff $\exists u \in \sigma(x), v \in \sigma(y)$, s.t. (u, v) satisfies this constraint.



List Satisfiability of CSP

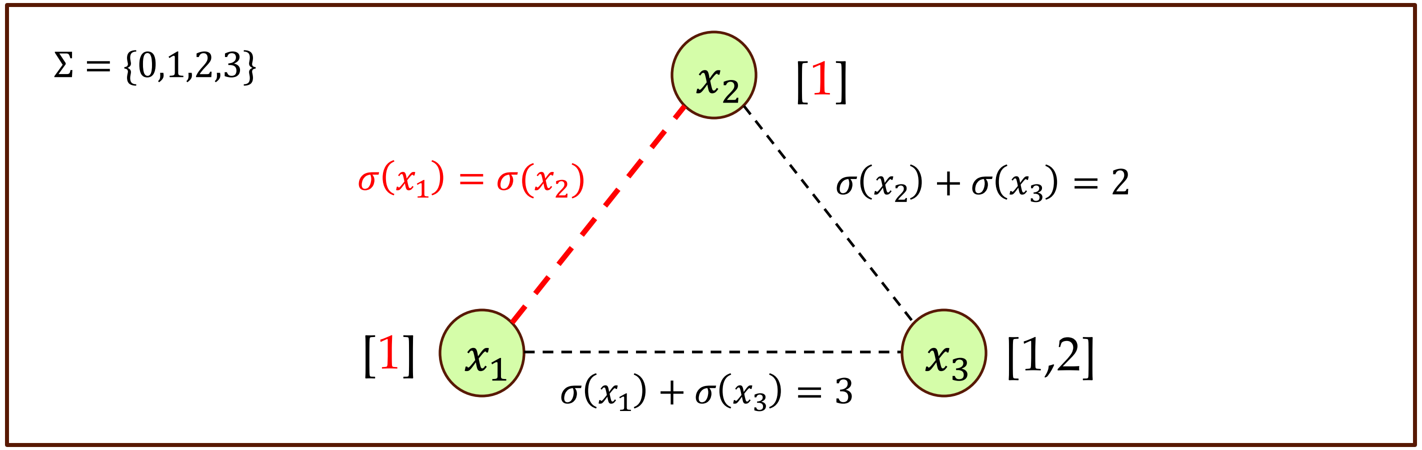
2-CSP

- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma: X \to 2^{\Sigma}$ list-satisfying all constraints?

List Value

max. list size of a list-satisfying multi-assignment $\sigma: X \to 2^{\Sigma}$

- A constraint on (x, y) is list-satisfied iff $\exists u \in \sigma(x), v \in \sigma(y)$, s.t. (u, v) satisfies this constraint.



List Satisfiability of CSP

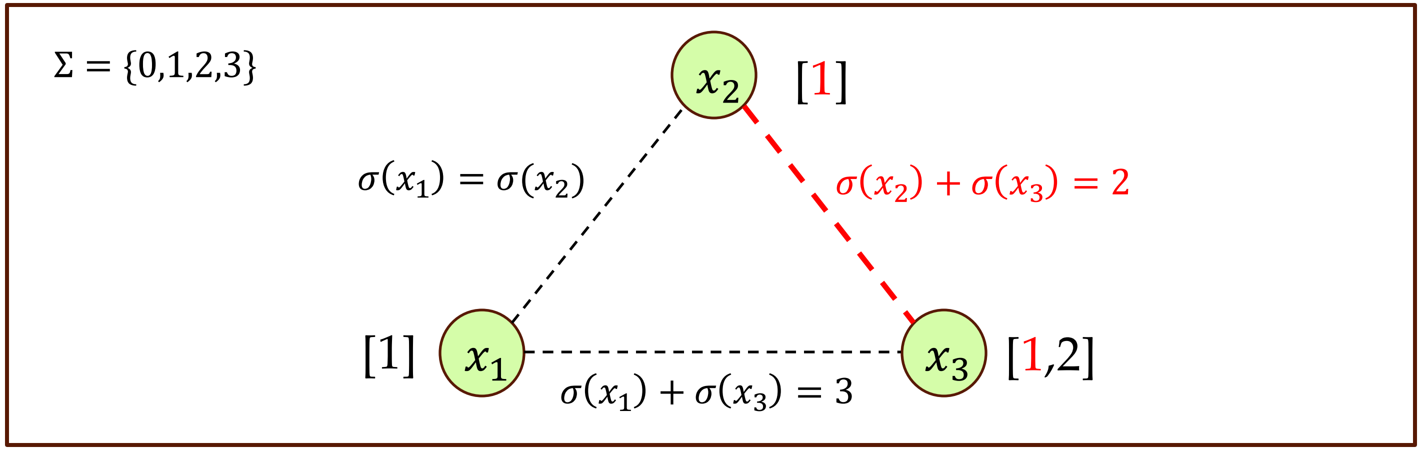
2-CSP

- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma: X \to 2^\Sigma$ list-satisfying all constraints?

List Value

max. list size of a list-satisfying multi-assignment $\sigma: X \to 2^\Sigma$

- A constraint on (x, y) is list-satisfied iff $\exists u \in \sigma(x), v \in \sigma(y)$, s.t. (u, v) satisfies this constraint.



List Satisfiability of CSP

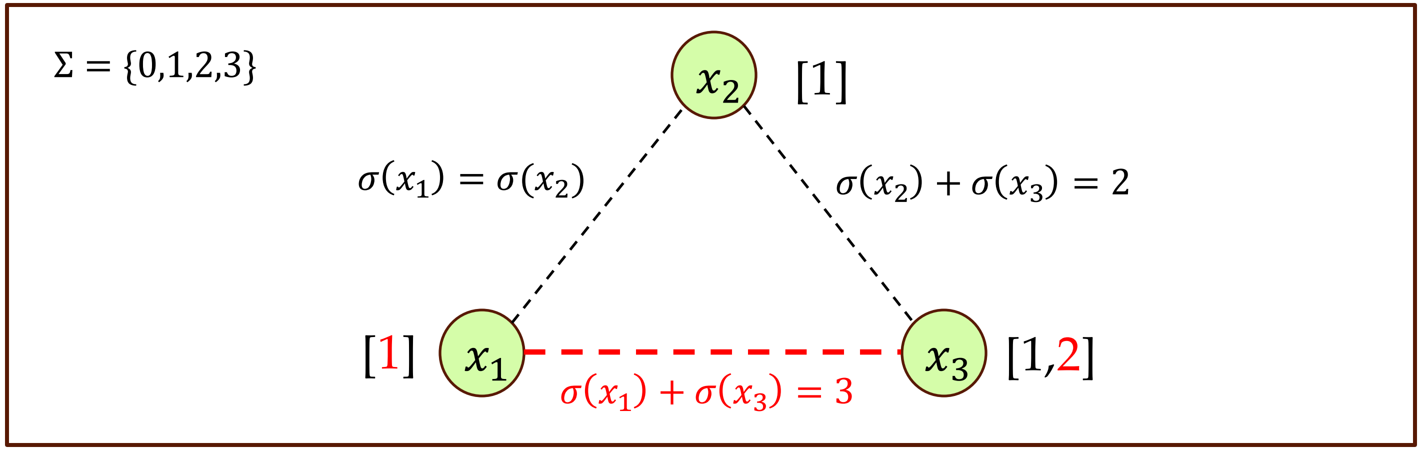
2-CSP

- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma: X \to 2^{\Sigma}$ list-satisfying all constraints?

List Value

max. list size of a list-satisfying multi-assignment $\sigma: X \to 2^{\Sigma}$

- A constraint on (x, y) is list-satisfied iff $\exists u \in \sigma(x), v \in \sigma(y)$, s.t. (u, v) satisfies this constraint.



List Satisfiability of CSP

2-CSP

- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma: X \to 2^{\Sigma}$ list-satisfying all constraints?

List Value

max. list size of a list-satisfying multi-assignment $\sigma: X \to 2^{\Sigma}$

- We say a 2-CSP is r-list satisfiable iff $\exists \sigma$ with $\max_{x \in X} |\sigma(x)| \leq r$ list-satisfying all constraints.

List Satisfiability of CSP

2-CSP

- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma: X \to 2^{\Sigma}$ list-satisfying all constraints?

List Value

max. list size of a list-satisfying multi-assignment $\sigma: X \to 2^{\Sigma}$

- We say a 2-CSP is r-list satisfiable iff $\exists\sigma$ with $\max_{x\in X}|\sigma(x)| \leq r$ list-satisfying all constraints.
  - CSP Value=1 $\Leftrightarrow$ 1-list satisfiable $\Rightarrow$ r-list satisfiable for $r \geq 2$
  - r-list satisfiable $\Rightarrow$ CSP Value $\geq 1/r^2$

Baby PCP

2-CSP
- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma: X \to 2^{\Sigma}$ list-satisfying all constraints?

List Value
max. list size of a list-satisfying multi-assignment $\sigma: X \to 2^{\Sigma}$

- We say a 2-CSP is r-list satisfiable iff $\exists \sigma$ with $\max_{x \in X} |\sigma(x)| \leq r$ list-satisfying all constraints.
  - CSP Value=1 $\Leftrightarrow$ 1-list satisfiable $\Rightarrow$ r-list satisfiable for $r \geq 2$
  - r-list satisfiable $\Rightarrow$ CSP Value $\geq 1/r^2$
- Baby PCP [Barto-Kozik'22]
  - For any $r > 1$, It's NP-hard to distinguish between [1-list satisfiable] and [not even r-list satisfiable].

Baby PCP

2-CSP

- Input: $\Pi = (X, \Sigma, \Phi)$
- Output:
  - $\exists$ multi-assignment $\sigma: X \to 2^{\Sigma}$ list-satisfying all constraints?

List Value

max. list size of a list-satisfying multi-assignment $\sigma: X \to 2^{\Sigma}$

- We say a 2-CSP is r-list satisfiable iff $\exists \sigma$ with $\max_{x \in X} |\sigma(x)| \le r$ list-satisfying all constraints.
  - CSP Value=1 $\Leftrightarrow$ 1-list satisfiable $\Rightarrow$ r-list satisfiable for $r \ge 2$
  - r-list satisfiable $\Rightarrow$ CSP Value $\ge 1/r^2$

- Baby PCP [Barto-Kozik'22]
  - For any $r > 1$, It's NP-hard to distinguish between [1-list satisfiable] and [not even r-list satisfiable].
- $\Leftarrow$ PCP $\Uparrow$ $\Uparrow$ (when $\varepsilon < 1/r^2$)
  - For any $\varepsilon > 0$, It's NP-hard to distinguish between [CSP Value =1] and [CSP Value $<\varepsilon$].

Baby PCP

- Baby PCP [Barto-Kozik'22]
  - Assuming NP≠P, for any $r > 1$, distinguishing between [1-list satisfiable] and [not even r-list satisfiable] cannot be done in $|\Pi|^{O(1)}$ time.
  - (A combinatorial proof)
  - (Enough to prove the NP-hardness of some PCSPs (e.g., $(2+\varepsilon)$-SAT))

Baby PIH

- Baby PCP [Barto-Kozik'22]
  - Assuming NP≠P, for any $r > 1$, distinguishing between [1-list satisfiable] and [not even r-list satisfiable] cannot be done in $|\Pi|^{O(1)}$ time.
  - (A combinatorial proof)
  - (Enough to prove the NP-hardness of some PCSPs (e.g., $(2+\varepsilon)$-SAT))

- Baby PIH [Guruswami-R-Sandeep'24]
  - Assuming W[1]≠FPT, … cannot be done in $f(|X|) \cdot |\Sigma|^{O(1)}$ time.
  - (An itself interesting inapproximability result for list-satisfiability of CSP)
  - (A step towards PIH)
  - (Enough to get some applications of PIH?)

Baby PIH

- Baby PCP [Barto-Kozik'22]
  - Assuming NP≠P, for any $r > 1$, distinguishing between [1-list satisfiable] and [not even r-list satisfiable] cannot be done in $|\Pi|^{O(1)}$ time.
  - (A combinatorial proof)
  - (Enough to prove the NP-hardness of some PCSPs (e.g., $(2+\varepsilon)$-SAT))

- Baby PIH [Guruswami-R-Sandeep'24]
  - Assuming W[1]≠FPT, … cannot be done in $f(|X|) \cdot |\Sigma|^{O(1)}$ time.
  - (An itself interesting inapproximability result for list-satisfiability of CSP)
  - (A step towards PIH)

- Average Baby PIH [Liu-Chen-Li-Lin-Zheng'25]
  - Enough to get some applications of PIH!

Question: Average Baby PIH

$|X| = 4,$

x_1: $\{0,\ 1,\ 3,\ 5,\ 7,\ 8,\ 9\}$

x_2: $\{2\}$

x_3: $\{2,\ 6\}$

x_4: $\{8\}$

$\leq t$

$(t = 7)$

Total # of values: $7 + 1 + 2 + 1 = 11 = 2.75|X|$.

Question: Average Baby PIH

- No FPT algorithm for deciding a 2CSP parameterized by $k = |X|$:

    - Being satisfiable, or

    - Cannot satisfy all constraints simultaneously even when assigning to X less than $t|X|$ values **in total**. $(t > 1)$

ℓ_1 instead of ℓ_∞

- Raised in [Guruswami,Ren,Sandeep'24].

Baby PIH

- Baby PCP [Barto-Kozik'22]
  - Assuming NP≠P, for any $r > 1$, distinguishing between [1-list satisfiable] and [not even r-list satisfiable] cannot be done in $|\Pi|^{O(1)}$ time.
  - (A combinatorial proof)
  - (Enough to prove the NP-hardness of some PCSPs (e.g., $(2+\varepsilon)$-SAT))
- Baby PIH [Guruswami-R-Sandeep'24]
  - Assuming W[1]≠FPT, … cannot be done in $f(|X|) \cdot |\Sigma|^{O(1)}$ time.
  - (An itself interesting inapproximability result for list-satisfiability of CSP)
  - (A step towards PIH)
- Average Baby PIH [Liu-Chen-Li-Lin-Zheng'25]
  - ⇒ constant inapprox. for ***k*-ExactCover** and ***k*-Nearest Codeword Problem** under W[1]≠FPT
  - PIH ⇒ Average Baby PIH ⇒ Baby PIH

Outline

- Background
  - Parameterized Complexity
  - Constraint Satisfaction Problem (CSP)
  - Parameterized Inapproximability Hypothesis (PIH)
- From Baby PIH to Average Baby PIH
- Proof Overview
  - Baby PIH
  - Average Baby PIH

Proof Overview – Baby PIH

- Follows from and extends [Barto-Kozik'22]'s proof of Baby PCP Theorem
- Direct Product Construction

2-CSP
$\Pi = (X, \Sigma, \Phi)$

$\longrightarrow$

t-wise Direct Product 2-CSP
$\Pi^{\odot t} = \left(\binom{X}{t}, \Sigma^t, \Phi'\right)$

partial satisfying assignments
for the set of variables

Proof Overview – Baby PIH

- Follows from and extends [Barto-Kozik'22]'s proof of Baby PCP Theorem
- Direct Product Construction

2-CSP
$\Pi = (X, \Sigma, \Phi)$

$\longrightarrow$

t-wise Direct Product 2-CSP
$\Pi^{\odot t} = \left(\binom{X}{t}, \Sigma^t, \Phi'\right)$

consistency checks

Proof Overview – Baby PIH

- Follows from and extends [Barto-Kozik'22]'s proof of Baby PCP Theorem
- Direct Product Construction

2-CSP
$\Pi = (X, \Sigma, \Phi)$

$\longrightarrow$

t-wise Direct Product 2-CSP
$\Pi^{\odot t} = \left(\binom{X}{t}, \Sigma^t, \Phi' \right)$

partial satisfying assignments for the set of variables

consistency checks

- (Want to show):
- For any $r > 1$, there exists t depending on r, such that for every Π,
  - (Completeness) If Π is satisfiable, then so is $\Pi^{\odot t}$.
  - (Soundness) If Π is not satisfiable, then $\Pi^{\odot t}$ is not r-list satisfiable.

Proof Overview – Baby PIH

- Follows from and extends [Barto-Kozik'22]'s proof of Baby PCP Theorem
- Direct Product Construction

2-CSP		t-wise Direct Product 2-CSP
$\Pi = (X, \Sigma, \Phi)$	$\longrightarrow$	$\Pi^{\odot t} = \left(\binom{X}{t}, \Sigma^t, \Phi'\right)$

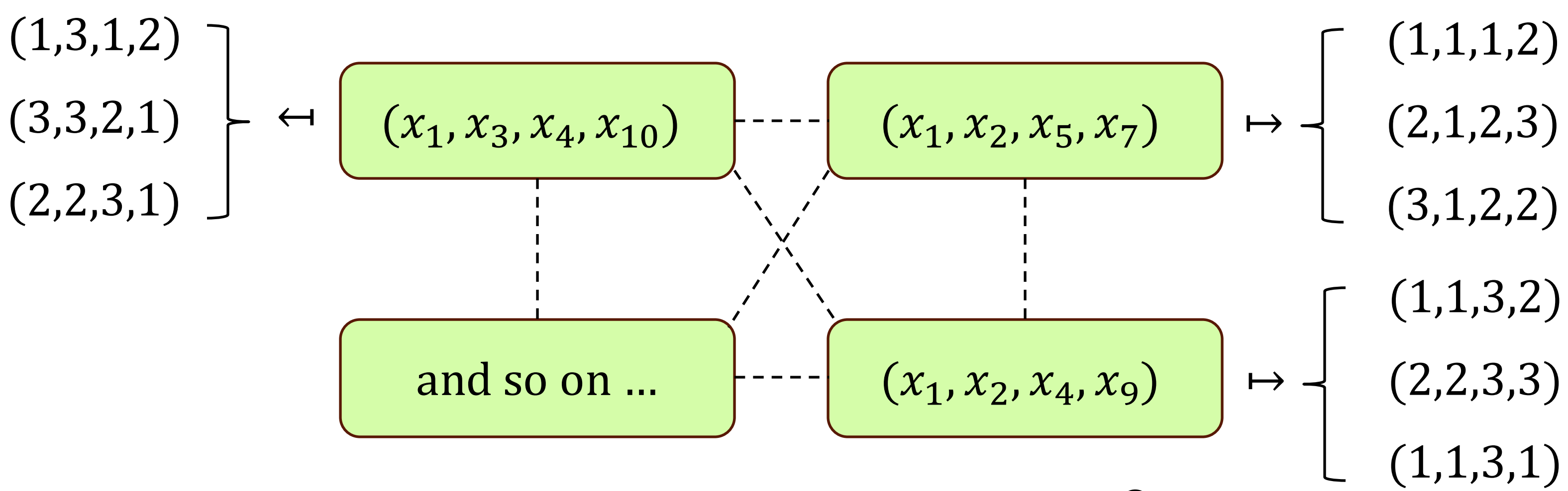
$\binom{X}{t}$: *partial satisfying assignments* for the set of variables

Φ': *consistency checks*

- (Want to show):
- For any $r > 1$, there exists t depending on r, such that for every Π,
  - (Completeness) If Π is satisfiable, then so is $\Pi^{\odot t}$.
  - (Soundness) If Π is not satisfiable, then $\Pi^{\odot t}$ is not r-list satisfiable.

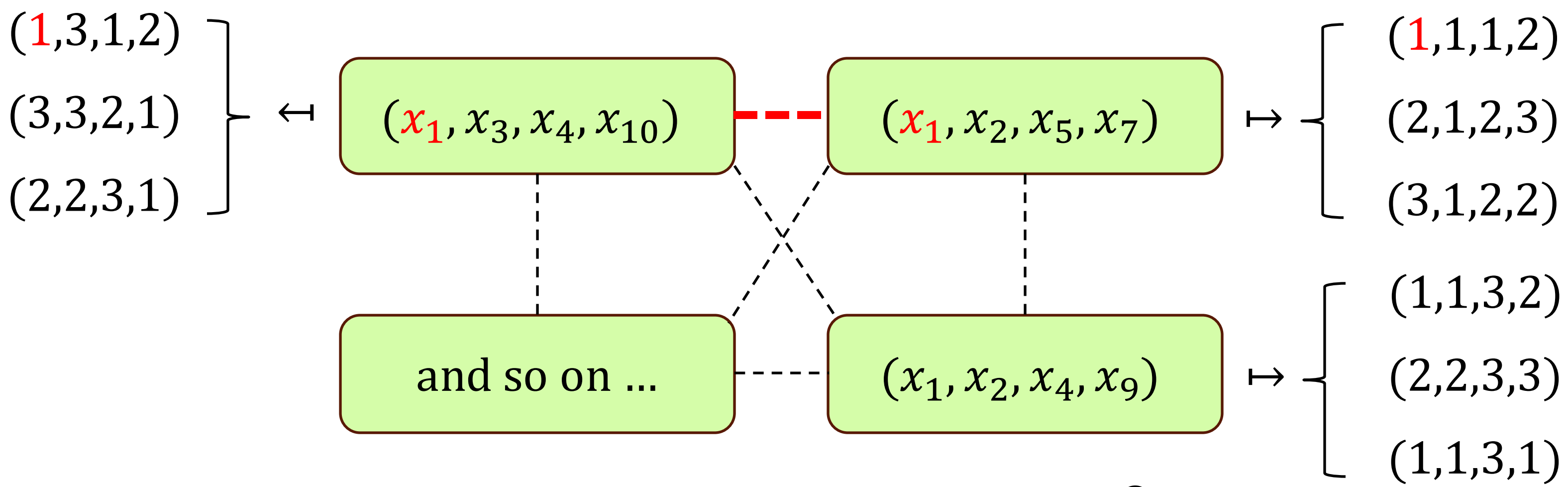
- Reduction time: $n^{O_r(1)}$ where $n = |\Pi|$
  - a unified proof for both Baby PCP and Baby PIH!

Proof Overview – Baby PIH



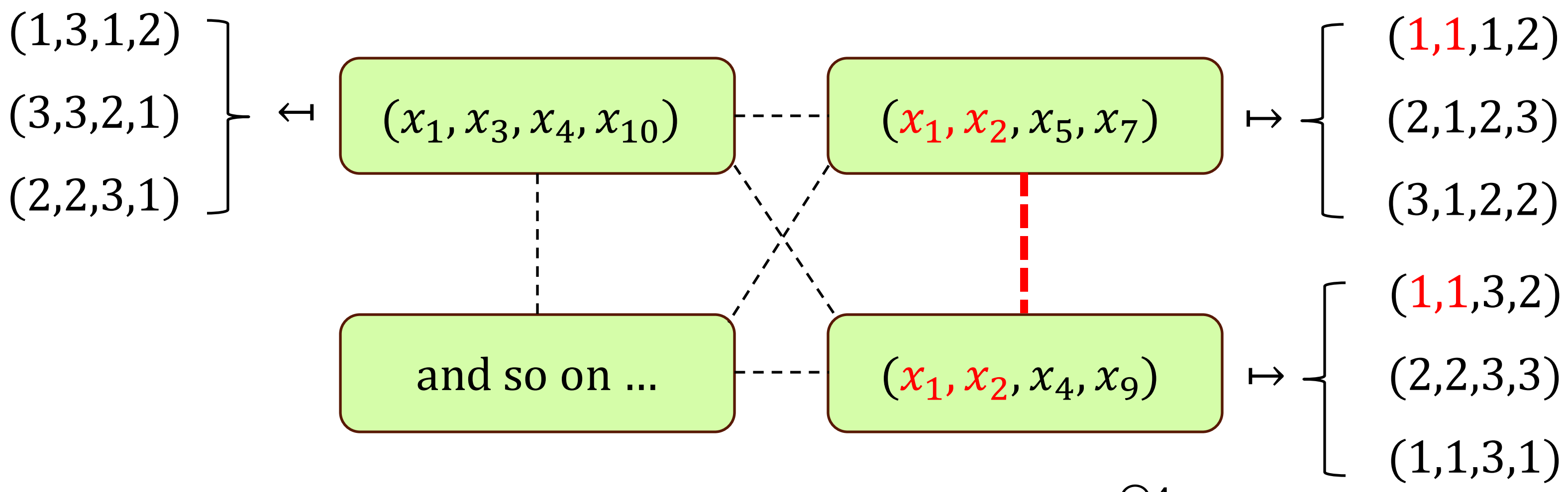
A 3-list satisfying assignment for $\Pi^{\odot 4}$

Proof Overview – Baby PIH



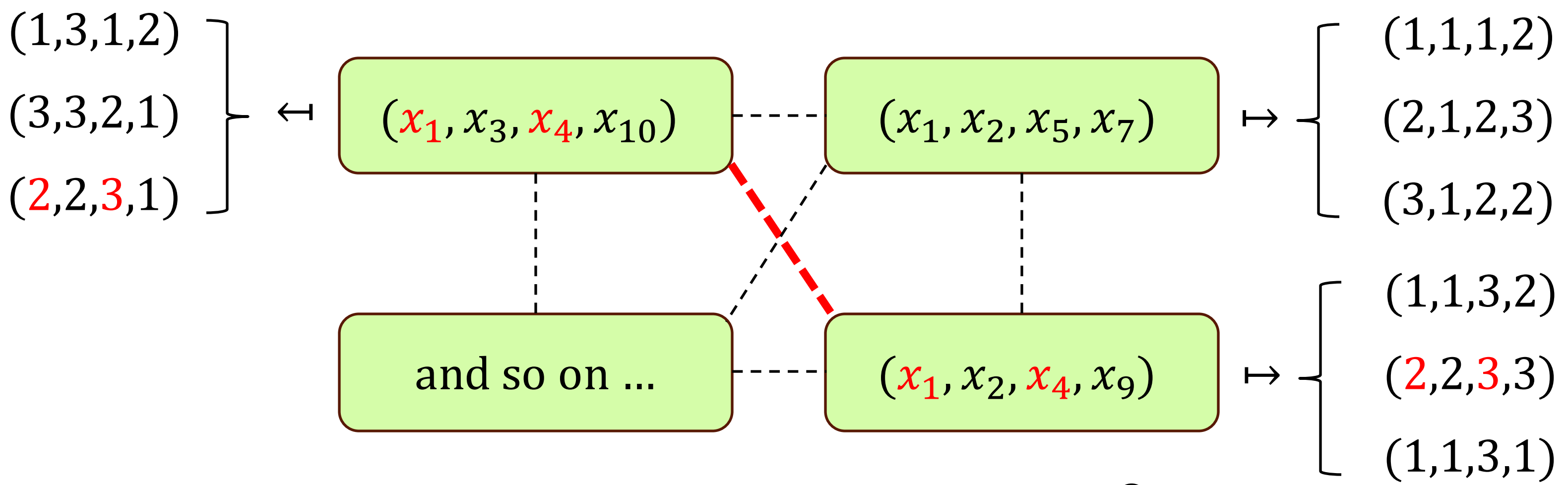
A 3-list satisfying assignment for $\Pi^{\odot 4}$

Proof Overview – Baby PIH



A 3-list satisfying assignment for $\Pi^{\odot 4}$

Proof Overview – Baby PIH



A 3-list satisfying assignment for $\Pi^{\odot 4}$

Proof Overview – Baby PIH

2-CSP
$\Pi = (X, \Sigma, \Phi)$

$\longrightarrow$

t-wise Direct Product 2-CSP
$\Pi^{\odot t} = \left(\binom{X}{t}, \Sigma^t, \Phi'\right)$

- For some sufficiently large $t = t(r)$,
  - given an r-list satisfying multi-assignment σ of $\Pi^{\odot t}$,
  - want to construct an $(r-1)$-list satisfying multi-assignment σ' of $\Pi^{\odot t'}$, for some $t' < t$.

Proof Overview – Baby PIH

2-CSP
$\Pi = (X, \Sigma, \Phi)$

$\longrightarrow$

***t*-wise Direct Product 2-CSP**
$\Pi^{\odot t} = \left(\binom{X}{t}, \Sigma^t, \Phi'\right)$

- For some sufficiently large $t = t(r)$,
  - given an r-list satisfying multi-assignment σ of $\Pi^{\odot t}$,
  - want to construct an $(r-1)$-list satisfying multi-assignment σ' of $\Pi^{\odot t'}$, for some $t' < t$.

- If we end up with the 1-list satisfiability of $\Pi^{\odot(\geq 2)}$, then we are done!

Proof Overview – Baby PIH

2-CSP

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

$\longrightarrow$

t-wise Direct Product 2-CSP

$\Pi^{\odot t} = \left(\binom{X}{t}, \Sigma^t, \Phi'\right)$

- For some sufficiently large $t = t(r)$,
  - given an r-list satisfying multi-assignment σ of $\Pi^{\odot t}$,
  - want to construct an $(r-1)$-list satisfying multi-assignment σ' of $\Pi^{\odot t'}$, for some $t' < t$.
    - for each set $S \in \binom{X}{t'}$, choose a set $T \in \binom{X}{t}$ with $S \subseteq T$
    - the list $\sigma'(S)$ is inherited from the list $\sigma(T)$ (at the hope of decreasing the list size by 1)

- If we end up with the 1-list satisfiability of $\Pi^{\odot(\geq 2)}$, then we are done!

Outline

- Background
  - Parameterized Complexity
  - Constraint Satisfaction Problem (CSP)
  - Parameterized Inapproximability Hypothesis (PIH)
- From Baby PIH to Average Baby PIH
- Proof Overview
  - Baby PIH
  - Average Baby PIH

Proof Overview – Average Baby PIH

- **Lemma** [Karthik-Navon'21, Lin-R-Sun-Wang'23]:
  - Fix any error-correcting code $C \subseteq \Sigma^n$ with relative distance $1 - \delta$.
  - Any subset of codewords $S \subseteq C$ that "collides" on εn positions must have size $\geq \sqrt{\frac{2\varepsilon}{\delta}}$.

- Proof: Simple counting argument.

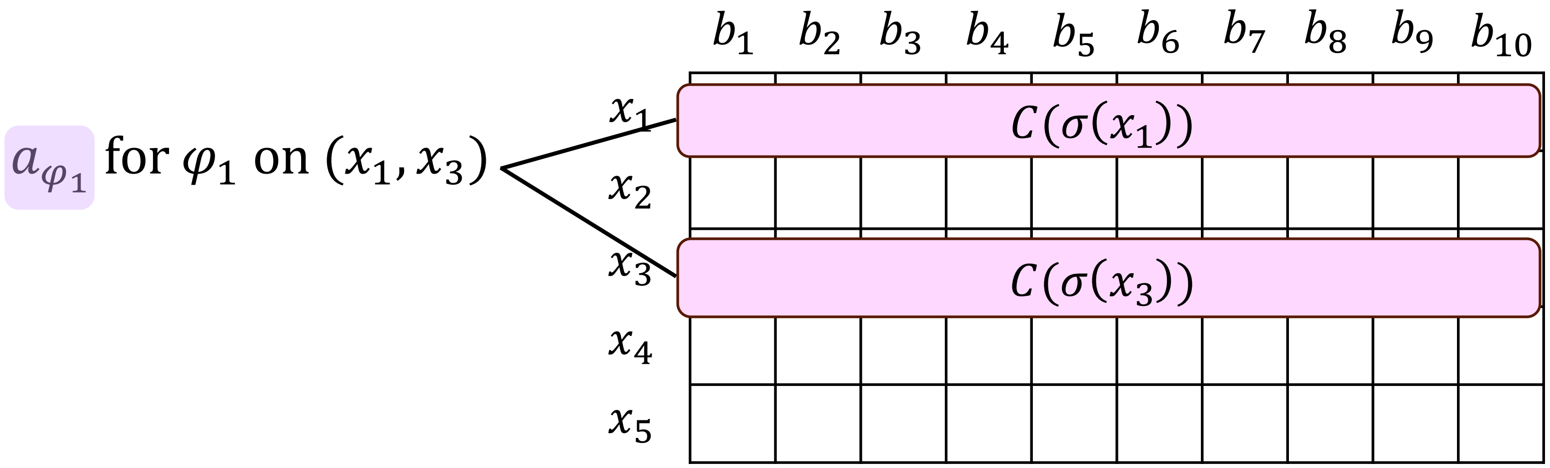
Proof Overview – Average Baby PIH

- Fix an error-correcting code $C: \Sigma \to \mathbb{F}^n$.
- Given a 2CSP instance $\Pi = (X, \Sigma, \Phi)$ that is either satisfiable or not (2r)-list satisfiable,

Proof Overview – Average Baby PIH

- Fix an error-correcting code $C: \Sigma \to \mathbb{F}^n$.
- Given a 2CSP instance $\Pi = (X, \Sigma, \Phi)$ that is either satisfiable or not (2r)-list satisfiable,
  - Variables: $\{a_\varphi | \forall \varphi \in \Phi\} \cup \{b_1, \ldots, b_n\}$

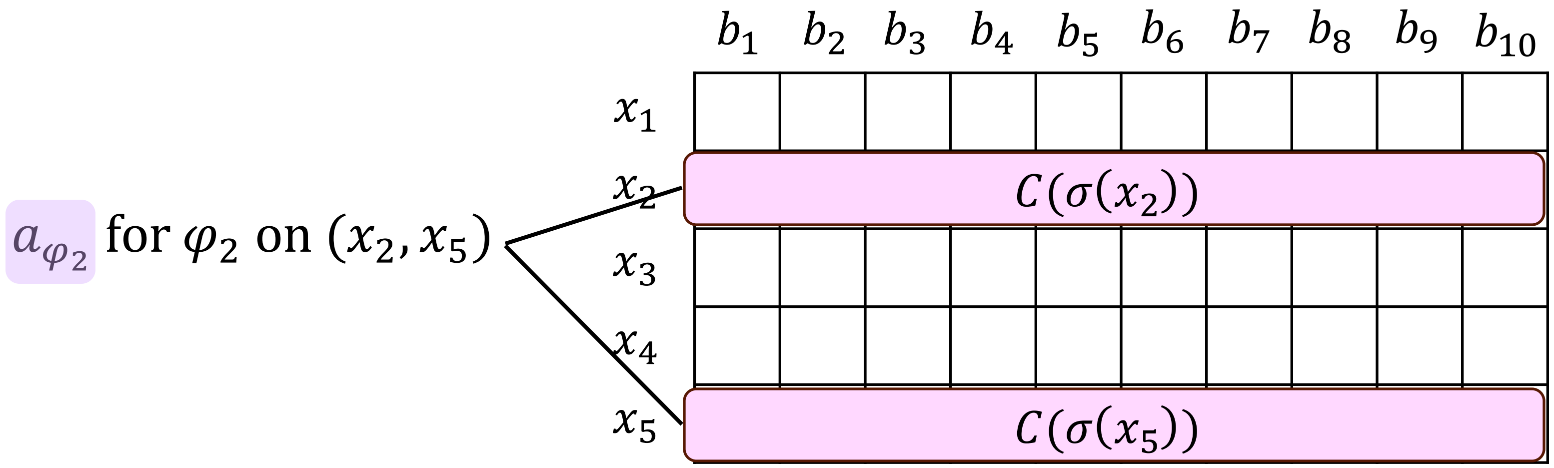
Alphabet: $\mathbb{F}^{2n}$



Proof Overview – Average Baby PIH

- Fix an error-correcting code $C: \Sigma \to \mathbb{F}^n$.
- Given a 2CSP instance $\Pi = (X, \Sigma, \Phi)$ that is either satisfiable or not (2r)-list satisfiable,
  - Variables: $\{a_\varphi | \forall \varphi \in \Phi\} \cup \{b_1, \dots, b_n\}$

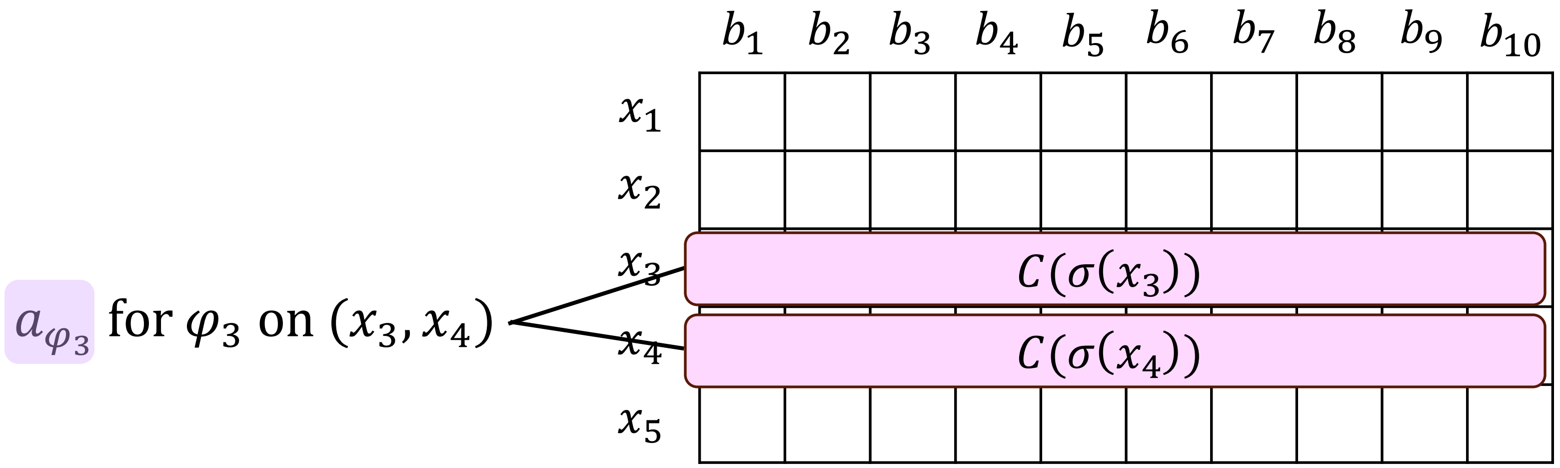
Alphabet: $\mathbb{F}^{2n}$



Proof Overview – Average Baby PIH

- Fix an error-correcting code $C: \Sigma \to \mathbb{F}^n$.
- Given a 2CSP instance $\Pi = (X, \Sigma, \Phi)$ that is either satisfiable or not (2r)-list satisfiable,
  - Variables: $\{a_\varphi | \forall \varphi \in \Phi\} \cup \{b_1, \ldots, b_n\}$

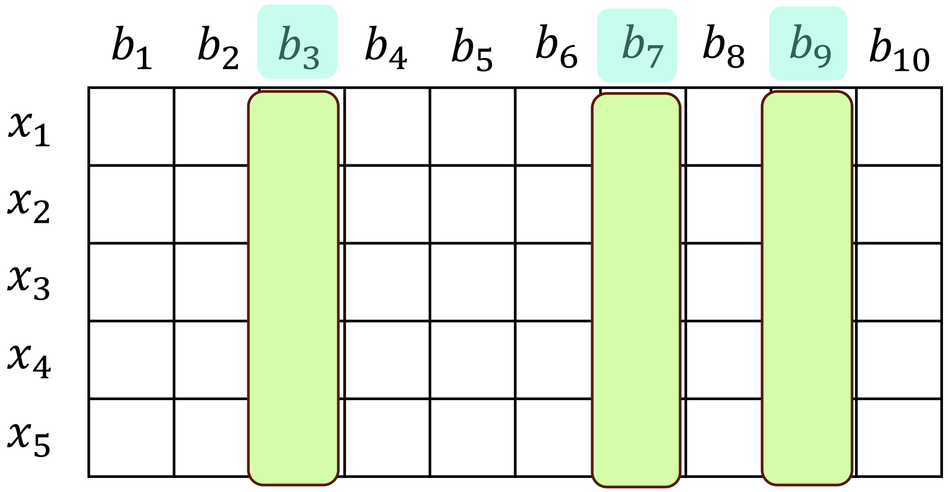
Alphabet: $\mathbb{F}^{2n}$



Proof Overview – Average Baby PIH

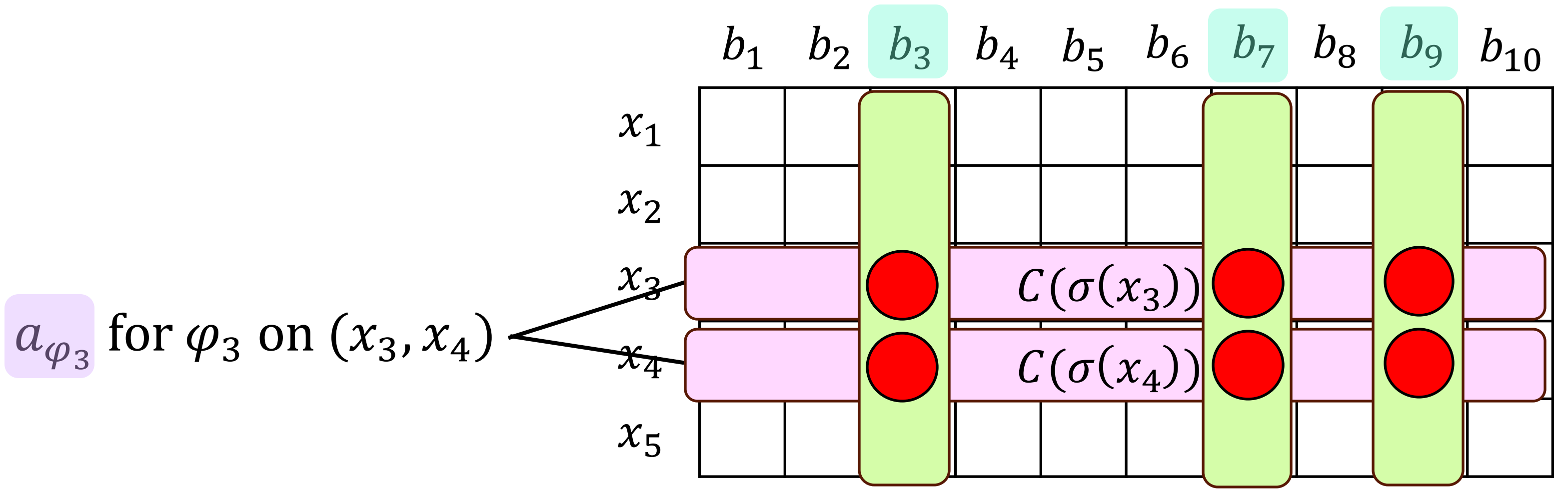
- Fix an error-correcting code $C\colon \Sigma \to \mathbb{F}^n$.
- Given a 2CSP instance $\Pi = (X, \Sigma, \Phi)$ that is either satisfiable or not (2r)-list satisfiable,
  - Variables: $\{a_\varphi | \forall \varphi \in \Phi\} \cup \{b_1, \ldots, b_n\}$

Alphabet: $\mathbb{F}^k$



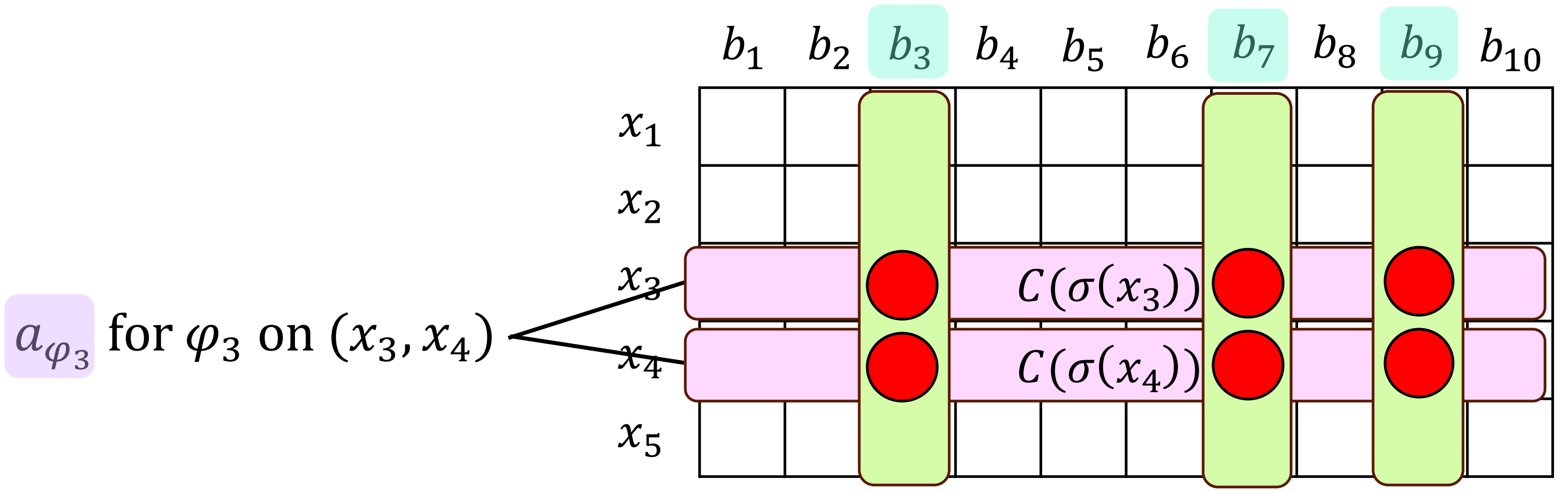
Proof Overview – Average Baby PIH

- Fix an error-correcting code $C: \Sigma \to \mathbb{F}^n$.
- Given a 2CSP instance $\Pi = (X, \Sigma, \Phi)$ that is either satisfiable or not (2r)-list satisfiable,
  - Variables: $\{a_\varphi | \forall \varphi \in \Phi\} \cup \{b_1, \dots, b_n\}$
  - Constraints: row-column consistency check



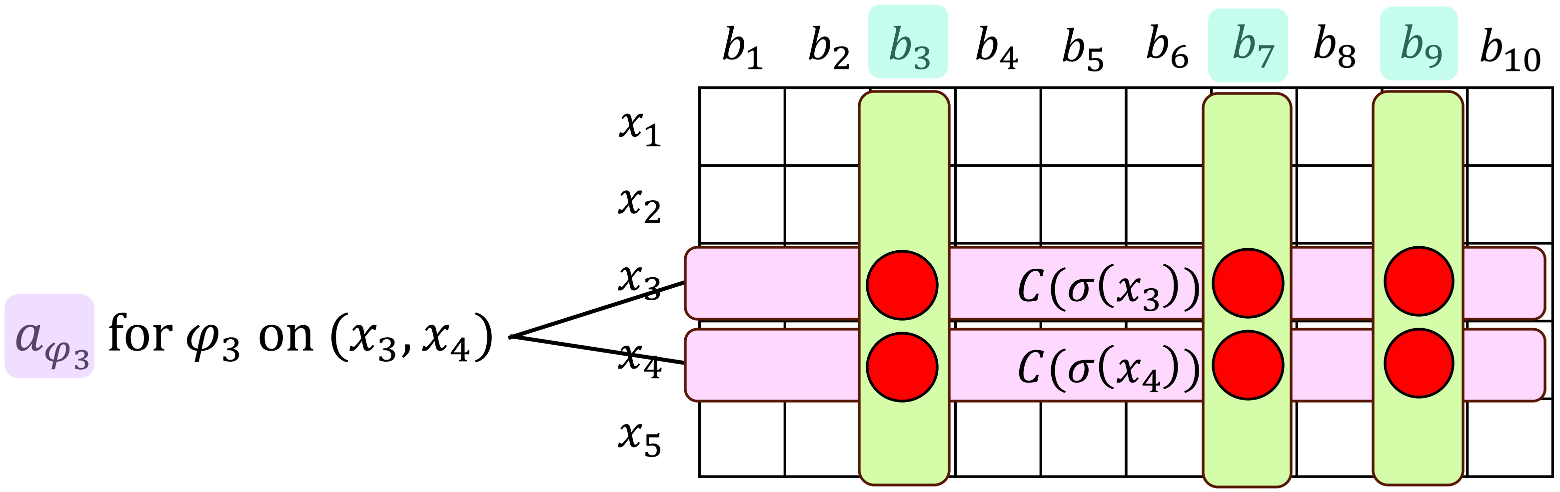
Proof Overview – Average Baby PIH

- Completeness is easy.
- Soundness idea: fix any average r-list satisfying assignment σ, argue:
  - either for $\geq (1-\varepsilon)$ fraction of b_j, $\sigma(b_j)$ contains $> r$ values;
  - or $\left|\bigcup_{\varphi\in\Phi} \sigma(a_\varphi)\right|$ is super large



Proof Overview – Average Baby PIH

- Completeness is easy.
- Soundness idea: fix any average r-list satisfying assignment σ, argue:
  - suppose for contradiction that, for $\geq \varepsilon n$ positions, $\sigma(b_j)$ contains $\leq r$ values;
  - $\Rightarrow \left|\bigcup_{\varphi\in\Phi} \sigma(a_\varphi)\right|$ has a collision on position j.



Takeaway

- Parameterized Inapproximability Hypothesis – parameterized analog of PCP
- Baby PIH – inapproximability of the list-satisfiability of (parameterized) 2CSP
  - W[1]-hard to distinguish between [1-list satisfiable] and [not even r-list satisfiable]
  - Proof Idea: induction on the list size

Takeaway

- Parameterized Inapproximability Hypothesis – parameterized analog of PCP
- Baby PIH – inapproximability of the list-satisfiability of (parameterized) 2CSP
  - W[1]-hard to distinguish between [1-list satisfiable] and [not even r-list satisfiable]
  - Proof Idea: induction on the list size
- Average Baby PIH
  - W[1]-hard to distinguish between [1-list satisfiable] and [not even *average* r-list satisfiable]
  - Proof Idea: using error-correcting codes with large collision numbers
  - Average Baby PIH ⇒ constant inapproximability of k-ExactCover, k-Nearest Codeword Problem

Takeaway

- Parameterized Inapproximability Hypothesis – parameterized analog of PCP
- Baby PIH – inapproximability of the list-satisfiability of (parameterized) 2CSP
  - W[1]-hard to distinguish between [1-list satisfiable] and [not even r-list satisfiable]
  - Proof Idea: induction on the list size
- Average Baby PIH
  - W[1]-hard to distinguish between [1-list satisfiable] and [not even *average* r-list satisfiable]
  - Proof Idea: using error-correcting codes with large collision numbers
  - Average Baby PIH $\Rightarrow$ constant inapproximability of k-ExactCover, k-Nearest Codeword Problem
- Thanks!