Hardness Magnification for all Sparse NP Languages

Lijie Chen
MIT

Ce Jin
Tsinghua U.

Ryan Williams
MIT
Minimum Circuit Size Problem

Problem: MCSP[$s(m)$]

- **Given:** $f: \{0,1\}^m \rightarrow \{0,1\}$ as a truth table of length $n = 2^m$
- **Decide:** Does $f$ have a circuit of size at most $s(m)$?
Problem: MCSP[$s(m)$]

- **Given:** $f: \{0,1\}^m \rightarrow \{0,1\}$ as a truth table of length $n = 2^m$
- **Decide:** Does $f$ have a circuit of size at most $s(m)$?

MCSP[$s(m)$] ∈ NP; solvable in $n \cdot 2^{\tilde{O}(s(m))}$ time.
Minimum Circuit Size Problem

**Problem:** MCSP\[s(m)\]

- **Given:** \( f: \{0,1\}^m \rightarrow \{0,1\} \) as a truth table of length \( n = 2^m \)
- **Decide:** Does \( f \) have a circuit of size at most \( s(m) \)?

MCSP\[s(m)\] \( \in \) NP; solvable in \( n \cdot 2^{\tilde{O}(s(m))} \) time.

We believe MCSP\[^{m10}\] \( \notin \) P/poly!
(otherwise, no strong PRGs exist [Razborov-Rudich])
Minimum Circuit Size Problem

Problem: MCSP[$s(m)$]

Given: $f: \{0,1\}^m \rightarrow \{0,1\}$ as a truth table of length $n = 2^m$

Decide: Does $f$ have a circuit of size at most $s(m)$?

MCSP[$s(m)$] $\in \text{NP}$; solvable in $n \cdot 2^{\tilde{O}(s(m))}$ time.

We believe MCSP[$m^{10}$] $\not\in \text{P/poly}$!
(otherwise, no strong PRGs exist [Razborov-Rudich])

If MCSP[$m^{10}$] doesn’t have circuits of $n \cdot \text{polylog } n$ size and polylog $n$ depth, then $\text{NP} \not\subset \text{P/poly}$.

(McKay-Murray-Williams’19)
Minimum Circuit Size Problem

**Problem:** MCSP\[s(m)\]
- **Given:** \(f : \{0,1\}^m \rightarrow \{0,1\}\) as a truth table of length \(n = 2^m\)
- **Decide:** Does \(f\) have a circuit of size at most \(s(m)\)?

MCSP\[s(m)\] ∈ NP; solvable in \(n \cdot 2^{\tilde{O}(s(m))}\) time.

We believe MCSP\[m^{10}\] ∉ P/poly!
(otherwise, no strong PRGs exist [Razborov-Rudich])

*If* MCSP\[m^{10}\] *doesn’t have* circuits of \(n \cdot \text{polylog } n\) size and polylog \(n\) depth, then NP ∉ P/poly.

(McKay-Murray-Williams’19)

“Hardness Magnification”
Hardness Magnification for MCSP

(Input length $n = 2^m$)

If MCSP[$m^{10}$] doesn’t have circuits of $n \cdot \text{polylog } n$ size and polylog $n$ depth, then $\text{NP} \not\subseteq \text{P/poly}$.

(McKay-Murray-Williams’19)

Similar magnification results for MKtP (Minimum time-bounded Kolmogorov Complexity, $Kt(x)$)
Hardness Magnification for MCSP

(Input length $n = 2^m$)

If $\text{MCSP}[m^{10}]$ doesn’t have circuits of $n \cdot \text{polylog } n$ size and polylog $n$ depth, then $\text{NP} \not\subset \text{P/poly}$.

(McKay-Murray-Williams’19)

Similar magnification results for $\text{MKtP}$ (Minimum time-bounded Kolmogorov Complexity, $Kt(x)$)

$Kt(x) =$ “measure of how much info needed to generate $x$ quickly”

$\text{MKtP} \approx \text{MCSP}$ with “EXP-oracle gates”
Hardness Magnification for MCSP

(Input length $n = 2^m$)

If $\text{MCSP}[m^{10}]$ doesn’t have circuits of $n \cdot \text{polylog } n$ size and polylog $n$ depth, then $\text{NP} \not\subseteq \text{P/poly}$.

(McKay-Murray-Williams’19)

Similar magnification results for MKtP (Minimum \emph{time-bounded Kolmogorov Complexity}, $\text{Kt}(x)$)

$\text{Kt}(x) =$ “measure of how much info needed to generate $x$ quickly”

$\text{MKtP} \approx \text{MCSP}$ with “EXP-oracle gates”

(“Gap-MKtP[$a, b$]”: distinguish between $\text{Kt}(x) \leq a$ and $\text{Kt}(x) \geq b$)

If $\text{Gap-MKtP}[m^{10}, m^{10} + O(m)]$ doesn’t have $n^3\text{polylog } n$-size (De Morgan) Formulas, then $\text{EXP} \not\subseteq \text{NC}^1$.

(Oliveira-Pich-Santhanam’19)
Hardness Magnification for MCSP

(Input length $n = 2^m$)

If MCSP$[m^{10}]$ doesn’t have circuits of $n \cdot \text{polylog } n$ size and polylog $n$ depth, then $\text{NP} \not\subset \text{P/poly}$.

(McKay-Murray-Williams’19)

Similar magnification results for MKtP (Minimum time-bounded Kolmogorov Complexity, $K_t(x)$)

$K_t(x)$ = “measure of how much info needed to generate $x$ quickly”

$\text{MKtP} \approx \text{MCSP}$ with “EXP-oracle gates”

(“Gap-MKtP”)

If Gap-MKtP$[m^{10}]$ doesn’t have circuits of $n \cdot \text{polylog } n$ size and polylog $n$ depth, then $\text{NP} \not\subset \text{P/poly}$.

(De Morgan...)

(Other Hardness Magnification Results)

$n^{1-\epsilon}$-approximate Clique [Sri’03]

Average-case MCSP [OS’18]

$k$-Vertex-Cover [OS’18]

low-depth circuit LBs for $\text{NC}^1$ [AK’10,CT’19]

sublinear-depth circuit LBs for $\text{P}$ [LW’13]

...
How to view Hardness Magnification?

Suggests new approaches to proving strong lower bounds?
How to view Hardness Magnification?

Suggests new approaches to proving strong lower bounds?

Weak LB

Magnification

Strong LB

It is argued that HM can bypass the Natural Proof Barrier [Razborov-Rudich]
It is argued that HM can bypass the **Natural Proof Barrier** [Razborov-Rudich]

- **A heuristic argument** [AK’10, OS’18]: HM seems to yield strong LBs only for *certain* functions, not for *most* of them (violating the “largeness” condition of Natural Proofs)

**Weak LB**

Magnification

**Strong LB**

Suggests new approaches to proving strong lower bounds?
It is argued that HM can bypass the Natural Proof Barrier [Razborov-Rudich]

• A heuristic argument [AK’10, OS’18]: HM seems to yield strong LBs only for certain functions, not for most of them (violating the “largeness” condition of Natural Proofs)

• A real theorem [CHOPRS to appear in ITCS’20]
  In some cases, the required weak LB actually implies the non-existence of natural proofs
Extending Known Lower Bounds?

(Input length $n = 2^m$)

If $\text{Gap-MKtP}[m^{10}, m^{10} + O(m)]$

· doesn’t have $n^3 \text{polylog } n$-size (De Morgan) Formulas, then $\text{EXP} \not\subset \text{NC}^1$.

[OPS’19]
Extending Known Lower Bounds?

(Input length $n = 2^m$)

If Gap-MKtP[$m^{10}, m^{10} + O(m)$]
  
  - doesn't have $n^3 \text{polylog } n$-size (De Morgan) Formulas, then $\text{EXP} \not\subset \text{NC}^1$.

We know how to prove $n^{1.99}$-size formula lower bound for Gap-MKtP! [OPS’19]
Extending Known Lower Bounds?

(Input length $n = 2^m$)

If Gap-MKtP[$m^{10}, m^{10} + O(m)$]
- doesn’t have $n^3 \text{polylog } n$-size (De Morgan) Formulas, then $\text{EXP} \not\subset \text{NC}^1$.

We know how to prove $n^{1.99}$-size formula lower bound for Gap-MKtP ! [OPS’19]

Can we improve it by a factor of $n^{1+\epsilon}$ ?
Extending Known Lower Bounds?

(Input length $n = 2^m$)

If $\text{Gap-MKtP}[m^{10}, m^{10} + O(m)]$
- doesn’t have $n^3 \text{polylog } n$-size (De Morgan) Formulas, then $\text{EXP} \not\subset \text{NC}^1$.
- doesn’t have $n \cdot \text{polylog } n$-size Formula-$\oplus$, then $\text{EXP} \not\subset \text{NC}^1$.

[OPS’19]

Formula-$\oplus$: De Morgan Formulas where each leaf node computes XOR of a subset of input bits
Extending Known Lower Bounds?

(Input length $n = 2^m$)

If Gap-MKtP$[m^{10}, m^{10} + O(m)]$

- doesn’t have $n^3 \text{polylog } n$-size (De Morgan) Formulas, then $\text{EXP} \not\subset \text{NC}^1$.
- doesn’t have $n \cdot \text{polylog } n$-size Formula-$\oplus$, then $\text{EXP} \not\subset \text{NC}^1$.

Known LB against Formula-$\oplus$(Tal’16):

$F_2$-Inner-Product $\not\in$ Formula-$\oplus$ $[n^2 / \text{polylog } n]$
Extending Known Lower Bounds?

(Input length $n = 2^m$)

If $\text{Gap-MKtP}[m^{10}, m^{10} + O(m)]$

* doesn’t have $n^3\text{polylog } n$-size (De Morgan) Formulas, then $\text{EXP} \not\subset \text{NC}^1$.
* doesn’t have $n \cdot \text{polylog } n$-size Formula-$\oplus$, then $\text{EXP} \not\subset \text{NC}^1$.

Known LB against Formula-$\oplus$(Tal’16):

$F_2$-Inner-Product $\not\in$ Formula-$\oplus$ $[n^2 / \text{polylog } n]$

Much easier than Gap-MKtP??!

Stronger LB than required
Extending Known Lower Bounds?

(Input length $n = 2^m$)

If Gap-MKtP[$m^{10}, m^{10} + O(m)$]
  - doesn’t have $n^3\text{polylog }n$-size (De Morgan) Formulas, then $\text{EXP} \not\subset \text{NC}^1$.
  - doesn’t have $n \cdot \text{polylog }n$-size Formula-$\oplus$, then $\text{EXP} \not\subset \text{NC}^1$.

Known LB against Formula-$\oplus$(Tal’16):
$F_2$-Inner-Product $\not\in \text{Formula-}\oplus [n^2 / \text{polylog }n]$

Much easier than Gap-MKtP??!

Stronger LB than required

*Can we adapt the proof techniques to Gap-MKtP?*
How to view Hardness Magnification?

- Indicates proving “weak” lower bounds are even harder than previously thought??
- Suggests new approaches to proving strong lower bounds?
How to view Hardness Magnification?

Suggests new approaches to proving strong lower bounds?

Indicates proving “weak” lower bounds are even harder than previously thought??

Weak LB

Magnification

Strong LB

• Hardness magnification:

Proving *almost-linear size lower bounds* is already as hard as proving *super-polynomial lower bounds*…
What is special about MCSP and MKtP?
Is it because they are “compression” problems?

Problem: MCSP[$s(m)$]
- **Given:** $f: \{0,1\}^m \rightarrow \{0,1\}$ as a truth table of length $n = 2^m$
- **Decide:** Does $f$ have a circuit of size at most $s(m)$?
(Input length $n = 2^m$)

If MCSP$[m^{10}]$ doesn’t have circuits of $n \cdot \text{polylog } n$ size and polylog $n$ depth, then NP $\not\subset$ P/poly.

If Gap-MKtP$[m^{10}, m^{10} + O(m)]$
  - doesn’t have $n^3 \text{polylog } n$-size (De Morgan) Formula, then EXP $\not\subset$ NC$^1$.
  - doesn’t have $n \cdot \text{polylog } n$-size Formula-$\oplus$, then EXP $\not\subset$ NC$^1$.

What is special about MCSP and MKtP?
Is it because they are “compression” problems?

Observation: MCSP$[m^{10}]$ and MKtP$[m^{10}]$ are sparse languages!

MCSP$s(m)$ is $2^{\tilde{O}(s(m))}$-sparse; there are at most $2^{\tilde{O}(s(m))}$ many circuits!

Problem: MCSP$[s(m)]$
  - Given: $f: \{0,1\}^m \rightarrow \{0,1\}$ as a truth table of length $n = 2^m$
  - Decide: Does $f$ have a circuit of size at most $s(m)$?
Our result: Hardness magnification holds for all sparse \( \text{NP} \) languages!

If \( \text{MCSP}[m^{10}] \) doesn’t have circuits of \( n \cdot \text{polylog } n \) size and \( \text{polylog } n \) depth, then \( \text{NP} \not\subset \text{P/poly} \).

If \( \text{Gap-MKtP}[m^{10}, m^{10} + O(m)] \)
- doesn’t have \( n^3 \text{polylog } n \)-size (De Morgan) Formula, then \( \text{EXP} \not\subset \text{NC}^1 \).
- doesn’t have \( n \cdot \text{polylog } n \)-size Formula-\( \oplus \), then \( \text{EXP} \not\subset \text{NC}^1 \).

What is special about \( \text{MCSP} \) and \( \text{MKtP} \)?
Is it because they are “compression” problems?

Observation: \( \text{MCSP}[m^{10}] \) and \( \text{MKtP}[m^{10}] \) are \textit{sparse} languages!

\( \text{MCSP}[s(m)] \) is \( 2^{\tilde{O}(s(m))} \)-sparse;
there are at most \( 2^{\tilde{O}(s(m))} \) many circuits!

Our result: Hardness magnification holds for \textit{all} sparse \( \text{NP} \) languages!
HM for all sparse NP languages

Theorem 1:

Let $L$ be any $2^{n^{o(1)}}$-sparse NP language.

- If $L$ doesn’t have $n^{1.01}$-size circuits, then for all $k$, $\text{NP} \not\subset \text{SIZE}[n^k]$. 
HM for all sparse NP languages

Theorem 1:

Let $L$ be any $2^{n^{o(1)}}$-sparse NP language.

- If $L$ doesn't have $n^{1.01}$-size circuits, then for all $k$, $\text{NP} \not\subset \text{SIZE}[n^k]$.
- If $L$ doesn't have $n^{3.01}$-size formulas, then for all $k$, NP doesn't have $n^k$-size formulas.
- If $L$ doesn't have $n^{2.01}$-size branching programs, then for all $k$, NP doesn't have $n^k$-size branching programs.

Similar results for other models!
HM for all sparse NP languages

Theorem 1:

Let $L$ be any $2^{n^{o(1)}}$-sparse NP language.

- If $L$ doesn’t have $n^{1.01}$-size circuits, then for all $k$, $\text{NP} \not\subset \text{SIZE}[n^k]$.
- If $L$ doesn’t have $n^{3.01}$-size formulas, then for all $k$, $\text{NP}$ doesn’t have $n^k$-size formulas.
- If $L$ doesn’t have $n^{2.01}$-size branching programs, then for all $k$, $\text{NP}$ doesn’t have $n^k$-size branching programs.

Similar results for other models!

Compared with [MMW’19]: Our techniques yield weaker consequences (e.g. they get $\text{NP} \not\subset \text{P}/\text{poly}$), but apply to more restricted models.
HM for all sparse NP languages

Theorem 1:

Let $L$ be any $2^{n^{o(1)}}$-sparse NP language.

- If $L$ doesn’t have $n^{1.01}$-size circuits, then for all $k$, $\mathsf{NP} \not\subseteq \mathsf{SIZE}[n^k]$.
- If $L$ doesn’t have $n^{3.01}$-size formulas, then for all $k$, $\mathsf{NP}$ doesn’t have $n^k$-size formulas.
- If $L$ doesn’t have $n^{2.01}$-size branching programs, then for all $k$, $\mathsf{NP}$ doesn’t have $n^k$-size branching programs.

Similar results for other models!

Compared with [MMW’19]: Our techniques yield weaker consequences (e.g. they get $\mathsf{NP} \not\subseteq \mathsf{P/poly}$), but apply to more restricted models.

(Best known formula LB: $n^3/\text{polylog } n$) [Håstad 90s, Tal]
(Best known branching program LB: $n^2/\text{polylog } n$) [Nečiporuk 60s]
Hardness Magnification for MCSP

(Input length $n = 2^m$)

Theorem 2:

If MCSP[$m^{10}$] doesn’t have $n^3 \text{polylog } n$-size (De Morgan) Formulas, then $\text{PSPACE} \not\subset (\text{nonuniform}) \text{NC}^1$.

Similar results for other models!
Theorem 2:

If $\text{MCSP}[m^{10}]$ doesn’t have $n^3 \text{polylog } n$-size (De Morgan) Formulas, then $\text{PSPACE} \not\subseteq (\text{nonuniform}) \text{NC}^1$.

Similar results for other models!

Best $\text{MCSP}$ lower bound (Cheraghchi-Kabanets-Lu-Myrisiotis’19):

$\text{MCSP}[2^m/10m]$ requires $n^{3-o(1)}$-size formulas.

(doesn’t work for $m^{10}$ …)
Hardness Magnification for MCSP

(Input length $n = 2^m$)

**Theorem 2:**

If $\text{MCSP}[m^{10}]$ doesn’t have $n^3\text{polylog } n$-size (De Morgan) Formulas, then $\text{PSPACE} \not\subset (\text{nonuniform}) \text{NC}^1$.

Similar results for other models!

Best MCSP lower bound (Cheraghchi-Kabanets-Lu-Myrisiotis’19) :

$\text{MCSP}[2^m/10m]$ requires $n^{3-o(1)}$-size formulas.

(doesn’t work for $m^{10}$ …)

Similar results for $\text{MKtP}[m^{10}]$ and $\text{EXP} \not\subset \text{NC}^1$ (improving upon [OPS’19] which required lower bounds for Gap-MKtP)
Algorithms with small non-uniformity

Theorem 3:

Let $L$ be a $2^{n^{o(1)}}$-sparse NP language not computable by an $n^{1.01}$-time $n^{0.01}$-space deterministic algorithm with $n^{0.01}$ bits of advice, then NP $\not\subset$ SIZE$[n^k]$ for all $k$. 


**Theorem 3:**

Let $L$ be a $2^{n^o(1)}$-sparse NP language not computable by an $n^{1.01}$-time $n^{0.01}$-space deterministic algorithm with $n^{0.01}$ bits of advice, then $\text{NP} \not\subset \text{SIZE}[n^k]$ for all $k$.

The hypothesis is “close” to what we can prove!

There is a $(2^{n^{0.01}} \cdot n)$-sparse language $L \in \text{DTIME}[\tilde{O}(n^{1.01})]$, not computable by an $n^{1.01}$-time deterministic algorithm with $n^{0.01}$ bits of advice.

(Adaptation of time hierarchy theorem)
Theorem 3:

Let \( L \) be a \( 2^{n^{o(1)}} \)-sparse \( \text{NP} \) language not computable by an \( n^{1.01} \)-time \( n^{0.01} \)-space deterministic algorithm with \( n^{0.01} \) bits of advice, then \( \text{NP} \not\subset \text{SIZE}[n^k] \) for all \( k \).

The hypothesis is “close” to what we can prove!

There is a \( \left(2^{n^{0.01}} \cdot n\right) \)-sparse language \( L \in \text{DTIME}[\tilde{O}(n^{1.01})] \), not computable by an \( n^{1.01} \)-time deterministic algorithm with \( n^{0.01} \) bits of advice. (Adaptation of time hierarchy theorem)

Can we make it sparser?
Proof of Theorem 1.2

Let $L$ be any $2^{n^{o(1)}}$-sparse NP language.

- If $L$ doesn’t have $n^{3.01}$-size formulas, then for every $k$, NP doesn’t have $n^k$-size formulas.
Proof of Theorem 1.2

Assume: $\mathbf{NP}$ has $n^k$-size formulas for some $k$.

Goal: Design $n^{3.01}$-size formulas for $2^{n^{o(1)}}$-sparse $\mathbf{NP}$ language $L$. 
Assume: \( \textbf{NP} \) has \( n^k \)-size formulas for some \( k \).

Goal: Design \( n^{3.01} \)-size formulas for \( 2^{n^{o(1)}} \)-sparse \( \textbf{NP} \) language \( L \).

(Sparse) \( L \cap \{0,1\}^n \)
Intuition

Assume: \( \text{NP} \) has \( n^k \)-size formulas for some \( k \).

Goal: Design \( n^{3.01} \)-size formulas for \( 2^{n^{o(1)}} \)-sparse \( \text{NP} \) language \( L \).

(Sparse) \( L \cap \{0,1\}^n \)

(Dense) Auxiliary \( \text{NP} \) language \( K \cap \{0,1\}^{n^{0.001/k}} \) ("kernel problem")
Intuition

Assume: \( \textbf{NP} \) has \( n^k \)-size formulas for some \( k \).

Goal: Design \( n^{3.01} \)-size formulas for \( 2^{n^{o(1)}} \)-sparse \( \textbf{NP} \) language \( L \).

We will construct cubic-size formulas for \( L \), with oracle access to \( K \).

(Dense) Auxiliary \( \textbf{NP} \) language \( K \cap \{0,1\}^{n^{0.001/k}} \) (“kernel problem”)

(Sparse) \( L \cap \{0,1\}^n \)
Proof of Theorem 1.2

Assume: NP has $n^k$-size formulas for some $k$.

Goal: Design $n^{3.01}$-size formulas for $2^{n^{o(1)}}$-sparse NP language $L$.

Set $t := n^{0.001/k} > \log (\text{Sparsity of } L)$.

Standard hashing tricks imply:
There is a hash function $H_S : \{0,1\}^n \rightarrow \{0,1\}^{O(t)}$ that is

- Perfect: maps YES-instances of $L$ into distinct images
- described by an $O(t)$-bit seed $s$
- linear over $F_2$

(there is a “correct” seed $s$ that makes the hash function $H_S$ perfect)
Proof of Theorem 1.2

Assume: NP has $n^k$-size formulas for some $k$.

Goal: Design $n^{3.01}$-size formulas for $2^{n^{o(1)}}$-sparse NP language $L$.

Set $t := n^{0.001/k} > \log \text{ (Sparsity of } L \text{)}$.

Standard hashing tricks imply:
There is a hash function $H_s: \{0,1\}^n \rightarrow \{0,1\}^{O(t)}$ that is
- Perfect: maps YES-instances of $L$ into distinct images
- described by an $O(t)$-bit seed $s$
- linear over $F_2$

(Construction: pick some coordinates from the Error Correcting Code)
Proof of Theorem 1.2

Assume: \( \mathbf{NP} \) has \( n^k \)-size formulas for some \( k \).

Goal: Design \( n^{3.01} \)-size formulas for \( 2^{n^{O(1)}} \)-sparse \( \mathbf{NP} \) language \( L \).

\[
( t := n^{0.001/k} > \log \text{(Sparsity of } L) )
\]

(Perfect hash \( H_s: \{0,1\}^n \to \{0,1\}^{O(t)} \) with seed \( |s| = O(t) \) )

Define an \( O(t) \)-input auxiliary \( \mathbf{NP} \) problem \( K \) (“kernel problem”):

**Input:** Hash seed \( s \), hash value \( h \), index \( i \in [n] \)

**Output:** The \( i \)-th bit of some \( x \in L \) such that \( H_s(x) = h \).

For the “correct” \( s \), this \( x \) is unique.

\( \text{(Perfect hash } H_s: \{0,1\}^n \to \{0,1\}^{O(t)} \text{ with seed } |s| = O(t) \text{ )} \)

Define an \( O(t) \)-input auxiliary \( \mathbf{NP} \) problem \( K \) (“kernel problem”):

**Input:** Hash seed \( s \), hash value \( h \), index \( i \in [n] \)

**Output:** The \( i \)-th bit of some \( x \in L \) such that \( H_s(x) = h \).

For the “correct” \( s \), this \( x \) is unique.
Proof of Theorem 1.2

Assume: \textbf{NP} has $n^k$-size formulas for some $k$.  

Goal: Design $n^{3.01}$-size formulas for $2^{n^{o(1)}}$-sparse \textbf{NP} language $L$.

\[(t \coloneqq n^{0.001/k} > \log (\text{Sparsity of } L))\]

(Perfect hash $H_s:\{0,1\}^n \rightarrow \{0,1\}^{O(t)}$ with seed $|s| = O(t)$ )

Define an $O(t)$-input auxiliary \textbf{NP} problem $K$ ("kernel problem"):

\begin{itemize}
  \item \textbf{Input}: Hash seed $s$, hash value $h$, index $i \in [n]$  
  \item \textbf{Output}: The $i$-th bit of \textbf{some} $x \in L$ such that $H_s(x) = h$.
\end{itemize}

\textbf{NP} has $n^k$-size formulas $\Rightarrow K$ has formulas of size $n^{0.001}$!
Proof of Theorem 1.2

Assume: \textsf{NP} has \( n^k \)-size formulas for some \( k \).

Goal: Design \( n^{3.01} \)-size formulas for \( 2^{n^{o(1)}} \)-sparse \textsf{NP} language \( L \).

\( (t := n^{0.001/k} > \log (\text{Sparsity of } L)) \)

(Perfect hash \( H_s: \{0,1\}^n \to \{0,1\}^{O(t)} \) with seed \( |s| = O(t) \) )

Define an \( O(t) \)-input auxiliary \textsf{NP} problem \( K \) ("kernel problem"):

**Input:** Hash seed \( s \), hash value \( h \), index \( i \in [n] \)

**Output:** The \( i \)-th bit of some \( x \in L \) such that \( H_s(x) = h \).

\textsf{NP} has \( n^k \)-size formulas \( \Rightarrow K \) has formulas of size \( n^{0.001} \)!

On input \((s, h, i)\), guess \((x, y)\), where \( y \) witnesses \( x \in L \).

Accept \( \iff x_i = 1 \) and \( H_s(x) = h \).
Proof of Theorem 1.2

Assume: \( \text{NP} \) has \( n^k \)-size formulas for some \( k \).

Goal: Design \( n^{3.01} \)-size formulas for \( 2^{n^{o(1)}} \)-sparse \( \text{NP} \) language \( L \).

\[
(t := n^{0.001/k} > \log \text{(Sparsity of } L))
\]

(Perfect hash \( H_s: \{0,1\}^n \rightarrow \{0,1\}^{O(t)} \) with seed \( |s| = O(t) \))

Define an \( O(t) \)-input auxiliary \( \text{NP} \) problem \( K \) (“kernel problem”):

**Input:** Hash seed \( s \), hash value \( h \), index \( i \in [n] \)

**Output:** The \( i \)-th bit of \textit{some} \( x \in L \) such that \( H_s(x) = h \).

Claim: for the “correct” \( s \), the following decides \( L \):

On input \( x \in \{0,1\}^n \), accept iff:

\[
\forall i \in [n], K(s, H_s(x), i) = x_i
\]
On input $x \in \{0,1\}^n$, accept iff:
\[
\forall i \in [n], K(s, H_s(x), i) = x_i
\]

**Goal:** Design $n^{3.01}$-size formulas for $2^{n^{o(1)}}$-sparse NP language $L$. 

$K: n^{0.001}$-size
Goal: Design $n^{3.01}$-size formulas for $2^{n^{o(1)}}$-sparse NP language $L$.

On input $x \in \{0,1\}^n$, accept iff:

$$\forall i \in [n], K(s, H_s(x), i) = x_i$$
On input $x \in \{0,1\}^n$, accept iff:

$$\forall i \in [n], K(s, H_s(x), i) = x_i$$

**Goal**: Design $n^{3.01}$-size formulas for $2^{n^{o(1)}}$-sparse NP language $L$. 

Each bit of $H_s(x)$ is an XOR function (implemented by De Morgan formulas of size $O(n^2)$).

Hash seed $s$ hardwired into formulas

$K$: $n^{0.001}$-size

$K$: $n^{0.001}$-size
On input $x \in \{0, 1\}^n$, accept iff:
\[ \forall i \in [n], K(s, H_s(x), i) = x_i \]

Goal: Design $n^{3.01}$-size formulas for $2^{n^{O(1)}}$-sparse NP language $L$. 

Hash seed $s$ hardwired into formulas

Each bit of $H_s(x)$ is an XOR function (implemented by De Morgan formulas of size $O(n^2)$)

Total size $n \cdot n^{0.001} \cdot O(n^2)$

$K: n^{0.001}$-size
Open Problems

• Are there any other natural sparse NP languages for which one can prove some concrete lower bounds?
Open Problems

• Are there any other natural sparse NP languages for which one can prove some concrete lower bounds?

• Is it possible to show hardness magnification results for “denser” variants of MCSP or MKtP, such as MCSP[$2^m/m^3$]?
Open Problems

• Are there any other natural sparse NP languages for which one can prove some concrete lower bounds?

• Is it possible to show hardness magnification results for “denser” variants of MCSP or MKtP, such as MCSP[2^m/m^3]?

Thank you!