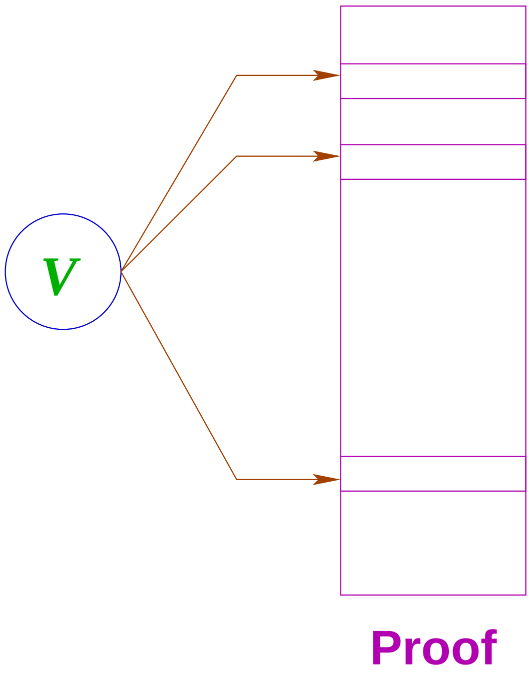
Small PCPs with Low Query Complexity

Prahladh Harsha and Madhu Sudan

Massachusetts Institute of Technology
Cambridge, U.S.A

Probabilistically Checkable Proofs

PCP's - Format of writing proofs and probabilistically validating the proof such that verifier checks only constant number of bits of proof.



New proof only polynomially larger than the original proof.

PCP's - **Definition**

$\text{PCP}_s[r, q]$ — class of languages L for which there exists a probabilistic polynomial time verifier V such that for any input x,

- V tosses at most r random coins

- V queries at most q bits of the proof and

  - if $x \in L$, then $\exists$ proof Π, $\Pr[V(x, \Pi) \text{ accepts}] = 1$
  - if $x \notin L$, then $\forall$ proof Π, $\Pr[V(x, \Pi) \text{ accepts}] < s$

    s - soundness.

The PCP Theorem and known results

The celebrated PCP Theorem [ALMSS-98] states that

$$\mathsf{NP} = \mathsf{PCP}_{\frac{1}{2}}[O(\log n), q]$$

for some constant q.

Parameters randomness and query complexity have been optimized individually since then.

- Randomness - [PS-94]
- Query Complexity - [Hås-97]

Small PCP's

[PS-94] Best result in terms of proof size.

Any NP Proof $\longrightarrow$ PCP Proof with

- Blowup of at most $n^{1+\epsilon}$ for any $\epsilon > 0$.

- constant number of queries.

Formally,

$$\mathrm{SAT} \in \mathrm{PCP}_{\frac{1}{2}}[(1+\epsilon)\log n, O(1)] \text{ for all } \epsilon > 0$$

However,

- Number of queries $= O(\frac{1}{\epsilon^2})$ with constant hidden in big-Oh $\approx 10^6$.

Note: To bound proof size, sufficient to bound randomness (proof size $\leq 2^{q+r}$).

Low Query PCP's

[Hås-97] Best result in terms of query complexity

$$\mathsf{NP} = \mathsf{PCP}_{\frac{1}{2}+\delta}[O(\log n), 3]$$

(3 Queries is optimal since $\mathsf{P} = \mathsf{PCP}_{\frac{1}{2}}[O(\log n), 2]$)

However,

- Blowup in proof size $\geq n^{10^6}$.
  (i.e., constant hidden in big-Oh of $O(\log n) \geq 10^6$.)

Small PCP's with Low Query Complexity?

"Nearly-optimal" results in proof-size and query complexity do not yield simultaneous optimality in both parameters.

Is simultaneous optimality possible or is the inefficiency inherent?

We examine size and query complexity jointly.

Previous Work:
[FS-95] claim (wasn't rigorously proved)

NP has PCP proofs with

- at most a slightly-super-quadratic blowup in proof size.

- query complexity of 165 bits.

Main Result

NP has PCP proofs with

- at most a slightly-super-cubic blowup in proof size.

- query complexity of 16 bits.

- soundness arbitrarily close to $\frac{1}{2}$.

Main Theorem: For all $\mu, \epsilon > 0$,

$$\mathrm{SAT} \in \mathsf{PCP}_{\frac{1}{2}+\mu}[(3+\epsilon)\log n, 16]$$

- Randomness of $(3+\epsilon)\log n$ implies proof size of at most $n^{3+\epsilon}$.

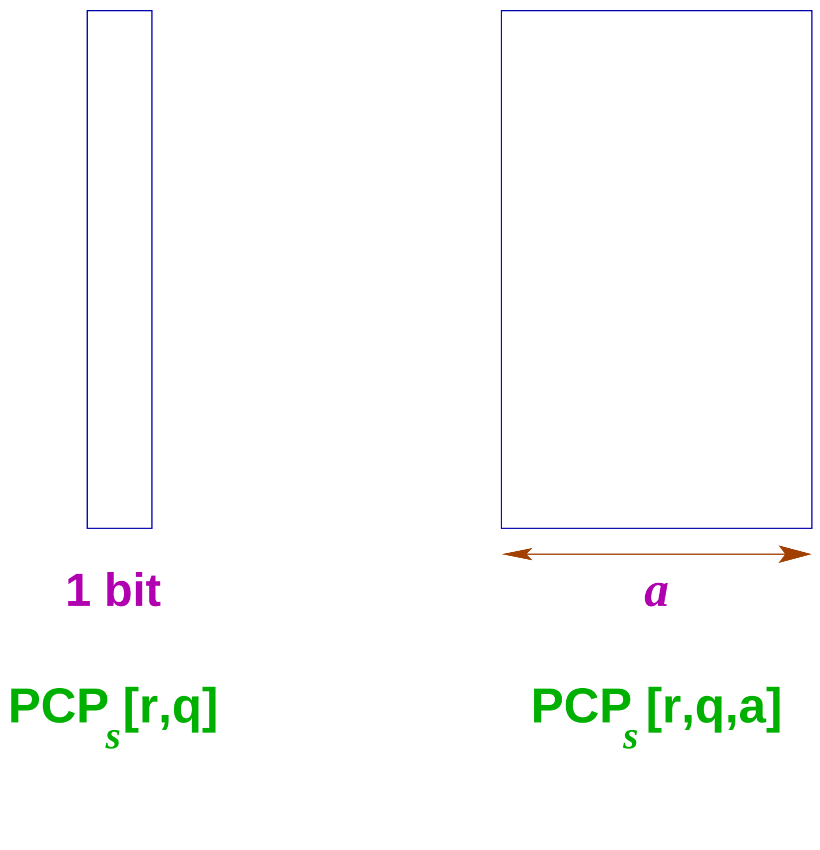
- Result for SAT extends easily [Cook-88] to $\mathrm{NTIME}(O(t(n)) \subseteq \mathsf{PCP}_{\frac{1}{2}+\mu}[(3+\epsilon)\log t(n), 16]$

Proof Techniques

Closer Look at PCP**'s**

Answer Size:

- So far we considered answer size of just 1 bit.
- Answer size could be potentially much larger, governed by another parameter, a.



- The corresponding PCP class is called $\text{PCP}_s[r, q, a]$

Parameters of PCP's

- RANDOMNESS: r - maximum number of random coins tossed by verifier.
- QUERY COMPLEXITY: q - number of queries issued by verifier.
- ANSWER SIZE: a - maximum answer size across all queries.
- SOUNDNESS: s - maximum probability of accepting a "false" proof.

PROOF SIZE is another interesting parameter. Proof size is at most $a2^{r+q}$. Proof size is implicitly governed by randomness, query complexity and answer size.

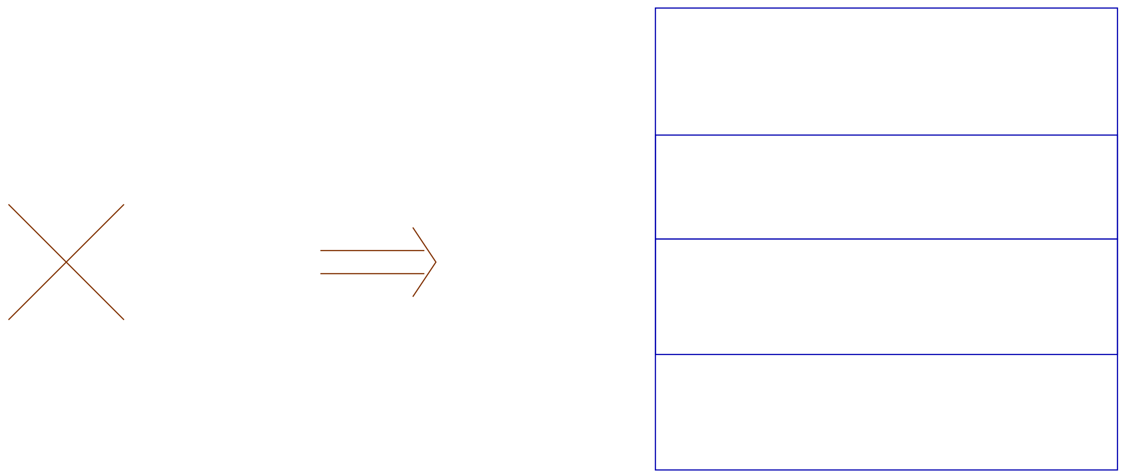
Proof Composition

Outer Verifier

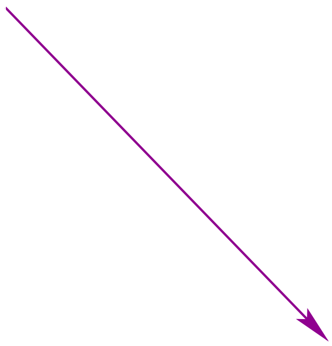
low randomness
low query complexity
large answer size
negligible soundness

large randomness
low query complexity
small answer size
small soundness

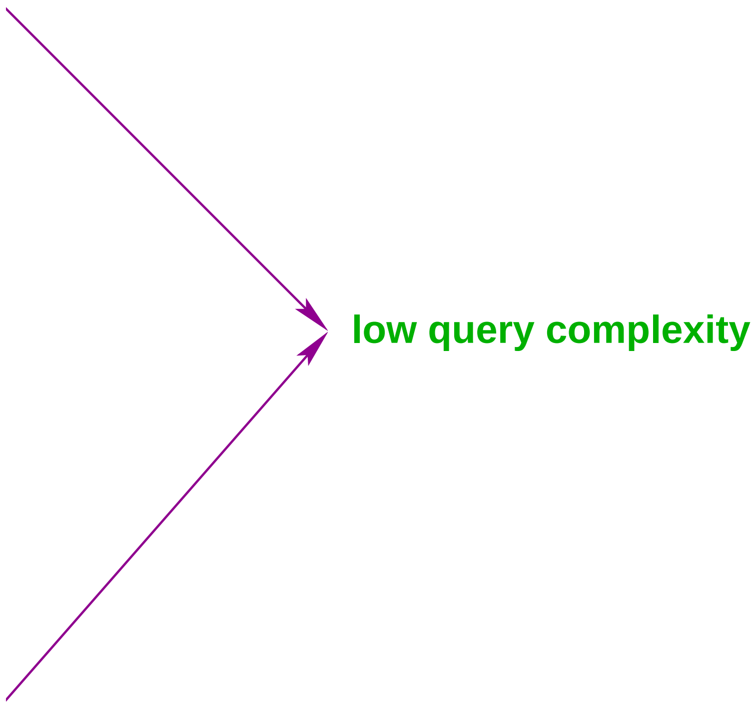
Inner Verifier



Composed Verifier



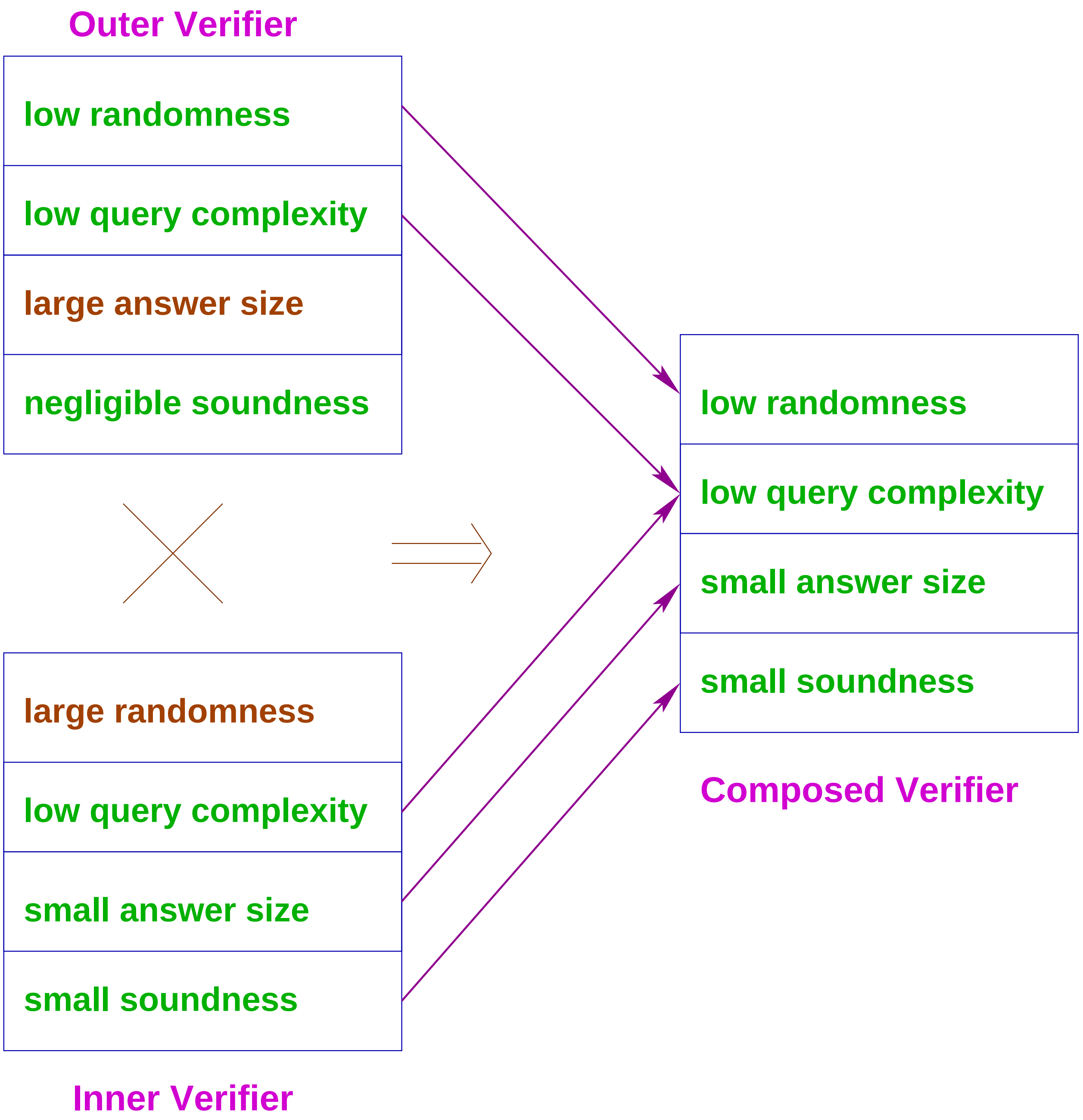
low randomness


low query complexity

small answer size

small soundness

Proof Composition



Why existing methods don't work?

- Small PCP's of [PS-94] have nearly optimal proof size at the cost of query complexity since
  - Outer Verifier – large query complexity
- Low Query PCP's of [Hås-97] have optimal query complexity at the cost of proof size since
  - Outer Verifier – large randomness

Thus, none of existing verifiers suffice for our cause.

What do we do?

Construct PCP's from scratch!

Obtain outer and inner verifiers without losing much on either randomness or query complexity.

Our construction revisits the construction of [ALMSS-98] using the improvements in

- analyses of Low-Degree Tests [RS-97]

- construction of inner verifiers [Hås-97].

Sketch of Construction

1. Construct Outer Verifier for SAT with good randomness and query complexity but large answer size. $(\mathrm{poly}\log)$

2. Reduce Answer Size from $\mathrm{poly}\log$ to $\mathrm{poly}\log\log\log$ by composing with PCP of [AS-97].

3. Construct Inner Verifier to reduce answer size finally to a bit. (similar to Håstad's inner verifier)

Step 2 necessary since Step 3 involves a 2^{2^a}-blowup in proof size (a - answer size).

Randomness Efficient PCP for SAT

Outermost verifier in our PCP is one efficient in randomness.

Construction of this PCP proceeds in 2 steps.

1. Transform SAT to a algebraic constraint problem, Polynomial Constraint Satisfaction (PCS)

2. Construct PCP for PCS using the Low-Degree test.

- Step 1 uses the results of [PS-94] and [LFKN-90] to provide a hardness result for PCS.

- Step 2 uses the Low-Degree Tests of [RS-97] along with the hardness result of PCS in Step 1 to obtain a PCP for SAT. This step is similar to the parallelization step in [ALMSS-98].

Polynomial Constraint Satisfaction (PCS)

- A neat algebraic constraint satisfaction problem.
- Easily amenable to construction of PCP's.

Hardness Result:

> Lemma: SAT reduces to PCS with at most a $n^{1+\epsilon}$ blowup in proof size.

Techniques used

- results from [PS-94].

- the "sum-check protocol" of [LFKN-90]

Most of the steps in the proof of result are implicit in literature, we provide a clean abstraction.

Low-Degree Tests (LDT)

Low-Degree Tests — procedures to test whether an arbitrary function $f : \mathbb{F}^m \to \mathbb{F}$ is close to some (unknown) polynomial p of low degree.

Use the Low-Degree Test of [RS-97] with hardness result of PCS to obtain

Lemma: For all $\mu, \epsilon > 0$,

$$\mathrm{SAT} \in \mathsf{PCP}_{\mu}[(3+\epsilon)\log n, 3, \mathrm{poly}\log n]$$

Steps in proof

- results from low-error analysis of Low-Degree test of [RS-97].
- Variant of parallelization step of [ALMSS-98] (higher dimensional varieties instead of 1-dimensional curves.)

Reducing Answer Size

Use result from [AS-97] to reduce the answer size substantially at the cost of a few more queries.

Composing with this result, we get

- answer size: $\text{poly} \log n \longrightarrow \text{poly} \log \log \log n$
- query complexity: $3 \longrightarrow 9$

Lemma: For all $\mu, \epsilon > 0$,

$$\text{SAT} \in \mathsf{PCP}_{\mu}[(3+\epsilon)\log n, 9, \text{poly} \log \log \log n]$$

Reducing answer size to a bit - Innermost verifier

Standard to use Håstad's inner verifier at this stage to reduce answer size to a bit.

However, Håstad's verifier cannot be directly composed with either

- PCP's with greater than 2 queries

  or

- "non-canonical" PCP's (of 2 or more queries).

And the PCP, we have constructed so far is both "non-canonical" and a 9-query one.

We give an alternate inner verifier which however takes 7 extra queries to reduce the answer size to a bit.

Finally,

- answer size: $\text{poly} \log \log \log \longrightarrow 1 \text{ bit}$

- query complexity: $9 \longrightarrow 16$

Summarizing...

Proved that SAT has a PCP verifier with

- proof size of at most $n^{3+\epsilon}$.

- query complexity of 16.

- soundness arbitrarily close to $\frac{1}{2}$.

Open Questions

- Do nearly-linear sized (or even linear !) PCP's with a query complexity of 3 exist?

- Any non-trivial limitations on joint size + query complexity of PCP's.

- We also pose a few technical questions that could potentially improve the size+query complexity of PCP's.