Tighter Circuit Lower Bounds for MA/1 With Efficient PCPs

Based on a Joint Work of
Joshua Cook and Dana Moshkovitz

Main Result

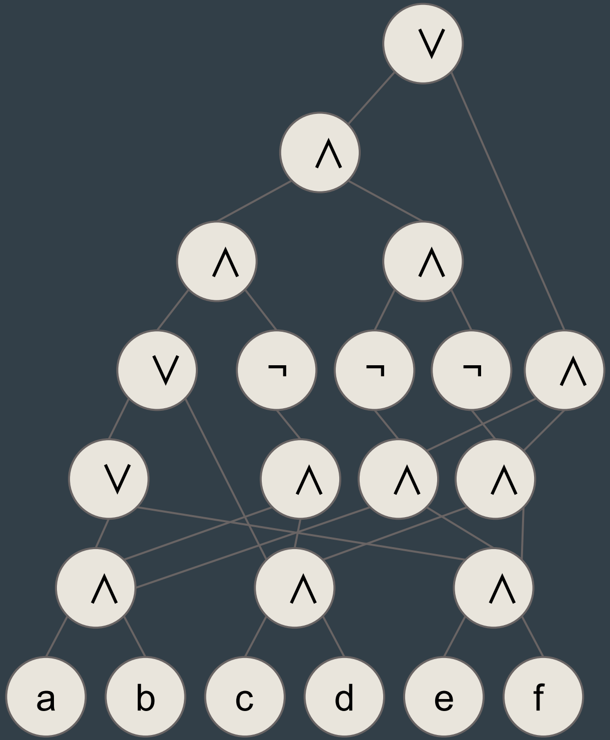
$\exists$ $a > 1$ and $g(n) = o(1)$ such that $\forall$ $k < a$

$$\mathrm{MATIME}[n^{k+g(n)}]/1 \not\subseteq \mathrm{SIZE}[O(n^k)]$$

- Super linear circuit lower bound.
- MA is similar to NP.
- Tighter parameters than previous results.

Explaining Our Result

Circuit Definition



Circuits have NOT, AND, OR gates, fan in at most 2.

SIZE[f(n)] are languages computable by families of circuits with f(n) gates.

Non uniform, circuits may be hard to find.

Uniform vs Non-Uniform

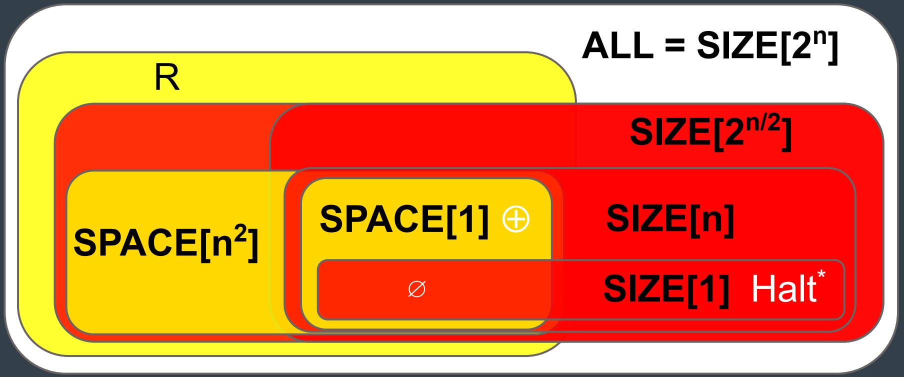
Uniform

- Fast Algorithm
- Constant Description
- No Preprocessing
- Static Program

Non-Uniform

- Fast Algorithm
- New Description For Every Input Size
- Precomputed
- Contains Unary Halting: HALT*

Circuit bounds



SPACE[T]: Programs That Use T bits of RAM

By Search:

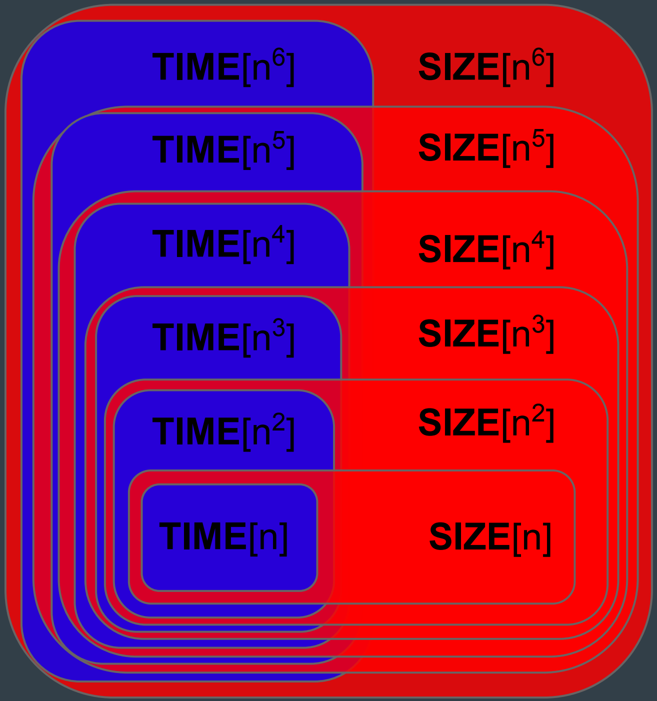
For $2^n/n > T_1 > T_0$,

$\mathrm{SPACE}[T_1] \not\subseteq \mathrm{SIZE}[T_0]$

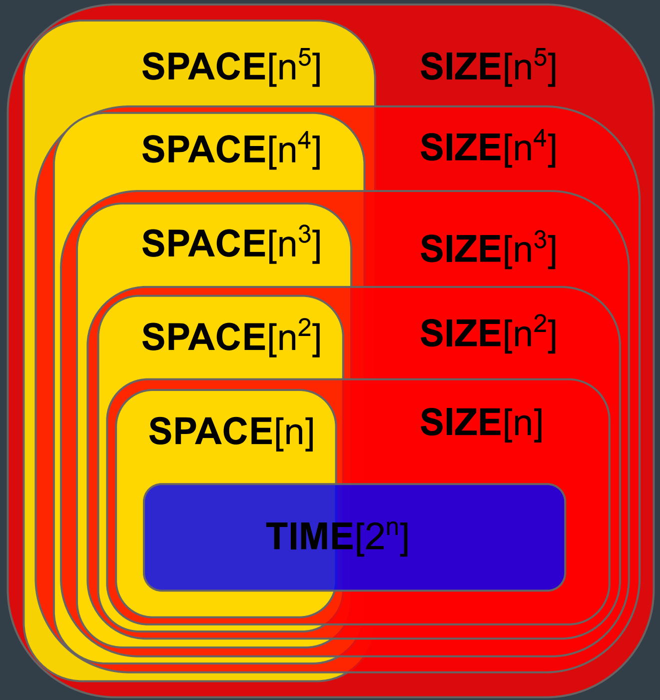
$\mathrm{HALT}^* \in \mathrm{SIZE}[O(1)]$

$\mathrm{HALT}^* \notin R$

Hope And Dream



Fear And Dread



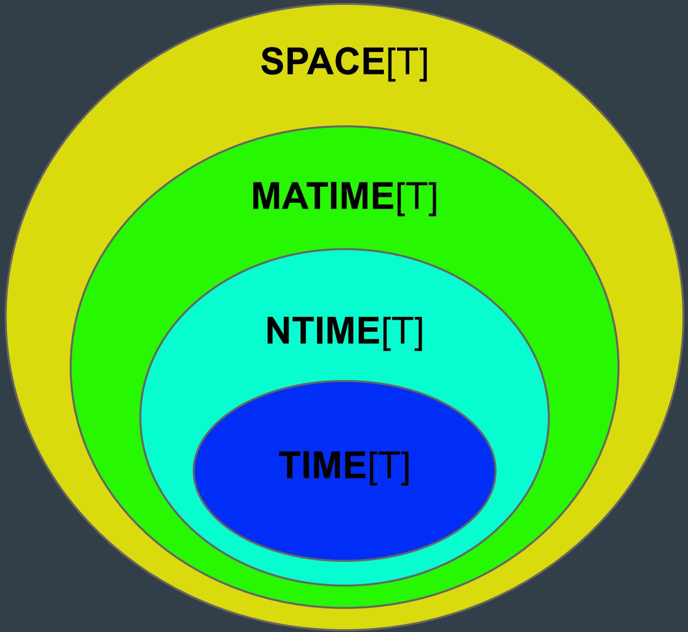
Towards Our Dreams

TIME circuit lower bounds hard?

Try NTIME!

Still too hard?

Try MATIME!

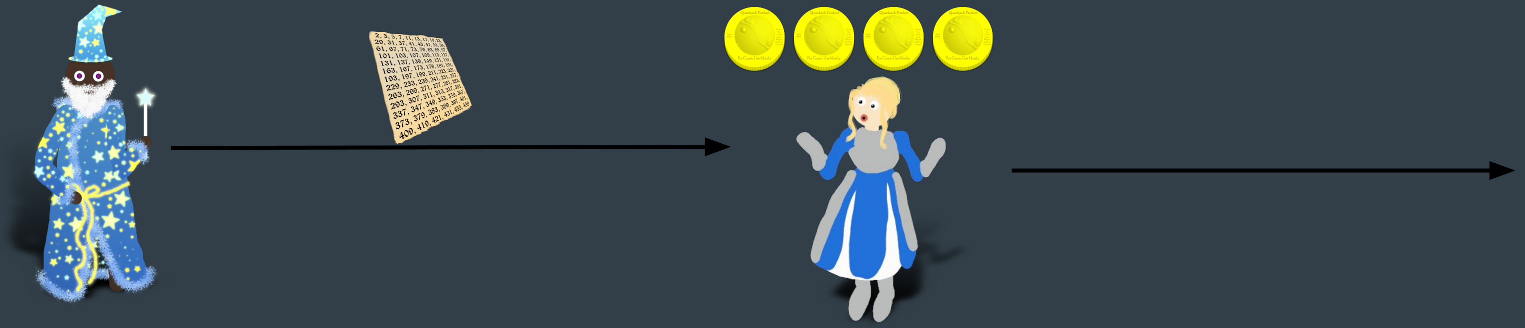


What is MATIME[T]?

MA, ‘Merlin Arthur’.

All Powerful Merlin Sends Proof.

Arthur Verifies in Time T with Randomness.



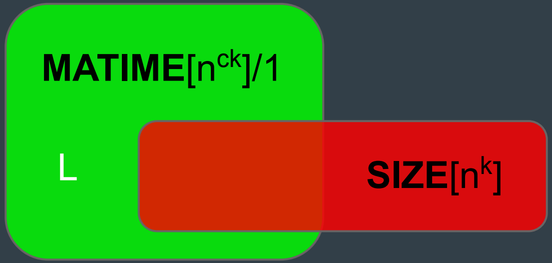
$x \overset{?}{\in} L$

Previous MA Lower Bounds

Santhanam, for some constant c, for all k:

$$\mathrm{MATIME}[n^{ck}]/1 \not\subseteq \mathrm{SIZE}[O(n^k)].$$

For some L



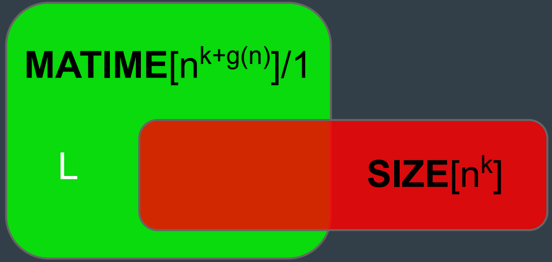
Might Still Have

Removing c!

We remove the factor of c, well, *almost*.

$$\mathrm{MATIME}[n^{k+g(n)}]/1 \not\subseteq \mathrm{SIZE}[O(n^k)].$$

- Has a subconstant, $g(n) = o(1)$.
- Only works for *some* $k > 1$, not **all** k.



What is “/1” in MATIME[T]/1?

A bit of trusted advice per input length.

A bit of non-uniformity.

Precomputing, Single Bit Result.

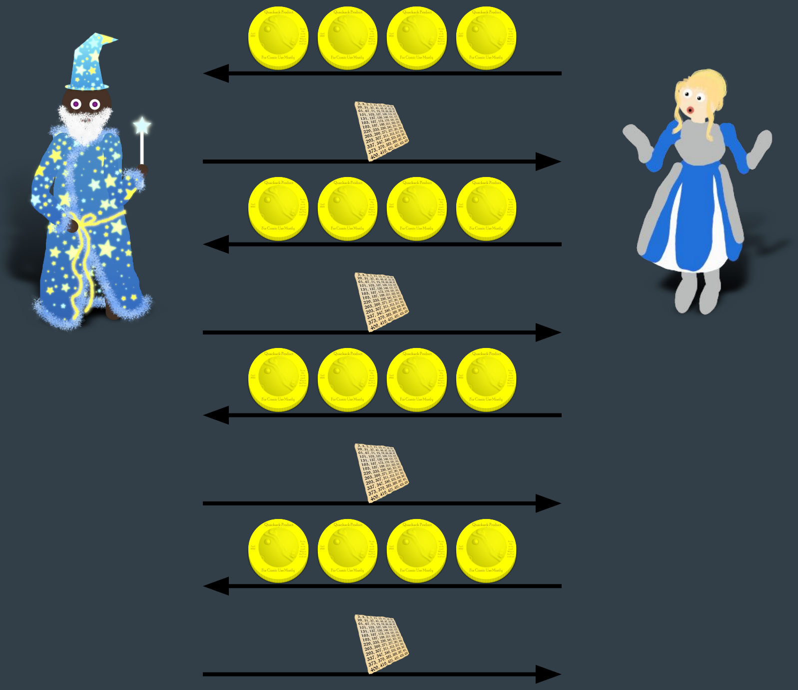
How to get Circuit Lower Bounds

Interactive Proofs (IPs)?

Untrusted Merlin
Randomized Arthur.

Many Questions and Answers.

IVTIME[T]: Arthur time T.



How powerful is IP?

Shamir 92 proved IP = PSPACE!

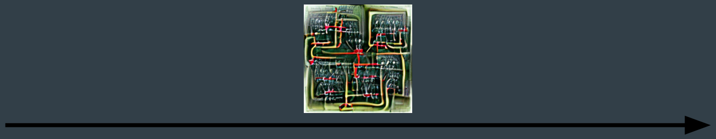
$$\mathrm{SPACE}[n] \subseteq \mathrm{IVTIME}[n^4]$$

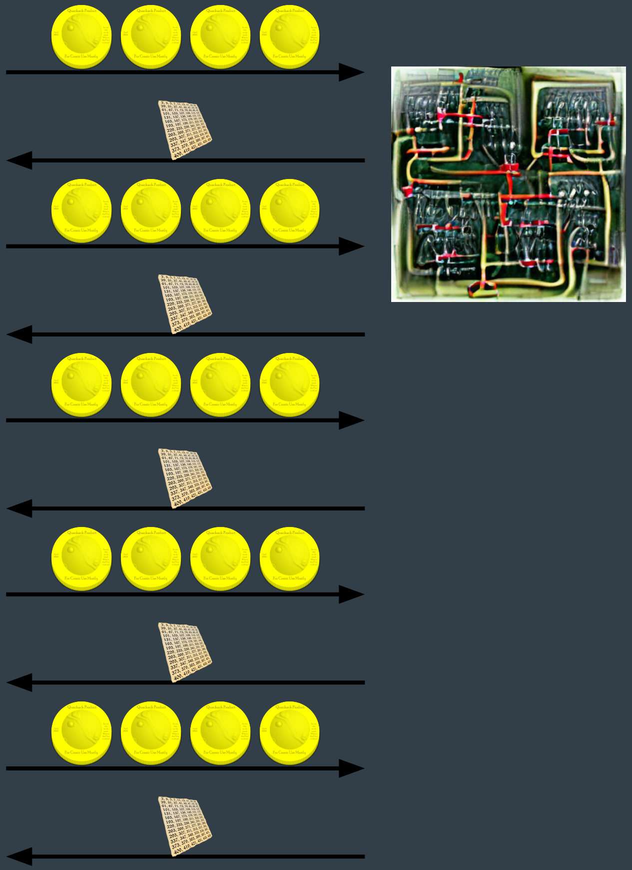
$$\mathrm{IVTIME}[n] \subseteq \mathrm{SPACE}[n]$$

Prover's for IP also small space!

Circuit bounds for SPACE apply to IP!

Main Idea





Use a Circuit as Merlin in IP.

Merlin Gives a Circuit
Arthur Uses it to run IP

Santhanam's Proof

If PSPACE $\subseteq$ P/poly

Problem in SPACE[n^k]

Hard for SIZE[$o(n^k)$]

Guess Circuit for Prover

PSPACE $\not\subseteq$ P/poly

SPACE[n] $\not\subseteq$ SIZE[n^k]

Pad SPACE[n] till prover has SIZE[n^k]

PSPACE $\not\subseteq$ P/poly Comments

Bit of Advice Needed for Pad Length.

Already Efficient, Case Unchanged by Us.

PSPACE ⊆ P/poly Analysis

PSPACE ⊆ P/poly → $\mathrm{SPACE}[n] \subseteq \mathrm{SIZE}[n^a]$

$L \in \mathrm{SPACE}[n^k]$ → L IP Verifier Time n^{4k}

→ L Prover Space n^{4k}

$\mathrm{SPACE}[n] \subseteq \mathrm{SIZE}[n^a]$ → L Prover SIZE n^{a4k}

L IP Verifier Time n^{4k} → n^{4k} Prover Queries

L MA Verifier Time → $n^{4k} + n^{4k}n^{a4k} = n^{(a+1)4k}$

Areas for improvement?

$$\mathrm{SPACE}[n^k] \subseteq \mathrm{MATIME}[n^{(a+1)4k}]$$

- a? Overhead From Circuit for SPACE.
  - Add Case Where $\mathrm{SPACE}[n] \subseteq \mathrm{SIZE}[n^{1+o(1)}]$
- +1? Too many Queries.
  - Use Low Query PCP.
- 4? IP Verifier is Slow.
  - Use Very Efficient PCP.

PCP: Non Adaptive Proof
Faster Verification

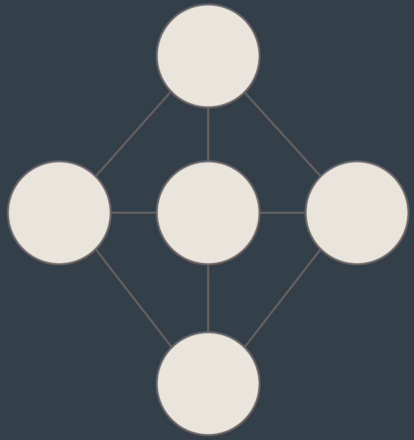
IP vs PCP (or IP vs MIP)

- PCP Prover Strategy Non-Adaptive
  - Prover Can’t Use Past Questions
  - New Prover Per Query
- PCP Can Use Fewer Queries
- PCP Is Faster
- Circuit Has No Memory, is PCP, **not** IP!

Example: Graph Three Coloring

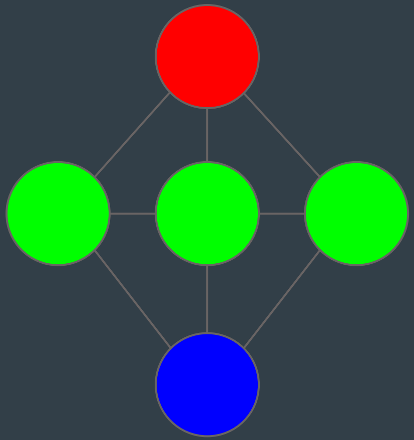
Assign Each Vertex a Color: Red, Green, or Blue.

Make Adjacent Vertices Different Colors.



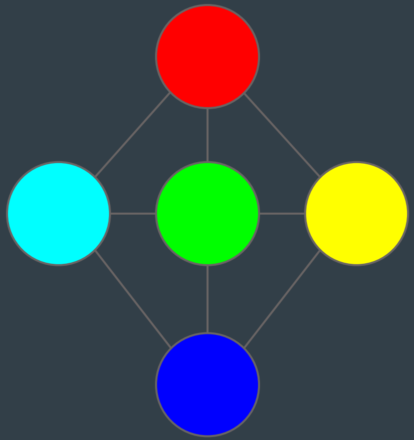
Bad

Green touching



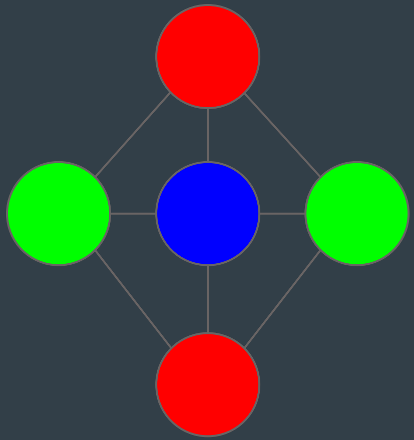
Bad

Uses 5 Colors



Good

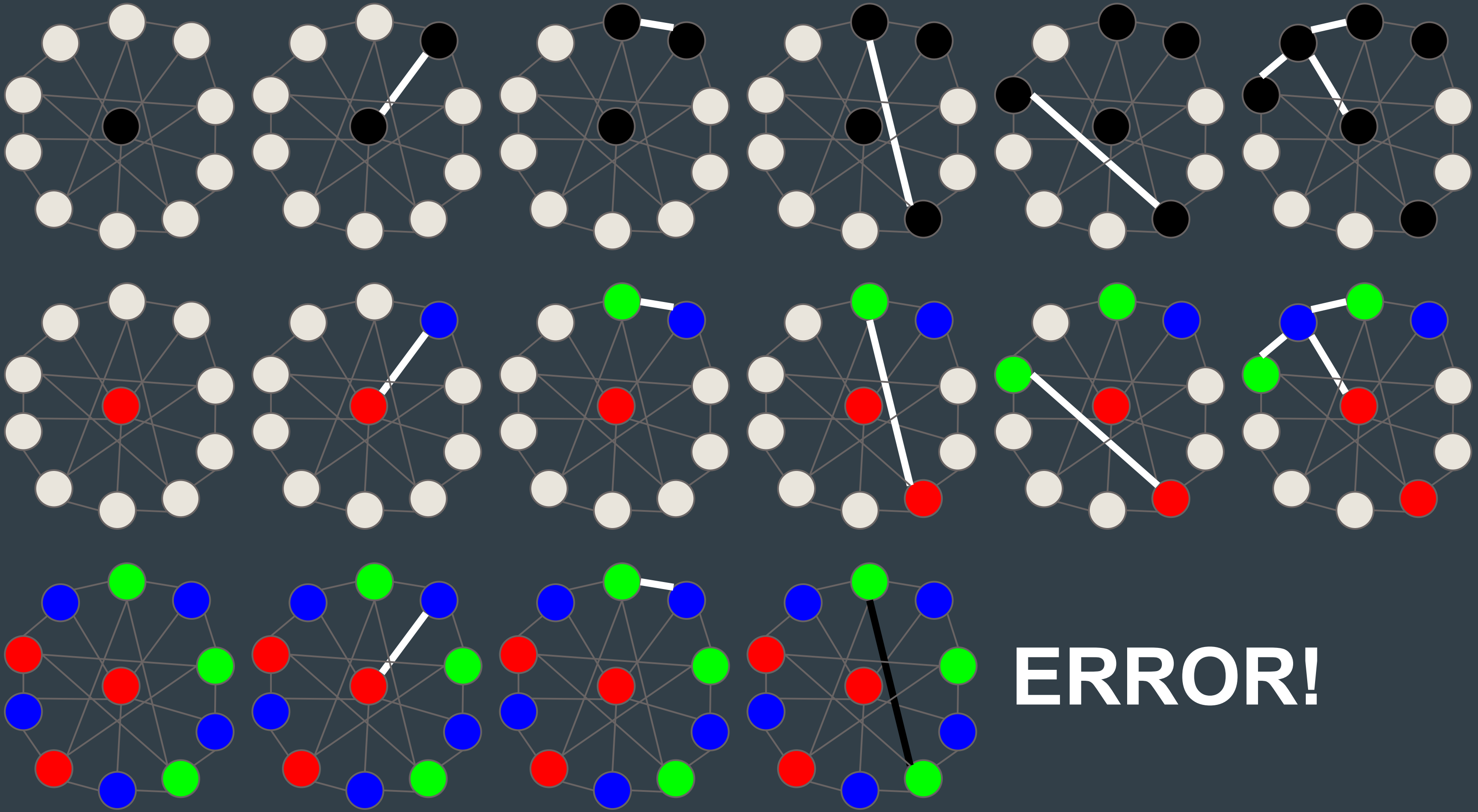
3 Colors,
No touching



Grötzsch graph

IP

PCP



Main Take Away

Fast Protocols Give Lower Bounds

Circuit Lower Bounds From Fast Verification / Algorithms

- Santhanam 2007 (Prior Work)
  - Circuit lower bound for MA/1
  - Through Efficient Interactive Proofs PSPACE
- Williams 2010
  - ACC Lower Bounds For NEXP
  - Through Fast SAT algorithms for ACC
- Murray Williams 2018
  - ACC Bounds for NQP
  - Through Interactive Proofs AND SAT algorithms

Second Result, Main Lemma

For L computable in time T and space S,

There is a PCP with

- Verifier time ~ n+log(T),
- polylog(n+log(T)) Queries
- and Prover space ~ n+S,

PCP Performance

For time T, space S algorithm

Old: Either verifier time $\sim n + \log(T)^2$

Queries $\sim \log(T)$

New: Verifier time $\sim n+\log(T)$, Prover space $\sim n+S$, $\log(n+\log(T))$ Queries.

Citations

Sanjeev Arora and Shmuel Safra. “Probabilistic Checking of Proofs: A New Characterization of NP”. JACM 1998.

L. Babai, L. Fortnow, and C. Lund. “Nondeterministic exponential time has two-prover interactive protocols”. FOCS 1990.

Joshua Cook, Dana Moshkovitz. “Tighter MA/1 Circuit Lower Bounds From Verifier Efficient PCPs for PSPACE”. 2022.

Cody Murray and Ryan Williams. “Circuit Lower Bounds for Non-deterministic Quasi-Polytime: An Easy Witness Lemma for NP and NQP”. STOC 2018.

Rahul Santhanam. “Circuit Lower Bounds for Merlin-Arthur Classes”. STOC ’07.

Adi Shamir. “IP = PSPACE”. JACM 1992.

Ryan Williams. “Non-uniform ACC Circuit Lower Bounds”. CCC 2011.