How Limited Interaction Hinders Real Communication (and What it Means for Proof and Circuit Complexity)

Marc Vinyals

KTH Royal Institute of Technology
Stockholm, Sweden

joint work with Susanna F. de Rezende and Jakob Nordström

August 12, University of Toronto, Canada
The SAT Problem

SAT solvers

- Very fast for industrial instances
- Scaling up to millions of variables
- But SAT is NP-complete!
The SAT Problem

SAT solvers
- Very fast for industrial instances
- Scaling up to millions of variables
- But SAT is NP-complete!

Proof complexity
- Examples of hard formulas
- Only theoretical tool so far
- Also easy formulas but hard in practice
  Why?
Proof Systems

Resolution

- Logic reasoning
- Most current SAT solvers
- Very well understood
Proof Systems

Resolution
- Logic reasoning
- Most current SAT solvers
- Very well understood

Polynomial calculus
- Algebraic reasoning
- Gaussian elimination used
- Reasonably understood
Proof Systems

Resolution
- Logic reasoning
- Most current SAT solvers
- Very well understood

Polynomial calculus
- Algebraic reasoning
- Gaussian elimination used
- Reasonably understood

Cutting planes
- Pseudobolean reasoning
- Experimental solvers
- Not well understood
Proof Systems

Resolution
- Logic reasoning
- Most current SAT solvers
- Very well understood

Polynomial calculus
- Algebraic reasoning
- Gaussian elimination used
- Reasonably understood

Cutting planes
- Pseudoboollean reasoning
- Experimental solvers
- Not well understood

Sums of squares
- Semidefinite programming
- Not used for SAT yet
- Not well understood
Cutting Planes

Work with inequalities

\[ x \lor \bar{y} \implies x + (1 - y) \geq 1 \implies x - y \geq 0 \]
Cutting Planes

Work with inequalities
\[ x \lor \bar{y} \rightarrow x + (1 - y) \geq 1 \rightarrow x - y \geq 0 \]

Rules

- **Variable axioms**
  \[
  \begin{align*}
  x & \geq 0 \quad & -x & \geq -1
  \end{align*}
  \]

- **Addition**
  \[
  \sum a_i x_i \geq a \quad \sum b_i x_i \geq b
  \]
  \[
  \sum (a_i + b_i) x_i \geq a + b
  \]

- **Division**
  \[
  \sum a_i x_i \geq a
  \]
  \[
  \sum (a_i / k) x_i \geq \lceil a / k \rceil
  \]
Cutting Planes

Work with inequalities

\[ x \lor \bar{y} \rightarrow x + (1 - y) \geq 1 \rightarrow x - y \geq 0 \]

Rules

Variable axioms

\[
\begin{align*}
x & \geq 0 \\
-x & \geq -1
\end{align*}
\]

Addition

\[
\frac{\sum a_i x_i \geq a}{\sum (a_i + b_i) x_i \geq a + b}
\]

Division

\[
\frac{\sum a_i x_i \geq a}{\sum (a_i / k) x_i \geq \lceil a / k \rceil}
\]

Goal: derive \( 0 \geq 1 \)
Complexity Measures

**Size**  # bits in proof
- Size $2^{O(N)}$ always possible.

**Length**  # lines in proof
- Worst case $2^{\Omega(N^\epsilon)}$. [Pudlák ’97]
Complexity Measures

**Size**  # bits in proof
- Size $2^{O(N)}$ always possible.

**Length**  # lines in proof
- Worst case $2^{\Omega(N^\epsilon)}$. [Pudlák ’97]

**Total space**  max # bits in memory at the same time
- Space $O(N^2)$ always possible; worst case $\Omega(N)$.

**Line space**  max # lines in memory at the same time
- Space 5 always possible. [Galesi, Pudlák, Thapen ’15]
Trade-offs

Question

Assume $F$ has a proof in length $L$ and another proof in space $s$. Is there a proof in length $O(L)$ and space $O(s)$?
Trade-offs

Question

Assume \( F \) has a proof in length \( L \) and another proof in space \( s \).
Is there a proof in length \( O(L) \) and space \( O(s) \)?

No
Trade-offs

Question

Assume $F$ has a proof in length $L$ and another proof in space $s$. Is there a proof in length $O(L)$ and space $O(s)$?

No

Previously studied for resolution and polynomial calculus

[Ben Sasson, Nordström ’11] [Beame, Beck, Impagliazzo ’12] [Beck, Nordström, Tang ’13]
Trade-offs

[Huynh, Nordström ’12]
Can do length $O(N)$, space $N^{1/2}$.
But space $N^{1/4-\epsilon}$ requires size $\exp(N^{\epsilon-o(1)})$. 
Trade-offs

$2^N$

$2^{N^\epsilon}$

$N$

$5$

$N^{1/4-\epsilon}$

$N^{1/2-\epsilon}$

$N^{1/2}$

[Goös, Pitassi ’14]

Can do length $N^{1+o(1)}$, space $N^{1/2+o(1)}$.

But space $N^{1/2-\epsilon}$ requires size $\exp(N^{\epsilon-o(1)})$. 
Trade-offs

\[ 2^N \]

\[ 2^{N^\epsilon} \]

\[ N \]

\[ 5 \quad N^{1/4-\epsilon} \quad N^{1/2-\epsilon} \quad N^{1/2} \]

[\text{Galesi, Pudlák, Thapen ’15}]
Can do length \( 2^N \), space 5.
 Trade-offs

\[ 2^N, 2^{N^\epsilon}, N, N^{1/4-\epsilon}, N^{1/2-\epsilon}, N^{1/2} \]

[Gallesi, Pudlák, Thapen ’15]

Can do length \(2^N\), space 5.

But exponential coefficients and quadratic total space.
Trade-offs

Question

Assume $F$ has a proof in small total space with polynomial coefficients. Are there still trade-offs?
Trade-offs

**Question**

Assume $F$ has a proof in small total space with polynomial coefficients. Are there still trade-offs?

Cannot answer with previous techniques (provably)
Trade-offs

Question

Assume $F$ has a proof in small total space with polynomial coefficients. Are there still trade-offs?

Cannot answer with previous techniques (provably)

This talk:

Yes
Main Result

Theorem

There is a family of 6-CNF formulas with

- short proofs: \( \text{size } O(N) \), \( \text{total space } O(N^{2/5}) \);
Main Result

Theorem

There is a family of 6-CNF formulas with

- short proofs: size $O(N)$, total space $O(N^{2/5})$;
- small space proofs: total space $O(N^{1/40})$, size $2^{O(N^{1/40})}$;
Main Result

Theorem

There is a family of 6-CNF formulas with

- short proofs: size \( \mathcal{O}(N) \), total space \( \mathcal{O}(N^{2/5}) \);
- small space proofs: total space \( \mathcal{O}(N^{1/40}) \), size \( 2^{\mathcal{O}(N^{1/40})} \);
- but line space \( N^{1/20 - \epsilon} \) requires length \( \exp(\Omega(N^{1/40})) \).
Main Result

Theorem

There is a family of 6-CNF formulas with

- short proofs: size $O(N)$, total space $O(N^{2/5})$;
- small space proofs: total space $O(N^{1/40})$, size $2^{O(N^{1/40})}$;
- but line space $N^{1/20 - \epsilon}$ requires length $\exp(\Omega(N^{1/40}))$.

- Upper bounds with constant coefficients, counting all bits.
- Lower bound with unbounded coefficients, only counting lines.
- Lower bound for semantic cutting planes.
Main Result

Theorem

There is a family of 6-CNF formulas with

- short proofs: \( \text{size } O(N), \text{ total space } O(N^{2/5}) \);
- small space proofs: \( \text{total space } O(N^{1/40}), \text{ size } 2^{O(N^{1/40})} \);
- but line space \( N^{1/20 - \epsilon} \) requires length \( \exp(\Omega(N^{1/40})) \).

- Upper bounds with constant coefficients, counting all bits.
- Lower bound with unbounded coefficients, only counting lines.
- Lower bound for semantic cutting planes.

- Holds for resolution and polynomial calculus proof systems.
Spin-off

Exponential separation of the monotone-AC hierarchy

Theorem

There is a monotone Boolean function with

- small monotone circuits: size $O(n)$, depth $\log^i(n)$, fan-in $n^{4/5}$
- but monotone circuits of depth $O(\log^{i-1} n)$ require size $\exp(\Omega(n^\epsilon))$.

Superpolynomial separation known [Raz, McKenzie '97]
Devious Plan

Assume refutation in length $L$ and space $s$
Devious Plan

Assume refutation in length $L$ and space $s$

\[ \Downarrow \]

1. Communication protocol for falsified clause search problem
Devious Plan

Assume refutation in length $L$ and space $s$

↓

1. Communication protocol for Search($F$)
Devious Plan

Assume refutation in length $L$ and space $s$

↓

1 Communication protocol for $\text{Search}(F)$

↓

2 Parallel decision tree for $\text{Search}(F)$
Devious Plan

Assume refutation in length $L$ and space $s$

1. Communication protocol for Search($F$)
2. Parallel decision tree for Search($F$)
3. Strategy for Dymond–Tompa pebble game
Devious Plan

Assume refutation in length $L$ and space $s$

1. Communication protocol for Search$(F)$

2. Parallel decision tree for Search$(F)$

3. Strategy for Dymond–Tompa pebble game

4. Construct graph with trade-offs
Devious Plan 1: Proof $\rightarrow$ Protocol

Refutation in length $L$, space $s \rightarrow$
Protocol for Search$(F)$ in $\log L$ rounds, communication $s \log L$

- Inspired by [Beame, Pitassi, Segerlind ’05] [Beame, Huynh, Pitassi ’10], explicit in [Huynh, Nordström ’12].

- Key twists:
  - Real communication model
  - Measure number of rounds
Real Communication

Introduced in [Krajíček ’98] to study cutting planes

▶ Compare real numbers at cost 1

Alice $\geq$ Referee

Bob
Real Communication

Introduced in [Krajíček ’98] to study cutting planes

- Compare real numbers at cost 1
Real Communication

Introduced in [Krajíček ’98] to study cutting planes

- Compare real numbers at cost 1

Alice $\geq -10^6, e^\pi$ Referee $\geq 8, \pi^e$ Bob $0, 1$
Real Communication

Introduced in [Krajíček ’98] to study cutting planes

- Compare real numbers at cost 1

Simulates deterministic communication (Alice sends $m$, Bob sends $1/2$)

Stronger than deterministic communication (EQ)
Devious Plan 1: Proof → Protocol

Falsified clause search on CNF $F(x, y)$

- Alice $\leftarrow$ assignment to $x$ variables
- Bob $\leftarrow$ assignment to $y$ variables
- Task: Find falsified clause
Devious Plan ①: Proof → Protocol

Falsified clause search on CNF $F(x, y)$

- Alice ← assignment to $x$ variables
- Bob ← assignment to $y$ variables
- Task: Find falsified clause

\[ \emptyset \]
Devious Plan 1: Proof $\rightarrow$ Protocol

Falsified clause search on CNF $F(x, y)$

- Alice $\leftarrow$ assignment to $x$ variables
- Bob $\leftarrow$ assignment to $y$ variables
- Task: Find falsified clause
Devious Plan 1: Proof → Protocol

Falsified clause search on CNF $F(x, y)$

- Alice ← assignment to $x$ variables
- Bob ← assignment to $y$ variables
- Task: Find falsified clause

- Alice evaluates $\sum a_i x_i - a$ in $s$ inequalities
- Bob evaluates $- \sum a_i y_i$ in $s$ inequalities
- $\alpha(C) = 1$ iff Referee answers 111...1
Devious Plan ①: Proof → Protocol

Falsified clause search on CNF $F(x, y)$

- Alice ← assignment to $x$ variables
- Bob ← assignment to $y$ variables
- Task: Find falsified clause
Devious Plan \(\textbf{1}: \) Proof $\rightarrow$ Protocol

Falsified clause search on CNF $F(x, y)$

- Alice $\leftarrow$ assignment to $x$ variables
- Bob $\leftarrow$ assignment to $y$ variables
- Task: Find falsified clause
Devious Plan 1: Proof $\rightarrow$ Protocol

Falsified clause search on CNF $F(x, y)$

- Alice $\leftarrow$ assignment to $x$ variables
- Bob $\leftarrow$ assignment to $y$ variables
- Task: Find falsified clause
Devious Plan 1: Proof → Protocol

Falsified clause search on CNF $F(x, y)$

- Alice ← assignment to $x$ variables
- Bob ← assignment to $y$ variables
- Task: Find falsified clause
Devious Plan 1: Proof → Protocol

Falsified clause search on CNF $F(x, y)$

- Alice ← assignment to $x$ variables
- Bob ← assignment to $y$ variables
- Task: Find falsified clause

\[ \alpha(C) = 1 \quad \alpha(C \cup \{A\}) = 0 \quad \Rightarrow \quad \alpha(A) = 0 \]

- $\log L$ rounds, communication $s \log L$
Devious Plan

Assume refutation in length $L$ and space $s$

1. Communication protocol for $\text{Search}(F)$ in $\log L$ rounds and communication $s \log L$
2. Parallel decision tree for $\text{Search}(F)$
3. Strategy for Dymond–Tompa pebble game
4. Construct graph with trade-offs
Devious Plan 2: Protocol $\rightarrow$ Decision Tree

Protocol for Lift$(S)$ in $r$ rounds, communication $c$ $\rightarrow$
Parallel decision tree for $S$ of depth $r$, $c$ queries
Lifted Problem

- Function $f(z_1, \ldots, z_n)$
- Alice $\leftarrow n$ indices $x_1, \ldots, x_n$
- Bob $\leftarrow n$ arrays $y_1, \ldots, y_n$

\[
\begin{align*}
  z_1 &= y_1[5] = 1 \\
  x_1 &= \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \\
  y_1 &= \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}
\end{align*}
\]

- Lifted function $\text{Lift}(f)(x, y) = f(y_1[x_1], \ldots, y_n[x_n])$
Parallel Decision Trees

Decision tree with many queries per node [Valiant ’75]

Depth  Longest branch
Queries  # queries in a branch
Devious Plan ②: Protocol $\rightarrow$ Decision Tree

Protocol for Lift($S$) in $r$ rounds, communication $c$  $\rightarrow$
Parallel decision tree for $S$ of depth $r$, $c$ queries
Devious Plan $\mathcal{S}$: Protocol $\leftarrow$ Decision Tree

 Protocol for Lift($S$) in $r$ rounds, communication $c$ $\leftarrow$
 Parallel decision tree for $S$ of depth $r$, $c$ queries

Communication          Decision tree
Query $\{z_3, z_{28}\}$
Devious Plan $\mathcal{S}$: Protocol $\leftarrow$ Decision Tree

Protocol for Lift$(S)$ in $r$ rounds, communication $c \leftarrow$
Parallel decision tree for $S$ of depth $r$, $c$ queries

Communication
Alice sends $x_3, x_{28}$
Bob sends $y_3[x_3], y_{28}[x_{28}]$

Decision tree
Query $\{z_3, z_{28}\}$
Devious Plan 2: Protocol → Decision Tree

Protocol for $\text{Lift}(S)$ in $r$ rounds, communication $c$ → Parallel decision tree for $S$ of depth $r$, $c$ queries

Communication
Alice sends $x_1 + x_2 + \cdots + x_n$
Devious Plan 2: Protocol → Decision Tree

Protocol for \( \text{Lift}(S) \) in \( r \) rounds, communication \( c \) →
Parallel decision tree for \( S \) of depth \( r \), \( c \) queries

Communication
Alice sends \( x_1 + x_2 + \cdots + x_n \)

Decision tree
???
Devious Plan 2: Protocol $\rightarrow$ Decision Tree

Protocol for $\text{Lift}(S)$ in $r$ rounds, communication $c$ $\rightarrow$
Parallel decision tree for $S$ of depth $r$, $c$ queries

- Main technical result (Simulation Theorem)
  - Technique from [Raz, McKenzie ’97]
  - Adapted to real communication in [Bonet, Esteban, Galesi, Johannsen ’98]
  - Connection to decision trees made explicit in [Göös, Pitassi, Watson ’15]

- Our contribution
  - Introduce rounds
  - Adapt to real communication preserving rounds
Devious Plan

Assume refutation of lifted formula in length $L$ and space $s$

1. Communication protocol for $\text{Lift}(\text{Search}(F))$ in $\log L$ rounds and communication $s \log L$
2. Parallel decision tree for $\text{Search}(F)$ of depth $\log L$ and $s \log L$ queries
3. Strategy for Dymond–Tompa pebble game
4. Construct graph with trade-offs
Devious Plan 3: Decision Tree $\rightarrow$ Dymond–Tompa

Parallel decision tree for $\text{Search}(\text{Peb}_G)$ of depth $r$, $c$ queries $\leftrightarrow$
Dymond–Tompa pebble game strategy for $r$ rounds, $c$ pebbles
Pebbling Formulas

- Sources are true
  \[ u \]
  \[ v \]
  \[ w \]

- Truth propagates
  \[(u \land v) \rightarrow x\]
  \[(v \land w) \rightarrow y\]
  \[(x \land y) \rightarrow z\]

- Sink is false
  \[ \overline{z} \]
Dymond–Tompa Game

2-player pebble game on a DAG [Dymond, Dompa ’85]
Dymond–Tompa Game

2-player pebble game on a DAG [Dymond, Dompa ’85]

- Start with a challenged pebble on the sink

Rounds 0
Pebbles 1
Dymond–Tompa Game

2-player pebble game on a DAG [Dymond, Dompa ’85]

- Start with a challenged pebble on the sink
- Each round:
  - Pebbler adds some pebbles

Rounds 1
Pebbles 4
Dymond–Tompa Game

2-player pebble game on a DAG [Dymond, Dompa ’85]

- Start with a challenged pebble on the sink
- Each round:
  - Pebbler adds some pebbles
  - Challenger may challenge one new pebble

Rounds 1

Pebbles 4
Dymond–Tompa Game

2-player pebble game on a DAG [Dymond, Dompa ’85]

- Start with a challenged pebble on the sink
- Each round:
  - Pebbler adds some pebbles
  - Challenger may challenge one new pebble

Rounds 2
Pebbles 7
Dymond–Tompa Game

2-player pebble game on a DAG [Dymond, Dompa ’85]

- Start with a challenged pebble on the sink
- Each round:
  - Pebbler adds some pebbles
  - Challenger may challenge one new pebble

Rounds 2
Pebbles 7
Dymond–Tompa Game

2-player pebble game on a DAG [Dymond, Dompa ’85]

- Start with a challenged pebble on the sink
- Each round:
  - Pebbler adds some pebbles
  - Challenger may challenge one new pebble

Rounds 3
Pebbles 9
Dymond–Tompa Game

2-player pebble game on a DAG [Dymond, Dompa ’85]

- Start with a challenged pebble on the sink
- Each round:
  - Pebbler adds some pebbles
  - Challenger may challenge one new pebble
- Ends when challenged pebble is surrounded

Rounds  3
Pebbles  9
Devious Plan 3: Decision Tree $\rightarrow$ Dymond–Tompa

Parallel decision tree for $\text{Search}(\text{Peb}_G)$ of depth $r$, $c$ queries $\leftrightarrow$
Dymond–Tompa pebble game strategy for $r$ rounds, $c$ pebbles

- Done in [Chan ’13]
- Tweak to preserve rounds
Devious Plan

Assume refutation of lifted *pebbling* formula in length $L$ and space $s$

1. Communication protocol for \( \text{Lift} (\text{Search} (F)) \)
   in $\log L$ rounds and communication $s \log L$

2. Parallel decision tree for \( \text{Search} (F) \)
   of depth $\log L$ and $s \log L$ queries

3. Strategy for Dymond–Tompa pebble game
   for $\log L$ rounds and $s \log L$ pebbles [Chan ’13]

4. Construct graph with trade-offs
Devious Plan ④: Trade-off for Dymond–Tompa

Graph where $r$-round DT game needs $n/4$ pebbles

- Stack of $r + 1$ butterfly graphs
- Can do $2r \log n$ pebbles in $r \log n$ rounds
- Or $n \log (r \log n)$ pebbles in $\log (r \log n)$ rounds
Devious Plan

Assume refutation of lifted pebbling formula in length $L$ and space $s$

1. Communication protocol for $\text{Lift}(\text{Search}(F))$ in $\log L$ rounds and communication $s \log L$
2. Parallel decision tree for $\text{Search}(F)$ of depth $\log L$ and $s \log L$ queries
3. Strategy for Dymond–Tompa pebble game for $\log L$ rounds and $s \log L$ pebbles
4. Construct graph where such strategy does not exist
Take Home

Remarks

▶ Strong size-space trade-offs for cutting planes
▶ Hold for resolution, polynomial calculus, cutting planes
▶ Key to measure rounds
Take Home

Remarks
- Strong size-space trade-offs for cutting planes
- Hold for resolution, polynomial calculus, cutting planes
- Key to measure rounds

Open problems
- Smaller lift size
- Stronger models of communication
Take Home

Remarks

- Strong size-space trade-offs for cutting planes
- Hold for resolution, polynomial calculus, cutting planes
- Key to measure rounds

Open problems

- Smaller lift size
- Stronger models of communication

Thanks!