Time-space trade-offs in proof complexity
Lecture 3

Jakob Nordström

KTH Royal Institute of Technology

17th Estonian Winter School in Computer Science
Palmse, Estonia
February 26 – March 2, 2012

Agenda for Today's Lecture

- Focus on polynomial calculus (but also talk some about resolution)
- Recall definitions and discuss variants of polynomial calculus
- Brief overview of what is known for proof length and proof space
- Prove space lower bound
  - first for resolution (as warm-up)
  - then for polynomial calculus (start today, finish next lecture)
- News right from the research frontier — believe further improvements within reach if we can understand techniques better
- Going gets slightly tougher — might want to reviews slides at `www.csc.kth.se/~jakobn/teaching/ewscs12/` to follow 100%
- Makes it extra important to ask questions (especially "stupid" ones)

Resolution

- Lines in refutation are disjunctive clauses
- Just one inference rule, the resolution rule:

$$\frac{B \lor x \quad C \lor \overline{x}}{B \lor C}$$

  $B \lor C$ is the resolvent of $B \lor x$ and $C \lor \overline{x}$

- Prove F unsatisfiable by deriving the unsatisfiable empty clause $\bot$ (the clause with no literals) from F by resolution

Polynomial Calculus (PC)

- Axiom clauses of F interpreted as multilinear polynomial equations
- "Being true" corresponds to "evaluating to zero," so natural to flip convention and think of 0 as true and 1 as false
- By way of example, clause $x \lor y \lor \overline{z}$ gets translated to $xy(1-z)=0$
- To get unique representation, write polynomials in expanded form as sums of monomials; hence clause above becomes $xy - xyz = 0$
- Prove F unsatisfiable by showing that there is no common root for the polynomial equations corresponding to the axiom clauses

Polynomial Calculus: Inference Rules

Lines in polynomial calculus refutation: multivariate polynomial equations $p = 0$, where $p \in \mathbb{F}[x, y, z, \ldots]$ for some fixed field $\mathbb{F}$ (typically finite)

Customary to omit "$= 0$" from "$p = 0$" and only write "p"

Derivation rules ($\alpha, \beta \in \mathbb{F}$, $p, q \in \mathbb{F}[x, y, z, \ldots]$, x any variable):

Boolean axioms $\dfrac{}{x^2 - x}$ (forcing 0/1-solutions)

Linear combination $\dfrac{p \qquad q}{\alpha p + \beta q}$

Multiplication $\dfrac{p}{xp}$

PC-refutation ends when 1 is derived (i.e., $1 = 0$)

(Multilinearity follows w.l.o.g. from Boolean axioms)

Polynomial Calculus: Inference Rules

Lines in polynomial calculus refutation: multivariate polynomial equations $p = 0$, where $p \in \mathbb{F}[x, y, z, \ldots]$ for some fixed field $\mathbb{F}$ (typically finite)

Customary to omit "$= 0$" from "$p = 0$" and only write "p"

Derivation rules ($\alpha, \beta \in \mathbb{F}$, $p, q \in \mathbb{F}[x, y, z, \ldots]$, x any variable):

Boolean axioms $\dfrac{}{x^2 - x}$ (forcing 0/1-solutions)

Linear combination $\dfrac{p \qquad q}{\alpha p + \beta q}$

Multiplication $\dfrac{p}{xp}$

PC-refutation ends when 1 is derived (i.e., $1 = 0$)

(Multilinearity follows w.l.o.g. from Boolean axioms)

Polynomial Calculus: Inference Rules

Lines in polynomial calculus refutation: multivariate polynomial equations $p = 0$, where $p \in \mathbb{F}[x, y, z, \ldots]$ for some fixed field $\mathbb{F}$ (typically finite)

Customary to omit "$= 0$" from "$p = 0$" and only write "p"

Derivation rules ($\alpha, \beta \in \mathbb{F}$, $p, q \in \mathbb{F}[x, y, z, \ldots]$, x any variable):

Boolean axioms $\dfrac{}{x^2 - x}$ (forcing 0/1-solutions)

Linear combination $\dfrac{p \qquad q}{\alpha p + \beta q}$

Multiplication $\dfrac{p}{xp}$

PC-refutation ends when 1 is derived (i.e., $1 = 0$)

(Multilinearity follows w.l.o.g. from Boolean axioms)

Polynomial Calculus: Complexity Measures

Polynomial Calculus is sound and complete, just as resolution
(requires a proof, of course)

Complexity measures that we care about today:

- **Size**
  Total # monomials in the refutation counted with repetitions
  (Analogue of length in resolution)
- **(Monomial) space**
  Maximal # monomials in any configuration counted with repetitions
  (Analogue of clause space in resolution)

Annoying Problem with Encoding Clauses as Polynomials

Consider $\overline{x}_1 \vee \overline{x}_2 \vee \ldots \vee \overline{x}_w$

Gets translated to

$$\prod_{i=1}^{w}(1 - x_i) = \sum_{S \subseteq [w]} (-1)^{|S|} \prod_{i \in S} x_i$$

Exponential size in w ⇒ exponential lower bounds on size and space!

Great! Except that somehow this particular type of exponential lower bound is not what we are looking for...

Two fixes:

1. Consider only k-CNF formulas for constant k
2. Introduce extra variables for negated literals [Alekhnovich et al. '00]

Annoying Problem with Encoding Clauses as Polynomials

Consider $\overline{x}_1 \lor \overline{x}_2 \lor \ldots \lor \overline{x}_w$

Gets translated to

$$\prod_{i=1}^{w}(1 - x_i) = \sum_{S \subseteq [w]} (-1)^{|S|} \prod_{i \in S} x_i$$

Exponential size in w ⇒ exponential lower bounds on size and space!

Great! Except that somehow this particular type of exponential lower bound is not what we are looking for...

Two fixes:

1. Consider only k-CNF formulas for constant k
2. Introduce extra variables for negated literals [Alekhnovich et al. '00]

Annoying Problem with Encoding Clauses as Polynomials

Consider $\overline{x}_1 \lor \overline{x}_2 \lor \ldots \lor \overline{x}_w$

Gets translated to

$$\prod_{i=1}^{w}(1 - x_i) = \sum_{S \subseteq [w]} (-1)^{|S|} \prod_{i \in S} x_i$$

Exponential size in w ⇒ exponential lower bounds on size and space!

Great! Except that somehow this particular type of exponential lower bound is not what we are looking for...

Two fixes:

1. Consider only k-CNF formulas for constant k
2. Introduce extra variables for negated literals [Alekhnovich et al. '00]

Polynomial Calculus Resolution (PCR)

New variables $\overline{x}, \overline{y}, \overline{z}, \ldots$,

$x \vee y \vee \overline{z}$ gets translated to (surprise!) $xy\overline{z}$

Need to enforce that $\overline{x}$ is the negation of x:

Complementarity axiom $\dfrac{}{x + \overline{x} - 1}$ for all variables x

All other rules and complexity measures same as for polynomial calculus

Proposition

PCR simulates resolution proofs in essentially same length, size and space

In the best of worlds we want to:

- Prove upper bounds for PC
- Prove (matching) lower bounds for PCR

Polynomial Calculus Resolution (PCR)

New variables $\overline{x}, \overline{y}, \overline{z}, \ldots,$

$x \vee y \vee \overline{z}$ gets translated to (surprise!) $xy\overline{z}$

Need to enforce that $\overline{x}$ is the negation of x:

Complementarity axiom $\dfrac{}{x + \overline{x} - 1}$ for all variables x

All other rules and complexity measures same as for polynomial calculus

Proposition

PCR simulates resolution proofs in essentially same length, size and space

In the best of worlds we want to:

- Prove upper bounds for PC
- Prove (matching) lower bounds for PCR

Size and Space Bounds for PC/PCR

($n =$ size of formula)

Size: at most $\exp(\mathcal{O}(n))$ for PC for k-CNF formulas [Filmus et al. '12]
Matching lower bound for PCR up to constant factors in exponent
e.g. [Alekhnovich & Razborov '01]

Space: at most $\mathcal{O}(n)$ for PC for k-CNF formulas [Filmus et al. '12]
No matching lower bounds

In fact, until recently **no nontrivial space lower bounds** for k-CNF formulas even in PC!

$\Omega\left(\sqrt[3]{n}\right)$ bound for wide formulas in PCR [Alekhnovich et al. '00]
But proof doesn't work for constant-width formulas

Size and Space Bounds for PC/PCR

($n =$ size of formula)

Size: at most $\exp(\mathcal{O}(n))$ for PC for k-CNF formulas [Filmus et al. '12]
Matching lower bound for PCR up to constant factors in exponent
e.g. [Alekhnovich & Razborov '01]

Space: at most $\mathcal{O}(n)$ for PC for k-CNF formulas [Filmus et al. '12]
No matching lower bounds

In fact, until recently **no nontrivial space lower bounds** for k-CNF formulas even in PC!

$\Omega\left(\sqrt[3]{n}\right)$ bound for wide formulas in PCR [Alekhnovich et al. '00]
But proof doesn't work for constant-width formulas

Size and Space Bounds for PC/PCR

($n =$ size of formula)

Size: at most $\exp(\mathcal{O}(n))$ for PC for k-CNF formulas [Filmus et al. '12]
Matching lower bound for PCR up to constant factors in exponent
e.g. [Alekhnovich & Razborov '01]

Space: at most $\mathcal{O}(n)$ for PC for k-CNF formulas [Filmus et al. '12]
No matching lower bounds

In fact, until recently **no nontrivial space lower bounds** for k-CNF formulas even in PC!

$\Omega(\sqrt[3]{n})$ bound for wide formulas in PCR [Alekhnovich et al. '00]
But proof doesn't work for constant-width formulas

Size and Space Bounds for PC/PCR

($n =$ size of formula)

Size: at most $\exp(\mathcal{O}(n))$ for PC for k-CNF formulas [Filmus et al. '12]
Matching lower bound for PCR up to constant factors in exponent
e.g. [Alekhnovich & Razborov '01]

Space: at most $\mathcal{O}(n)$ for PC for k-CNF formulas [Filmus et al. '12]
No matching lower bounds

In fact, until recently **no nontrivial space lower bounds** for k-CNF formulas even in PC!

$\Omega\left(\sqrt[3]{n}\right)$ bound for wide formulas in PCR [Alekhnovich et al. '00]
But proof doesn't work for constant-width formulas

This Lecture and Next: Space Lower Bounds for k-CNFs

In the rest of this course, will prove first space lower bound for k-CNFs

Theorem (Filmus, Lauria, Nordström, Thapen & Zewi '12)

There are k-CNF formulas F_n of size n s.t. $Sp_{\mathcal{PCR}}(F_n \vdash \bot) = \Omega\big(\sqrt[3]{n}\big)$

Or, actually, we will:

- Do slightly weaker result with simpler proof and more natural formulas
- But provide all crucial ingredients needed for proof of stronger result
- To get a feel for the proof method, warm up with resolution

This Lecture and Next: Space Lower Bounds for k-CNFs

In the rest of this course, will prove first space lower bound for k-CNFs

Theorem (Filmus, Lauria, Nordström, Thapen & Zewi '12)

There are k-CNF formulas F_n of size n s.t. $Sp_{\mathcal{PCR}}(F_n \vdash \bot) = \Omega\bigl(\sqrt[3]{n}\bigr)$

Or, actually, we will:

- Do slightly weaker result with simpler proof and more natural formulas
- But provide all crucial ingredients needed for proof of stronger result
- To get a feel for the proof method, warm up with resolution

Pigeonhole Principle

- m pigeons
- n pigeonholes
- Every pigeon wants its own hole
- Impossible if $m > n$

Write (contradiction of) this as CNF formula

Various encodings of this combinatorial principle most studied formulas in proof complexity

(From now on, n is not formula size but # pigeon holes)

Pigeonhole Principle

- m pigeons
- n pigeonholes
- Every pigeon wants its own hole
- Impossible if $m > n$

Write (contradiction of) this as CNF formula

Various encodings of this combinatorial principle most studied formulas in proof complexity

(From now on, n is not formula size but $\#$ pigeon holes)

Bitwise Pigeonhole Principle Formula $BPHP^m_n$

$$x^b = \begin{cases} x & \text{if } b = 0 \\ \overline{x} & \text{if } b = 1 \end{cases}$$

(x^b is true if and only if $x = b$)

$[0, j) = \{0, 1, \ldots, j-1\}$ (will index pigeons and holes starting from 0)

$n = 2^\ell$ (only consider even powers of 2 for # holes)

Variables $x[p, i]$ for each $p \in [0, m)$ and $i \in [0, \ell)$

Pigeon p sent to hole $x[p, \ell{-}1] \cdots x[p, 1]x[p, 0]$ (in binary encoding)

For all $p \neq q \in [0, m)$, $h = h_{\ell-1} \cdots h_0 \in [0, n)$, hole axiom

$$H(p, q, h) = \bigvee_{i=0}^{\ell-1} x[p, i]^{1-h_i} \vee \bigvee_{i=0}^{\ell-1} x[q, i]^{1-h_i}$$

"Have $m > n$ integers between 0 and $n - 1$ and they're all distinct"

Bitwise Pigeonhole Principle Formula $BPHP^m_n$

$$x^b = \begin{cases} x & \text{if } b = 0 \\ \overline{x} & \text{if } b = 1 \end{cases}$$ (x^b is true if and only if $x = b$)

$[0, j) = \{0, 1, \ldots, j-1\}$ (will index pigeons and holes starting from 0)

$n = 2^\ell$ (only consider even powers of 2 for # holes)

Variables $x[p, i]$ for each $p \in [0, m)$ and $i \in [0, \ell)$

Pigeon p sent to hole $x[p, \ell-1] \cdots x[p, 1] x[p, 0]$ (in binary encoding)

For all $p \neq q \in [0, m)$, $h = h_{\ell-1} \cdots h_0 \in [0, n)$, hole axiom

$$H(p, q, h) = \bigvee_{i=0}^{\ell-1} x[p, i]^{1-h_i} \vee \bigvee_{i=0}^{\ell-1} x[q, i]^{1-h_i}$$

"Have $m > n$ integers between 0 and $n-1$ and they're all distinct"

Bitwise Pigeonhole Principle Formula $BPHP^m_n$

$$x^b = \begin{cases} x & \text{if } b = 0 \\ \overline{x} & \text{if } b = 1 \end{cases}$$ (x^b is true if and only if $x = b$)

$[0, j) = \{0, 1, \ldots, j - 1\}$ (will index pigeons and holes starting from 0)

$n = 2^\ell$ (only consider even powers of 2 for # holes)

Variables $x[p, i]$ for each $p \in [0, m)$ and $i \in [0, \ell)$

Pigeon p sent to hole $x[p, \ell-1] \cdots x[p, 1]x[p, 0]$ (in binary encoding)

For all $p \neq q \in [0, m)$, $h = h_{\ell-1} \cdots h_0 \in [0, n)$, hole axiom

$$H(p, q, h) = \bigvee_{i=0}^{\ell-1} x[p, i]^{1-h_i} \vee \bigvee_{i=0}^{\ell-1} x[q, i]^{1-h_i}$$

"Have $m > n$ integers between 0 and $n - 1$ and they're all distinct"

Restating the (Weaker) Theorem(s)

Theorem

$Sp_{\mathcal{PCR}}(BPHP^m_n \vdash \perp) > n/8$

Width of clauses $2\ell = \mathcal{O}(\log n)$ — non-constant

But space polynomial while width logarithmic — already exponential improvement over [Alekhnovich et al. '00] where space $<$ width

First do easier bound for resolution:

Theorem

$Sp_{\mathcal{R}}(BPHP^m_n \vdash \perp) \geq n$

Restating the (Weaker) Theorem(s)

Theorem

$Sp_{\mathcal{PCR}}(BPHP^m_n \vdash \bot) > n/8$

Width of clauses $2\ell = \mathcal{O}(\log n)$ — non-constant

But space polynomial while width logarithmic — already exponential improvement over [Alekhnovich et al. '00] where space $<$ width

First do easier bound for resolution:

Theorem

$Sp_{\mathcal{R}}(BPHP^m_n \vdash \bot) \geq n$

Restating the (Weaker) Theorem(s)

Theorem

$Sp_{\mathcal{PCR}}(BPHP^m_n \vdash \bot) > n/8$

Width of clauses $2\ell = \mathcal{O}(\log n)$ — non-constant

But space polynomial while width logarithmic — already exponential improvement over [Alekhnovich et al. '00] where space $<$ width

First do easier bound for resolution:

Theorem

$Sp_{\mathcal{R}}(BPHP^m_n \vdash \bot) \geq n$

The General Proof Strategy [Alekhnovich et al. '00]

- Given formula F and space bound s, want to prove $Sp(F \vdash \bot) \geq s$
- Equivalently, study derivations $\pi = \{\mathbb{P}_0 = \emptyset, \ldots, \mathbb{P}_\tau\}$ s.t. $Sp(\pi) < s$ and show π doesn't derive contradiction (i.e., $1 \notin \mathbb{P}_t$ for all t)
- Begs the question — we don't understand what $\mathbb{P}_t$ looks like!
- Study $\mathbb{P}_t$ by constructing auxiliary configuration $\mathcal{A}_t$ that is easier to understand but gives information about $\mathbb{P}_t$:
  1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
  2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
  3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough (that is, less than s)
- If we can do this, clearly we immediately get lower bound on space

The General Proof Strategy [Alekhnovich et al. '00]

- Given formula F and space bound s, want to prove $Sp(F \vdash \bot) \geq s$
- Equivalently, study derivations $\pi = \{\mathbb{P}_0 = \emptyset, \ldots, \mathbb{P}_\tau\}$ s.t. $Sp(\pi) < s$ and show π doesn't derive contradiction (i.e., $1 \notin \mathbb{P}_t$ for all t)
- Begs the question — we don't understand what $\mathbb{P}_t$ looks like!
- Study $\mathbb{P}_t$ by constructing auxiliary configuration $\mathcal{A}_t$ that is easier to understand but gives information about $\mathbb{P}_t$:
  1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
  2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
  3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough (that is, less than s)
- If we can do this, clearly we immediately get lower bound on space

The General Proof Strategy [Alekhnovich et al. '00]

- Given formula F and space bound s, want to prove $Sp(F \vdash \bot) \geq s$
- Equivalently, study derivations $\pi = \{\mathbb{P}_0 = \emptyset, \ldots, \mathbb{P}_\tau\}$ s.t. $Sp(\pi) < s$ and show π doesn't derive contradiction (i.e., $1 \notin \mathbb{P}_t$ for all t)
- Begs the question — we don't understand what $\mathbb{P}_t$ looks like!
- Study $\mathbb{P}_t$ by constructing auxiliary configuration $\mathcal{A}_t$ that is easier to understand but gives information about $\mathbb{P}_t$:
  1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
  2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
  3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough (that is, less than s)
- If we can do this, clearly we immediately get lower bound on space

The General Proof Strategy [Alekhnovich et al. ’00]

- Given formula F and space bound s, want to prove $Sp(F \vdash \bot) \geq s$
- Equivalently, study derivations $\pi = \{\mathbb{P}_0 = \emptyset, \ldots, \mathbb{P}_\tau\}$ s.t. $Sp(\pi) < s$ and show π doesn’t derive contradiction (i.e., $1 \notin \mathbb{P}_t$ for all t)
- Begs the question — we don’t understand what $\mathbb{P}_t$ looks like!
- Study $\mathbb{P}_t$ by constructing auxiliary configuration $\mathcal{A}_t$ that is easier to understand but gives information about $\mathbb{P}_t$:
    1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ “stronger” than $\mathbb{P}_t$)
    2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
    3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough (that is, less than s)
- If we can do this, clearly we immediately get lower bound on space

The General Proof Strategy [Alekhnovich et al. '00]

- Given formula F and space bound s, want to prove $Sp(F \vdash \bot) \geq s$
- Equivalently, study derivations $\pi = \{\mathbb{P}_0 = \emptyset, \ldots, \mathbb{P}_\tau\}$ s.t. $Sp(\pi) < s$ and show π doesn't derive contradiction (i.e., $1 \notin \mathbb{P}_t$ for all t)
- Begs the question — we don't understand what $\mathbb{P}_t$ looks like!
- Study $\mathbb{P}_t$ by constructing auxiliary configuration $\mathcal{A}_t$ that is easier to understand but gives information about $\mathbb{P}_t$:
  1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
  2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
  3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough (that is, less than s)
- If we can do this, clearly we immediately get lower bound on space

The General Proof Strategy [Alekhnovich et al. '00]

- Given formula F and space bound s, want to prove $Sp(F \vdash \bot) \geq s$
- Equivalently, study derivations $\pi = \{\mathbb{P}_0 = \emptyset, \ldots, \mathbb{P}_\tau\}$ s.t. $Sp(\pi) < s$ and show π doesn't derive contradiction (i.e., $1 \notin \mathbb{P}_t$ for all t)
- Begs the question — we don't understand what $\mathbb{P}_t$ looks like!
- Study $\mathbb{P}_t$ by constructing auxiliary configuration $\mathcal{A}_t$ that is easier to understand but gives information about $\mathbb{P}_t$:
  1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
  2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
  3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough (that is, less than s)
- If we can do this, clearly we immediately get lower bound on space

The General Proof Strategy [Alekhnovich et al. '00]

- Given formula F and space bound s, want to prove $Sp(F \vdash \bot) \geq s$

- Equivalently, study derivations $\pi = \{\mathbb{P}_0 = \emptyset, \ldots, \mathbb{P}_\tau\}$ s.t. $Sp(\pi) < s$ and show π doesn't derive contradiction (i.e., $1 \notin \mathbb{P}_t$ for all t)

- Begs the question — we don't understand what $\mathbb{P}_t$ looks like!

- Study $\mathbb{P}_t$ by constructing auxiliary configuration $\mathcal{A}_t$ that is easier to understand but gives information about $\mathbb{P}_t$:
  1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
  2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
  3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough (that is, less than s)

- If we can do this, clearly we immediately get lower bound on space

The General Proof Strategy [Alekhnovich et al. ’00]

- Given formula F and space bound s, want to prove $Sp(F \vdash \bot) \geq s$
- Equivalently, study derivations $\pi = \{\mathbb{P}_0 = \emptyset, \ldots, \mathbb{P}_\tau\}$ s.t. $Sp(\pi) < s$ and show π doesn’t derive contradiction (i.e., $1 \notin \mathbb{P}_t$ for all t)
- Begs the question — we don’t understand what $\mathbb{P}_t$ looks like!
- Study $\mathbb{P}_t$ by constructing auxiliary configuration $\mathcal{A}_t$ that is easier to understand but gives information about $\mathbb{P}_t$:
  1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ “stronger” than $\mathbb{P}_t$)
  2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
  3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough (that is, less than s)
- If we can do this, clearly we immediately get lower bound on space

Proof of Space Lower Bound for Resolution

Commit to placing pigeons in holes: $C(p,h) = \bigwedge_{i=0}^{\ell-1} x[p,i]^{h_i}$

$\mathcal{A}_t$: collection of such commitments for all pigeons and holes distinct
Satisifiable by construction; maintain invariant $|\mathcal{A}_t| \leq Sp(\mathbb{D}_t)$

Case analysis for derivation step $\mathbb{D}_t \rightsquigarrow \mathbb{D}_{t+1}$:

Download of axiom clause $H(p,q,h)$:

1. If $\mathcal{A}_t \vDash H(p,q,h)$, set $\mathcal{A}_{t+1} = \mathcal{A}_t$
2. Else, $\mathcal{A}_t$ doesn't mention p, say, and by counting
   $\exists\ h' \neq h$ also not mentioned by $\mathcal{A}_t$
   $\Rightarrow$ Set $\mathcal{A}_{t+1} = \mathcal{A}_t \cup \{C(p,h')\}$

Inference of clause D: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t \vDash \mathbb{D}_t \vDash D$
(by soundness of resolution)

Erasure of clause D: **Problem!**
$\mathcal{A}_t \vDash \mathbb{D}_{t+1}$ but $\mathcal{A}_t$ can be too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq Sp(\mathbb{D})$ s.t. $\mathcal{B} \vDash \mathbb{D}$

Proof of Space Lower Bound for Resolution

Commit to placing pigeons in holes: $C(p, h) = \bigwedge_{i=0}^{\ell-1} x[p, i]^{h_i}$

$\mathcal{A}_t$: collection of such commitments for all pigeons and holes distinct
Satisifiable by construction; maintain invariant $|\mathcal{A}_t| \leq Sp(\mathbb{D}_t)$

Case analysis for derivation step $\mathbb{D}_t \rightsquigarrow \mathbb{D}_{t+1}$:

Download of axiom clause $H(p, q, h)$:

1. If $\mathcal{A}_t \vDash H(p, q, h)$, set $\mathcal{A}_{t+1} = \mathcal{A}_t$
2. Else, $\mathcal{A}_t$ doesn't mention p, say, and by counting $\exists\ h' \neq h$ also not mentioned by $\mathcal{A}_t$
   $\Rightarrow$ Set $\mathcal{A}_{t+1} = \mathcal{A}_t \cup \{C(p, h')\}$

Inference of clause D: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t \vDash \mathbb{D}_t \vDash D$ (by soundness of resolution)

Erasure of clause D: **Problem!**
$\mathcal{A}_t \vDash \mathbb{D}_{t+1}$ but $\mathcal{A}_t$ can be too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq Sp(\mathbb{D})$ s.t. $\mathcal{B} \vDash \mathbb{D}$

Proof of Space Lower Bound for Resolution

Commit to placing pigeons in holes: $C(p, h) = \bigwedge_{i=0}^{\ell-1} x[p, i]^{h_i}$

$\mathcal{A}_t$: collection of such commitments for all pigeons and holes distinct
Satisifiable by construction; maintain invariant $|\mathcal{A}_t| \leq Sp(\mathbb{D}_t)$

Case analysis for derivation step $\mathbb{D}_t \rightsquigarrow \mathbb{D}_{t+1}$:

Download of axiom clause $H(p, q, h)$:

1. If $\mathcal{A}_t \vDash H(p, q, h)$, set $\mathcal{A}_{t+1} = \mathcal{A}_t$
2. Else, $\mathcal{A}_t$ doesn't mention p, say, and by counting
   $\exists\ h' \neq h$ also not mentioned by $\mathcal{A}_t$
   $\Rightarrow$ Set $\mathcal{A}_{t+1} = \mathcal{A}_t \cup \{C(p, h')\}$

Inference of clause D: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t \vDash \mathbb{D}_t \vDash D$
(by soundness of resolution)

Erasure of clause D: **Problem!**
$\mathcal{A}_t \vDash \mathbb{D}_{t+1}$ but $\mathcal{A}_t$ can be too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq Sp(\mathbb{D})$ s.t. $\mathcal{B} \vDash \mathbb{D}$

Proof of Space Lower Bound for Resolution

Commit to placing pigeons in holes: $C(p,h) = \bigwedge_{i=0}^{\ell-1} x[p,i]^{h_i}$

$\mathcal{A}_t$: collection of such commitments for all pigeons and holes distinct
Satisifiable by construction; maintain invariant $|\mathcal{A}_t| \leq Sp(\mathbb{D}_t)$

Case analysis for derivation step $\mathbb{D}_t \rightsquigarrow \mathbb{D}_{t+1}$:

Download of axiom clause $H(p,q,h)$:

1. If $\mathcal{A}_t \vDash H(p,q,h)$, set $\mathcal{A}_{t+1} = \mathcal{A}_t$
2. Else, $\mathcal{A}_t$ doesn't mention p, say, and by counting $\exists\, h' \neq h$ also not mentioned by $\mathcal{A}_t$ $\Rightarrow$ Set $\mathcal{A}_{t+1} = \mathcal{A}_t \cup \{C(p,h')\}$

Inference of clause D: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t \vDash \mathbb{D}_t \vDash D$ (by soundness of resolution)

Erasure of clause D: **Problem!**
$\mathcal{A}_t \vDash \mathbb{D}_{t+1}$ but $\mathcal{A}_t$ can be too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq Sp(\mathbb{D})$ s.t. $\mathcal{B} \vDash \mathbb{D}$

Proof of Space Lower Bound for Resolution

Commit to placing pigeons in holes: $C(p,h) = \bigwedge_{i=0}^{\ell-1} x[p,i]^{h_i}$

$\mathcal{A}_t$: collection of such commitments for all pigeons and holes distinct
Satisifiable by construction; maintain invariant $|\mathcal{A}_t| \leq Sp(\mathbb{D}_t)$

Case analysis for derivation step $\mathbb{D}_t \rightsquigarrow \mathbb{D}_{t+1}$:

Download of axiom clause $H(p,q,h)$:

1. If $\mathcal{A}_t \vDash H(p,q,h)$, set $\mathcal{A}_{t+1} = \mathcal{A}_t$
2. Else, $\mathcal{A}_t$ doesn't mention p, say, and by counting
   $\exists\ h' \neq h$ also not mentioned by $\mathcal{A}_t$
   $\Rightarrow$ Set $\mathcal{A}_{t+1} = \mathcal{A}_t \cup \{C(p,h')\}$

Inference of clause D: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t \vDash \mathbb{D}_t \vDash D$
(by soundness of resolution)

Erasure of clause D: **Problem!**
$\mathcal{A}_t \vDash \mathbb{D}_{t+1}$ but $\mathcal{A}_t$ can be too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq Sp(\mathbb{D})$ s.t. $\mathcal{B} \vDash \mathbb{D}$

Proof of Space Lower Bound for Resolution

Commit to placing pigeons in holes: $C(p,h) = \bigwedge_{i=0}^{\ell-1} x[p,i]^{h_i}$

$\mathcal{A}_t$: collection of such commitments for all pigeons and holes distinct

Satisifiable by construction; maintain invariant $|\mathcal{A}_t| \leq Sp(\mathbb{D}_t)$

Case analysis for derivation step $\mathbb{D}_t \rightsquigarrow \mathbb{D}_{t+1}$:

Download of axiom clause $H(p,q,h)$:

1. If $\mathcal{A}_t \vDash H(p,q,h)$, set $\mathcal{A}_{t+1} = \mathcal{A}_t$
2. Else, $\mathcal{A}_t$ doesn't mention p, say, and by counting $\exists\ h' \neq h$ also not mentioned by $\mathcal{A}_t$ $\Rightarrow$ Set $\mathcal{A}_{t+1} = \mathcal{A}_t \cup \{C(p,h')\}$

Inference of clause D: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t \vDash \mathbb{D}_t \vDash D$ (by soundness of resolution)

Erasure of clause D: **Problem!**
$\mathcal{A}_t \vDash \mathbb{D}_{t+1}$ but $\mathcal{A}_t$ can be too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq Sp(\mathbb{D})$ s.t. $\mathcal{B} \vDash \mathbb{D}$

Proof of Space Lower Bound for Resolution

Commit to placing pigeons in holes: $C(p,h) = \bigwedge_{i=0}^{\ell-1} x[p,i]^{h_i}$

$\mathcal{A}_t$: collection of such commitments for all pigeons and holes distinct
Satisifiable by construction; maintain invariant $|\mathcal{A}_t| \leq Sp(\mathbb{D}_t)$

Case analysis for derivation step $\mathbb{D}_t \rightsquigarrow \mathbb{D}_{t+1}$:

Download of axiom clause $H(p,q,h)$:

1. If $\mathcal{A}_t \vDash H(p,q,h)$, set $\mathcal{A}_{t+1} = \mathcal{A}_t$
2. Else, $\mathcal{A}_t$ doesn't mention p, say, and by counting $\exists\ h' \neq h$ also not mentioned by $\mathcal{A}_t$
   $\Rightarrow$ Set $\mathcal{A}_{t+1} = \mathcal{A}_t \cup \{C(p,h')\}$

Inference of clause D: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t \vDash \mathbb{D}_t \vDash D$ (by soundness of resolution)

Erasure of clause D: **Problem!**
$\mathcal{A}_t \vDash \mathbb{D}_{t+1}$ but $\mathcal{A}_t$ can be too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq Sp(\mathbb{D})$ s.t. $\mathcal{B} \vDash \mathbb{D}$

Taking Care of Erasures by Locality Lemma

Lemma (Locality lemma for resolution)

Suppose $\mathcal{A}$ commitment set; $\mathbb{D}$ clause configuration; $\mathcal{A}$ implies $\mathbb{D}$.
Then $\exists$ commitment set $\mathcal{B}$ of size $|\mathcal{B}| \leq Sp(\mathbb{D})$ s.t. $\mathcal{B}$ implies $\mathbb{D}$.

Proof.

Consider bipartite graph with

- clauses $D \in \mathbb{D}$ on left
- commitments $C \in \mathcal{A}$ on right
- edge between D and C if $C \vDash D$ (share a literal)

For every $D \in \mathbb{D}$, pick one neighbour $C \in \mathcal{A}$ (must exist) and let $\mathcal{B}$ be collection of these commitments

Then by construction:

- $|\mathcal{B}| \leq Sp(\mathbb{D})$
- $\mathcal{B} \vDash \mathbb{D}$

□

Taking Care of Erasures by Locality Lemma

Lemma (Locality lemma for resolution)

Suppose $\mathcal{A}$ *commitment set;* $\mathbb{D}$ *clause configuration;* $\mathcal{A}$ *implies* $\mathbb{D}$.
Then $\exists$ *commitment set* $\mathcal{B}$ *of size* $|\mathcal{B}| \leq Sp(\mathbb{D})$ *s.t.* $\mathcal{B}$ *implies* $\mathbb{D}$.

Proof.

Consider bipartite graph with

- clauses $D \in \mathbb{D}$ on left
- commitments $C \in \mathcal{A}$ on right
- edge between D and C if $C \vDash D$ (share a literal)

For every $D \in \mathbb{D}$, pick one neighbour $C \in \mathcal{A}$ (must exist) and let $\mathcal{B}$ be collection of these commitments

Then by construction:

- $|\mathcal{B}| \leq Sp(\mathbb{D})$
- $\mathcal{B} \vDash \mathbb{D}$

□

Taking Care of Erasures by Locality Lemma

Lemma (Locality lemma for resolution)

Suppose $\mathcal{A}$ *commitment set;* $\mathbb{D}$ *clause configuration;* $\mathcal{A}$ *implies* $\mathbb{D}$.
Then $\exists$ *commitment set* $\mathcal{B}$ *of size* $|\mathcal{B}| \leq Sp(\mathbb{D})$ *s.t.* $\mathcal{B}$ *implies* $\mathbb{D}$.

Proof.

Consider bipartite graph with

- clauses $D \in \mathbb{D}$ on left
- commitments $C \in \mathcal{A}$ on right
- edge between D and C if $C \vDash D$ (share a literal)

For every $D \in \mathbb{D}$, pick one neighbour $C \in \mathcal{A}$ (must exist) and let $\mathcal{B}$ be collection of these commitments

Then by construction:

- $|\mathcal{B}| \leq Sp(\mathbb{D})$
- $\mathcal{B} \vDash \mathbb{D}$

□

Taking Care of Erasures by Locality Lemma

Lemma (Locality lemma for resolution)

Suppose $\mathcal{A}$ *commitment set;* $\mathbb{D}$ *clause configuration;* $\mathcal{A}$ *implies* $\mathbb{D}$.
Then $\exists$ *commitment set* $\mathcal{B}$ *of size* $|\mathcal{B}| \leq Sp(\mathbb{D})$ *s.t.* $\mathcal{B}$ *implies* $\mathbb{D}$.

Proof.

Consider bipartite graph with

- clauses $D \in \mathbb{D}$ on left
- commitments $C \in \mathcal{A}$ on right
- edge between D and C if $C \vDash D$ (share a literal)

For every $D \in \mathbb{D}$, pick one neighbour $C \in \mathcal{A}$ (must exist) and let $\mathcal{B}$ be collection of these commitments

Then by construction:

- $|\mathcal{B}| \leq Sp(\mathbb{D})$
- $\mathcal{B} \vDash \mathbb{D}$ □

Outline of PCR Space Lower Bound

For PCR, same high-level approach

Construct auxiliary configuration, or "commitment set," $\mathcal{A}_t$

- highly structured, so easier to understand
- but still gives information about $\mathbb{P}_t$

Want to maintain invariants for $\mathcal{A}_t$:

1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough

But details get much trickier...

Outline of PCR Space Lower Bound

For PCR, same high-level approach

Construct auxiliary configuration, or "commitment set," $\mathcal{A}_t$

- highly structured, so easier to understand
- but still gives information about $\mathbb{P}_t$

Want to maintain invariants for $\mathcal{A}_t$:

1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough

But details get much trickier. . .

Outline of PCR Space Lower Bound

For PCR, same high-level approach

Construct auxiliary configuration, or “commitment set,” $\mathcal{A}_t$

- highly structured, so easier to understand
- but still gives information about $\mathbb{P}_t$

Want to maintain invariants for $\mathcal{A}_t$:

1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ “stronger” than $\mathbb{P}_t$)
2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough

But details get much trickier. . .

Outline of PCR Space Lower Bound

For PCR, same high-level approach

Construct auxiliary configuration, or "commitment set," $\mathcal{A}_t$

- highly structured, so easier to understand
- but still gives information about $\mathbb{P}_t$

Want to maintain invariants for $\mathcal{A}_t$:

1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough

But details get much trickier. . .

Outline of PCR Space Lower Bound

For PCR, same high-level approach

Construct auxiliary configuration, or "commitment set," $\mathcal{A}_t$

- highly structured, so easier to understand
- but still gives information about $\mathbb{P}_t$

Want to maintain invariants for $\mathcal{A}_t$:

1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough

But details get much trickier. . .

Outline of PCR Space Lower Bound

For PCR, same high-level approach

Construct auxiliary configuration, or “commitment set,” $\mathcal{A}_t$

- highly structured, so easier to understand
- but still gives information about $\mathbb{P}_t$

Want to maintain invariants for $\mathcal{A}_t$:

1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ “stronger” than $\mathbb{P}_t$)
2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough

But details get much trickier. . .

Outline of PCR Space Lower Bound

For PCR, same high-level approach

Construct auxiliary configuration, or "commitment set," $\mathcal{A}_t$

- highly structured, so easier to understand
- but still gives information about $\mathbb{P}_t$

Want to maintain invariants for $\mathcal{A}_t$:

1. $\mathcal{A}_t$ implies $\mathbb{P}_t$ (i.e., $\mathcal{A}_t$ "stronger" than $\mathbb{P}_t$)
2. $\mathcal{A}_t$ is satisfiable (so, in particular, $\mathbb{P}_t$ also satisfiable)
3. At derivation step $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$, can do a local update $\mathcal{A}_t \rightsquigarrow \mathcal{A}_{t+1}$ if $Sp(\mathbb{P}_t)$ small enough

But details get much trickier. . .

PCR Space Lower Bound: Commitment Sets

(Disjunctive) commitment

- 2-clause of the form $C = x[p, i]^b \lor x[q, j]^c$
- Pigeons $p \neq q$ distinct
- No restrictions on $i, j \in [0, l)$, $b, c \in \{0, 1\}$
- Domain $\mathrm{dom}(C)$ = set of pigeons $\{p, q\}$ mentioned in C

Commitment set

- $\mathcal{A} = \{C_1, C_2, \ldots, C_s\}$ — think of $\mathcal{A}_t$ as 2-CNF formula
- For all $i \neq j$, $\mathrm{dom}(C_i) \cap \mathrm{dom}(C_j) = \emptyset$
  (i.e., all pigeons mentioned are distinct)
- $\mathrm{dom}(\mathcal{A}) = \bigcup_{C \in \mathcal{A}} \mathrm{dom}(C)$
- Size $|\mathcal{A}|$ = number of commitments in $\mathcal{A}$

PCR Space Lower Bound: Commitment Sets

(Disjunctive) commitment

- 2-clause of the form $C = x[p,i]^b \lor x[q,j]^c$
- Pigeons $p \neq q$ distinct
- No restrictions on $i, j \in [0, l)$, $b, c \in \{0, 1\}$
- Domain $\mathrm{dom}(C)$ = set of pigeons $\{p, q\}$ mentioned in C

Commitment set

- $\mathcal{A} = \{C_1, C_2, \ldots, C_s\}$ — think of $\mathcal{A}_t$ as 2-CNF formula
- For all $i \neq j$, $\mathrm{dom}(C_i) \cap \mathrm{dom}(C_j) = \emptyset$
  (i.e., all pigeons mentioned are distinct)
- $\mathrm{dom}(\mathcal{A}) = \bigcup_{C \in \mathcal{A}} \mathrm{dom}(C)$
- Size $|\mathcal{A}|$ = number of commitments in $\mathcal{A}$

Commitment Sets Implying PC-configurations

Any (total) assignment α to $Vars(BPHP^m_n)$ defines function $f_\alpha : [0,m) \rightarrow [0,n)$ — in what follows, identify α and f_α

A (total) assignment α to $Vars(BPHP^m_n)$ is well-behaved over set of pigeons $S \subseteq [0,m)$ if it sends pigeons in S to distinct holes

An assignment α is well-behaved on and satisfies commitment set $\mathcal{A}$ if

- α well-behaved on dom($\mathcal{A}$)
  (defines partial matching for all pigeons $\mathcal{A}$ mentions)
- α satisfies $\mathcal{A}$

Definition (Entailment)

$\mathcal{A}$ entails PCR-configuration $\mathbb{P}$ over well-behaved assignments if every assignment α which is well-behaved on and satisfies $\mathcal{A}$ must also satisfy $\mathbb{P}$ (i.e., for every polynomial $P \in \mathbb{P}$ have $P(\alpha) = 0$)

Commitment Sets Implying PC-configurations

Any (total) assignment α to $Vars(BPHP^m_n)$ defines function $f_\alpha : [0, m) \to [0, n)$ — in what follows, identify α and f_α

A (total) assignment α to $Vars(BPHP^m_n)$ is well-behaved over set of pigeons $S \subseteq [0, m)$ if it sends pigeons in S to distinct holes

An assignment α is well-behaved on and satisfies commitment set $\mathcal{A}$ if

- α well-behaved on dom($\mathcal{A}$)
  (defines partial matching for all pigeons $\mathcal{A}$ mentions)
- α satisfies $\mathcal{A}$

Definition (Entailment)

$\mathcal{A}$ entails PCR-configuration $\mathbb{P}$ over well-behaved assignments if every assignment α which is well-behaved on and satisfies $\mathcal{A}$ must also satisfy $\mathbb{P}$ (i.e., for every polynomial $P \in \mathbb{P}$ have $P(\alpha) = 0$)

Commitment Sets Implying PC-configurations

Any (total) assignment α to $Vars\big(BPHP_n^m\big)$ defines function $f_\alpha : [0,m) \to [0,n)$ — in what follows, identify α and f_α

A (total) assignment α to $Vars\big(BPHP_n^m\big)$ is well-behaved over set of pigeons $S \subseteq [0,m)$ if it sends pigeons in S to distinct holes

An assignment α is well-behaved on and satisfies commitment set $\mathcal{A}$ if

- α well-behaved on dom($\mathcal{A}$)
  (defines partial matching for all pigeons $\mathcal{A}$ mentions)
- α satisfies $\mathcal{A}$

Definition (Entailment)

$\mathcal{A}$ entails PCR-configuration $\mathbb{P}$ over well-behaved assignments if every assignment α which is well-behaved on and satisfies $\mathcal{A}$ must also satisfy $\mathbb{P}$ (i.e., for every polynomial $P \in \mathbb{P}$ have $P(\alpha) = 0$)

Commitment Sets Implying PC-configurations

Any (total) assignment α to $Vars(BPHP^m_n)$ defines function $f_\alpha : [0, m) \rightarrow [0, n)$ — in what follows, identify α and f_α

A (total) assignment α to $Vars(BPHP^m_n)$ is well-behaved over set of pigeons $S \subseteq [0, m)$ if it sends pigeons in S to distinct holes

An assignment α is well-behaved on and satisfies commitment set $\mathcal{A}$ if

- α well-behaved on dom($\mathcal{A}$) (defines partial matching for all pigeons $\mathcal{A}$ mentions)
- α satisfies $\mathcal{A}$

Definition (Entailment)

$\mathcal{A}$ entails PCR-configuration $\mathbb{P}$ over well-behaved assignments if every assignment α which is well-behaved on and satisfies $\mathcal{A}$ must also satisfy $\mathbb{P}$ (i.e., for every polynomial $P \in \mathbb{P}$ have $P(\alpha) = 0$)

Proof of Space Lower Bound for PCR (Sketch)

Proof invariants:

- $\mathcal{A}_t$ entails $\mathbb{P}_t$ over well-behaved assignments
- $|\mathcal{A}_t| \leq 2 \cdot Sp(\mathbb{P}_t)$

Case analysis for $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$:

Download of axiom clause $H(p,q,h)$:

1. If $\{p,q\} \cap \mathrm{dom}(\mathcal{A}_t) = \emptyset$, add $x[p,0]^{1-h_0} \vee x[q,0]^{1-h_0}$ to $\mathcal{A}_t$
2. If p and/or q in $\mathrm{dom}(\mathcal{A}_t)$, also easy (skip details now)

Works as long as # pigeons not too large

Inference of polynomial P: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t$ entails $\mathbb{P}_t$ and $\mathbb{P}_t \vDash P$ (by soundness of PCR)

Erasure of polynomial P: **Problem!**
$\mathcal{A}_t$ entails $\mathbb{P}_{t+1}$ but $\mathcal{A}_t$ can be much, much too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq 2 \cdot Sp(\mathbb{P})$ s.t. $\mathcal{B}$ entails $\mathbb{P}$

Proof of Space Lower Bound for PCR (Sketch)

Proof invariants:

- $\mathcal{A}_t$ entails $\mathbb{P}_t$ over well-behaved assignments
- $|\mathcal{A}_t| \leq 2 \cdot Sp(\mathbb{P}_t)$

Case analysis for $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$:

Download of axiom clause $H(p, q, h)$:

1. If $\{p, q\} \cap \mathrm{dom}(\mathcal{A}_t) = \emptyset$, add $x[p,0]^{1-h_0} \vee x[q,0]^{1-h_0}$ to $\mathcal{A}_t$
2. If p and/or q in dom($\mathcal{A}_t$), also easy (skip details now)

Works as long as # pigeons not too large

Inference of polynomial P: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t$ entails $\mathbb{P}_t$ and $\mathbb{P}_t \vDash P$ (by soundness of PCR)

Erasure of polynomial P: **Problem!**
$\mathcal{A}_t$ entails $\mathbb{P}_{t+1}$ but $\mathcal{A}_t$ can be much, much too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq 2 \cdot Sp(\mathbb{P})$ s.t. $\mathcal{B}$ entails $\mathbb{P}$

Proof of Space Lower Bound for PCR (Sketch)

Proof invariants:

- $\mathcal{A}_t$ entails $\mathbb{P}_t$ over well-behaved assignments
- $|\mathcal{A}_t| \leq 2 \cdot Sp(\mathbb{P}_t)$

Case analysis for $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$:

Download of axiom clause $H(p, q, h)$:

1. If $\{p, q\} \cap \mathrm{dom}(\mathcal{A}_t) = \emptyset$, add $x[p,0]^{1-h_0} \vee x[q,0]^{1-h_0}$ to $\mathcal{A}_t$
2. If p and/or q in $\mathrm{dom}(\mathcal{A}_t)$, also easy (skip details now)

Works as long as # pigeons not too large

Inference of polynomial P: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t$ entails $\mathbb{P}_t$ and $\mathbb{P}_t \vDash P$ (by soundness of PCR)

Erasure of polynomial P: **Problem!**

$\mathcal{A}_t$ entails $\mathbb{P}_{t+1}$ but $\mathcal{A}_t$ can be much, much too large!

But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq 2 \cdot Sp(\mathbb{P})$ s.t. $\mathcal{B}$ entails $\mathbb{P}$

Proof of Space Lower Bound for PCR (Sketch)

Proof invariants:

- $\mathcal{A}_t$ entails $\mathbb{P}_t$ over well-behaved assignments
- $|\mathcal{A}_t| \leq 2 \cdot Sp(\mathbb{P}_t)$

Case analysis for $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$:

Download of axiom clause $H(p, q, h)$:

1. If $\{p, q\} \cap \mathrm{dom}(\mathcal{A}_t) = \emptyset$, add $x[p,0]^{1-h_0} \vee x[q,0]^{1-h_0}$ to $\mathcal{A}_t$
2. If p and/or q in $\mathrm{dom}(\mathcal{A}_t)$, also easy (skip details now)

Works as long as # pigeons not too large

Inference of polynomial P: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t$ entails $\mathbb{P}_t$ and $\mathbb{P}_t \vDash P$ (by soundness of PCR)

Erasure of polynomial P: **Problem!**
$\mathcal{A}_t$ entails $\mathbb{P}_{t+1}$ but $\mathcal{A}_t$ can be much, much too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq 2 \cdot Sp(\mathbb{P})$ s.t. $\mathcal{B}$ entails $\mathbb{P}$

Proof of Space Lower Bound for PCR (Sketch)

Proof invariants:

- $\mathcal{A}_t$ entails $\mathbb{P}_t$ over well-behaved assignments
- $|\mathcal{A}_t| \leq 2 \cdot Sp(\mathbb{P}_t)$

Case analysis for $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$:

Download of axiom clause $H(p, q, h)$:

1. If $\{p, q\} \cap \mathrm{dom}(\mathcal{A}_t) = \emptyset$, add $x[p,0]^{1-h_0} \vee x[q,0]^{1-h_0}$ to $\mathcal{A}_t$
2. If p and/or q in $\mathrm{dom}(\mathcal{A}_t)$, also easy (skip details now)

Works as long as # pigeons not too large

Inference of polynomial P: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t$ entails $\mathbb{P}_t$ and $\mathbb{P}_t \vDash P$ (by soundness of PCR)

Erasure of polynomial P: **Problem!**

$\mathcal{A}_t$ entails $\mathbb{P}_{t+1}$ but $\mathcal{A}_t$ can be much, much too large!

But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq 2 \cdot Sp(\mathbb{P})$ s.t. $\mathcal{B}$ entails $\mathbb{P}$

Proof of Space Lower Bound for PCR (Sketch)

Proof invariants:

- $\mathcal{A}_t$ entails $\mathbb{P}_t$ over well-behaved assignments
- $|\mathcal{A}_t| \leq 2 \cdot Sp(\mathbb{P}_t)$

Case analysis for $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$:

Download of axiom clause $H(p, q, h)$:

1. If $\{p, q\} \cap \mathrm{dom}(\mathcal{A}_t) = \emptyset$, add $x[p,0]^{1-h_0} \vee x[q,0]^{1-h_0}$ to $\mathcal{A}_t$
2. If p and/or q in $\mathrm{dom}(\mathcal{A}_t)$, also easy (skip details now)

Works as long as # pigeons not too large

Inference of polynomial P: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t$ entails $\mathbb{P}_t$ and $\mathbb{P}_t \vDash P$ (by soundness of PCR)

Erasure of polynomial P: **Problem!**
$\mathcal{A}_t$ entails $\mathbb{P}_{t+1}$ but $\mathcal{A}_t$ can be much, much too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq 2 \cdot Sp(\mathbb{P})$ s.t. $\mathcal{B}$ entails $\mathbb{P}$

Proof of Space Lower Bound for PCR (Sketch)

Proof invariants:

- $\mathcal{A}_t$ entails $\mathbb{P}_t$ over well-behaved assignments
- $|\mathcal{A}_t| \leq 2 \cdot Sp(\mathbb{P}_t)$

Case analysis for $\mathbb{P}_t \rightsquigarrow \mathbb{P}_{t+1}$:

Download of axiom clause $H(p, q, h)$:

1. If $\{p, q\} \cap \mathrm{dom}(\mathcal{A}_t) = \emptyset$, add $x[p,0]^{1-h_0} \vee x[q,0]^{1-h_0}$ to $\mathcal{A}_t$
2. If p and/or q in $\mathrm{dom}(\mathcal{A}_t)$, also easy (skip details now)

Works as long as # pigeons not too large

Inference of polynomial P: Set $\mathcal{A}_{t+1} = \mathcal{A}_t$; OK since $\mathcal{A}_t$ entails $\mathbb{P}_t$ and $\mathbb{P}_t \vDash P$ (by soundness of PCR)

Erasure of polynomial P: **Problem!**
$\mathcal{A}_t$ entails $\mathbb{P}_{t+1}$ but $\mathcal{A}_t$ can be much, much too large!
But can find $\mathcal{B}$ of size $|\mathcal{B}| \leq 2 \cdot Sp(\mathbb{P})$ s.t. $\mathcal{B}$ entails $\mathbb{P}$

Taking Care of Erasures in PCR

Lemma (Locality lemma for PCR)

Suppose

- $\mathcal{A}$ *commitment set*
- $\mathbb{P}$ *PCR-configuration*
- $\mathcal{A}$ *entails* $\mathbb{P}$ *over well-behaved assignments*

Then $\exists$ *commitment set* $\mathcal{B}$ *of size* $|\mathcal{B}| \leq 2 \cdot Sp(\mathbb{P})$ *s.t.* $\mathcal{B}$ *entails* $\mathbb{P}$ *over well-behaved assignments*

This is where the action is...

(But maybe we already had enough action for today)

Taking Care of Erasures in PCR

Lemma (Locality lemma for PCR)

Suppose

- $\mathcal{A}$ *commitment set*
- $\mathbb{P}$ *PCR-configuration*
- $\mathcal{A}$ *entails* $\mathbb{P}$ *over well-behaved assignments*

Then $\exists$ *commitment set* $\mathcal{B}$ *of size* $|\mathcal{B}| \leq 2 \cdot Sp(\mathbb{P})$ *s.t.* $\mathcal{B}$ *entails* $\mathbb{P}$ *over well-behaved assignments*

This is where the action is. . .

(But maybe we already had enough action for today)

Plan for Next Time and Wrapping up This Lecture

Proof of Locality lemma borrows ideas heavily from [Alekhnovich et al. '00]

But also needs some extra twists

Will spend the better part of final lecture proving this lemma
(and filling in other missing details)

To conclude today's lecture, let's discuss some remaining open problems

Separation of PCR and Resolution(?)

“Folklore” result:

Theorem

PCR is exponentially stronger than resolution with respect to proof size

What about space?

Open Problem

Is PCR strictly stronger than resolution with respect to space?
(I.e., when comparing monomial space to clause space)

Would seem likely, somehow. . .

Separation of PCR and Resolution(?)

"Folklore" result:

Theorem

PCR is exponentially stronger than resolution with respect to proof size

What about space?

Open Problem

Is PCR strictly stronger than resolution with respect to space?
(I.e., when comparing monomial space to clause space)

Would seem likely, somehow. . .

Optimal Lower Bounds on PCR Space?

Open Problem

Are random k-CNF formulas hard with respect to space for PCR?

Any other answer than "yes" would be very surprising

(Already known to be exponentially hard w.r.t. size
[Ben-Sasson & Impagliazzo '99, Alekhnovich & Razborov '01])

Open Problem

Are there (k-)CNF formulas F_n of size n such that $Sp_{\mathcal{PCR}}(F \vdash \bot) = \Omega(n)$?

Again, would expect answer to be "yes"

Obvious candidate formula family: random k-CNF formulas

Optimal Lower Bounds on PCR Space?

Open Problem

Are random k-CNF formulas hard with respect to space for PCR?

Any other answer than "yes" would be very surprising

(Already known to be exponentially hard w.r.t. size
[Ben-Sasson & Impagliazzo '99, Alekhnovich & Razborov '01])

Open Problem

Are there (k-)CNF formulas F_n of size n such that $Sp_{\mathcal{PCR}}(F \vdash \perp) = \Omega(n)$?

Again, would expect answer to be "yes"

Obvious candidate formula family: random k-CNF formulas