

# A Tutorial on Proof Complexity

Stephen Cook

BCTCS 2006

April 7, 2006

## Fundamental Problem of Computational Complexity:

**P = NP?**

Other problems:

**LogSPACE  $\subseteq$  P  $\subseteq$  NP  $\subseteq$  PSPACE**

**Easy Theorem: LogSPACE  $\neq$  PSPACE**

Thus some adjacent inclusion above must hold  
(but none is known).

**Embarrassing Ignorance:  $AC^0(6) = PH??$**

## Propositional Proof Complexity

A propositional formula is built from

atoms  $p, q, r, \dots$

connectives  $\wedge, \vee, \neg$

parentheses  $(, )$

A *tautology* is a valid propositional formula.  
(true under all truth assignments to its atoms.)

$$\neg(p \wedge (\neg p \vee q)) \vee q$$

A *proof system* is a polytime map

$$f : \{0, 1\}^* \xrightarrow{\text{onto}} \{\text{tautologies}\}$$

If  $f(x) = A$ , then  $x$  is a *proof* of  $A$ .

**Def'n:** The system is *super* iff for some polynomial  $p(n)$ , every tautology of length  $n$  has a proof of length at most  $p(n)$ .

Recall  $\mathbf{coNP} = \{\bar{L} \mid L \in \mathbf{NP}\}$

$\mathbf{NP} \neq \mathbf{coNP} \rightarrow \mathbf{P} \neq \mathbf{NP}$

Simple Fact:  $\mathbf{NP} = \mathbf{coNP}$  iff there exists a super proof system.

Proof: {tautologies} is complete for  $\mathbf{coNP}$ .

Clearly {tautologies}  $\in \mathbf{NP}$  iff there exists a super proof system.  $\square$

Conjecture:  $\mathbf{NP} \neq \mathbf{coNP}$

(i.e. there is no super proof system).

Complexity theorists believe this, but should they???

Activity: Try to prove specific proof systems are not super.

## Resolution

Refutation system for CNF formulas.

Just one rule:

$$\frac{p \vee A \quad \bar{p} \vee B}{A \vee B}$$

$$(p \vee q) \wedge (\bar{p} \vee r) \wedge (\bar{q} \vee r) \wedge \bar{r}$$

$A$  is unsatisfiable iff the empty clause  $\Lambda$  can be derived from  $A$ .

Resolution can be made into a proof system in our sense:

**Theorem:** Every formula  $A$  can be transformed in polytime to a CNF formula  $A'$  such that  $A$  is a tautology iff  $A'$  is unsatisfiable.

Proof: For each subformula  $B$  of  $A$  introduce a new atom  $p_B$  with defining clauses in  $A'$ .

## Resolution is not super

(Haken, 1985)

### Combinatorial Principle

Pigeonhole Principle: If  $n+1$  pigeons are placed in  $n$  holes, some hole has at least 2 pigeons.

Atoms  $p_{ij}$  (pigeon  $i$  placed in hole  $j$ )

$1 \leq i \leq n+1, 1 \leq j \leq n$   $\neg\text{PHP}_n^{n+1}$  is the conjunction of clauses:

$(p_{i1} \vee \dots \vee p_{in})$  (pigeon  $i$  placed in some hole)  
 $1 \leq i \leq n+1$

$(\neg p_{ik} \vee \neg p_{jk})$  (pigeons  $i, j$  not both in hole  $k$ )  
 $1 \leq i < j \leq n+1, 1 \leq k \leq n$

$\neg\text{PHP}_n^{n+1}$  is unsatisfiable:  $O(n^3)$  clauses

**Theorem** (Haken) Every resolution refutation of  $\neg\text{PHP}_n^{n+1}$  has  $2^{\Omega(n)}$  clauses.

**Theorem**[Chvatal/Szemerédi]:

Random unsatisfiable instances of 3-CNF  
(with a fixed clause/variable ratio)  
almost certainly require exponential size  
resolution refutations.

## **Importance of Resolution Lower Bounds**

Most practical satisfiability testers, when run  
on an unsatisfiable CNF formula, implicitly gen-  
erate a resolution refutation.

Google search: [SAT competitions](#)

**Interpolation:** Given a refutation of  $A(\vec{p}) \wedge B(\vec{q})$

Determine which of  $A(\vec{p}), B(\vec{q})$  is unsatisfiable.

**Simple Fact:** If  $A(\vec{p}, \vec{q}) \rightarrow B(\vec{p}, \vec{r})$  is a tautology, then there is an interpolant  $I(\vec{p})$  such that

$$A(\vec{p}, \vec{q}) \rightarrow I(\vec{p}) \text{ and } I(\vec{p}) \rightarrow B(\vec{p}, \vec{r})$$

are tautologies.

**Theorem** (KPBPR): A resolution refutation  $R$  of

$$A(\vec{p}, \vec{q}) \wedge \neg B(\vec{p}, \vec{r}) \tag{1}$$

can be transformed in polytime to Boolean circuit  $C(\vec{p})$  computing some  $I(\vec{p})$ . Further if  $\vec{p}$  occur only positively in  $A$ , then  $C(\vec{p})$  is monotone.

$C(\vec{p}) = 0$  implies  $A(\vec{p}, \vec{q})$  unsatisfiable

$C(\vec{p}) = 1$  implies  $\neg B(\vec{p}, \vec{r})$  unsatisfiable



**Application:** Fix  $n$  nodes in a graph  $G$ . Let  $\vec{p}$  be variables corresponding to the edge slots in  $G$ . Let

$A_0(\vec{p}, \vec{q})$  assert  $G$  has a  $k$  clique

$A_1(\vec{p}, \vec{r})$  assert  $G$  has a  $k - 1$  co-clique

Then (2) is unsatisfiable, but

**Complexity Theorem:** [Razb, AB, 1985]: Any monotone circuit  $C(\vec{p})$  telling which of  $A_0, A_1$  is unsatisfiable must be exponentially large.

**Corollary:**

$$A_0(\vec{p}, \vec{q}) \wedge A_1(\vec{p}, \vec{r}) \quad (2)$$

requires exponential resolution refutations in this case.

The method of interpolation has been used to prove lower bounds for other proof systems: cutting planes, Nullstellensatz, polynomial calculus.

## Frege Systems

(Hilbert Style Systems)

Standard textbook proof systems

Finitely many axiom schemes and rule schemes.  
Must be implicational complete.

Example for connectives  $\vee, \neg$

Axiom scheme:  $\neg A \vee A$

Rules:

$$\frac{A}{B \vee A} \qquad \frac{A \vee A}{A}$$
$$\frac{(A \vee B) \vee C}{A \vee (B \vee C)} \qquad \frac{A \vee B \quad \neg A \vee C}{B \vee C}$$

All Frege systems p-simulate each other.

Gentzen's propositional LK is p-equivalent to every Frege system.

**Major Open Question:** Are Frege systems super?

**Definition:** A proof system  $F$  admits *uniform feasible interpolation* (UFI) if there is a poly-time algorithm which takes an  $F$ -refutation  $R$  of

$$A_0(\vec{p}, \vec{q}) \wedge A_1(\vec{p}, \vec{r}) \quad (3)$$

together with values for  $\vec{p}$  and determines which of  $A_0, A_1$  is unsatisfiable.

Resolution admits UFI.

**Theorem** (KP, BPR) Frege systems do not admit UFI unless there is a polynomial time algorithm for integer factoring.  
(Same for  $\mathbf{TC}^0$ -Frege.)

**Proof Idea 1** [KP]: Let  $h$  be a one way permutation.

$$h : \{0, 1\}^n \rightarrow \{0, 1\}^n$$

Given  $i$ , define  $A_0, A_1$  by

$A_0(\vec{p}, \vec{q})$  asserts  $h(\vec{q}) = \vec{p}$  and  $q_i = 0$ .

$A_1(\vec{p}, \vec{r})$  asserts  $h(\vec{r}) = \vec{p}$  and  $r_i = 1$ .

$A_0 \wedge A_1$  is unsatisfiable because  $h$  is one-one.

If  $F$  admits UFI and if for each  $i$  we can find a short  $F$ -refutation of (3), we can use these to compute  $h^{-1}(\vec{p})$  bit by bit.

**Proof Idea 2** [BPR]: Apply this to the Diffie-Hellman secret key exchange scheme.

## What combinatorial principles might yield tautologies hard for Frege systems?

NOT the pigeonhole principle (Contrary to my 1970s conjecture)

**Theorem** (Buss87):  $\{PHP_n^{n+1}\}$  have polysize Frege proofs (in fact polysize  $\mathbf{TC}^0$ -Frege proofs).

Proof idea: There are polysize formulas  $A_k(\vec{p}_{ij})$ :

“Pigeons  $1, \dots, k$  occupy at least  $k$  holes”

Prove if no two pigeons occupy same hole, then  $A_1 \rightarrow A_2 \rightarrow \dots \rightarrow A_{n+1}$  to get a contradiction.

[BBP] Suggested possible hard principles for Frege systems, whose proofs seem to require **linear algebra** e.g. the Odd Town Theorem).

Here is another principle:

$$AB = I \supset BA = I \quad (4)$$

where  $A, B$  are  $n \times n$  matrices (say over  $GF(2)$ ).

Product  $AB$  easily expressed with formulas polynomial in  $n$ . But proof of (4) seems to require harder concepts (Gaussian elimination, matrix inverse,...)

**Conjecture** The tautologies (4) do not have polysize Frege proofs.

**Reason** Concepts such as matrix inverse apparently cannot be expressed with polynomial size formulas.

## Complexity Classes [Google: Complexity Zoo]

$$\mathbf{AC}^0 \subset \mathbf{AC}^0(2) \subset \mathbf{TC}^0 \subseteq \mathbf{NC}^1 \subseteq \mathbf{NC}^2 \subseteq \mathbf{P}$$

$\mathbf{AC}^0$  – Polysize family of bounded-depth circuits (unbounded  $\wedge, \vee$ )

$\mathbf{AC}^0(2)$  – Allow parity gates  $p_1 \oplus p_2 \oplus \dots \oplus p_k$  in above circuits.

$\mathbf{TC}^0$  – Allow threshold gates

$\mathbf{NC}^1$  – polynomial formula size (no depth restriction)

$\mathbf{NC}^2$  – polysize  $\log^2$  depth families of Boolean circuits  
(contains matrix inverse, determinant, etc)

$\mathbf{P}$  – polysize families of Boolean circuits (no depth restriction)

(uniform vs nonuniform classes)

## Proof Systems Corresponding to Complexity Classes

(Restrict formulas in a proof corresponding to the class)

**AC<sup>0</sup>-Frege** – Formulas have depth  $\leq d$  (**dAC<sup>0</sup>-Frege**)

**AC<sup>0</sup>(2)-Frege**

**TC<sup>0</sup>-Frege**

**NC<sup>1</sup>-Frege** = Frege

**NC<sup>2</sup>-Frege**

**P-Frege** = **EFrege** (Extended Frege):

Allows introduction of new variables by definition, corresponding to gates in a circuit)



**Major Theorem** (Ajtai; KPWPBI): The Pigeonhole tautologies  $\{PHP_n^{n+1}\}$  require exponential size  $\mathbf{AC}^0$ -Frege proofs.

Thus  $\mathbf{AC}^0$ -Frege is not super.

$\mathbf{AC}^0$ -Frege  $\leq_p$   $\mathbf{AC}^0(2)$ -Frege  $\leq_p$   $\mathbf{TC}^0$ -Frege...

Theorem [B,P,R]  $\{ontoPHP_n^{n+1}\}$  separates  $\mathbf{AC}^0(2)$ -Frege from  $\mathbf{AC}^0$ -Frege.

**Major Open Question** Is  $\mathbf{AC}^0(2)$ -Frege a super proof system?

(We know  $\mathbf{AC}^0 \neq \mathbf{AC}^0(2)$  by [Razborov/Smolensky]: *Parity*  $\neq \mathbf{AC}^0$ .)

## [Soltys 2001]: Thesis on Proof Complexity of Linear Algebra

Consider tautology families corresponding to each of the following matrix identities over  $\text{GF}(2)$ :

$$AB = I \supset BA = I$$

$$(AB = I \wedge AC = I) \supset B = C$$

$$AB = I \supset (AC \neq 0 \vee C = 0)$$

$$AB = I \supset A^t B^t = I$$

For which proof systems do these have polysize proofs?

Facts:

**AC<sup>0</sup>-Frege**: None has polysize proofs. [S,Ur]

**AC<sup>0</sup>(2)-Frege**: They are all equivalent.

**P-Frege (EFrege)**: All have polysize proofs.

**NC<sup>2</sup>-Frege**: **Big Open Question**: They ought to have polysize proofs, but none known.

## Other candidates hard for Frege systems

Partial consistency of EFrege:  $Con_{EF}$  asserts

“  $x$  does not code an EFrege proof of  $p \wedge \neg p$  ”

**Theorem:** [C 75, Buss 91] (a) The  $Con_{EF}$  tautologies have polysize EFrege proofs.  
(b) Frege +  $Con_{EF}$  p-simulate EFrege.

Thus if Frege has uniform polysize proofs of  $Con_{EF}$  then Frege p-simulates EFrege.

**Avigad** has found a combinatorial principle involving DAGs equivalent to  $Con_{EF}$ .

## Candidates hard for EFrege:

Consistency of PA or ZF.

If the latter have polysize Frege (or EFrege) proofs, then these systems are likely super.

## Quantified Propositional Calculus [KP,BK]

A  $\Sigma_1^q$  formula has the form

$$\exists x_1 \dots \exists x_n A(\vec{x}, \vec{p})$$

where  $A(\vec{x}, \vec{p})$  is a propositional formula.

The problem of witnessing quantifiers given a  $\Sigma_1^q$  formula is **NP**-complete.

System  $G_1$  is Gentzen style where lines are sequents

$$A_1, \dots, A_k \rightarrow B_1, \dots, B_m$$

where the A's and B's are  $\Sigma_1^q$  or quantifier-free.

$G_1^*$  (treelike  $G_1$ ) is essentially equivalent to EFrege.

$G_1$  is associated with the complexity class **PLS** (Polynomial Local Search) [JPY].

**Theorem:** (a) The problem: Given a  $G_1^*$  proof  $P$  of a  $\Sigma_1^q$  formula  $\exists \vec{x} A(\vec{x}, \vec{p})$  and values for  $\vec{p}$ , find a witness for  $\vec{x}$ , is complete for polytime.  
(b) The same problem when  $P$  is a  $G_1$  proof is complete for **PLS**.

## Comparing P vs NP with NP vs coNP

Why do we believe that  $P \neq NP$ ?

(1) Computer Scientists are really good at finding algorithms.

(2) Complexity Theorists are really bad at separating complexity classes: e.g.

$$\mathbf{LOGSPACE} \subseteq \mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE}$$

We know  $\mathbf{LOGSPACE} \neq \mathbf{PSPACE}$ , but we cannot prove any of the three inclusions above is proper.

(2a) There is an oracle  $A$  such that  $\mathbf{P}^A = \mathbf{NP}^A$ , so a separation cannot relativize. Also no “Natural Proof” [RR] is likely to show  $\mathbf{NP} \not\subseteq \mathbf{P/poly}$ .

Why do we believe that **NP**  $\neq$  **coNP**?

Perhaps because we can't show they're equal, and item (2) above applies.

**Claim:** Item (1) does not apply.

Suppose Frege systems are super. Assuming **NP**  $\not\subseteq$  **BPP**, any proof would require a non-constructive existence argument. Such proofs are rare. (Even LLL usually yields **BPP** algorithms.)

So maybe **NP** = **coNP** ??

# Conclusion

Bounded Arithmetic

VS

Propositional Proof Systems

VS

Complexity Classes

## EXAMPLE

$\langle \text{VNC}^1, \text{Frege Systems}, \text{NC}^1 \rangle$

What is the complexity of the concepts needed to prove a given theorem.

## LOW LEVEL REVERSE MATHEMATICS

Book in progress:

[Foundations of Proof Complexity](#)

Stephen Cook and Phuong Nguyen

(Download from our web sites)