Lower Bounds in Proof Complexity

Paul Beame

University of Washington

PCMI 2000 Tues-Wed July 24-25

Recall: Frege Systems

- Finite, implicationally complete set **R** of axioms/inference rules

- Refutation version:
  - Proof of unsatisfiability of **F** - sequence $F_1, \ldots, F_r$ of formulas (called lines) s.t.
    - $F_1 = F$
    - each F_j follows from an axiom in **R** or follows from previous ones via an inference rule in **R**
    - $F_r = \Lambda$ trivial falsehood, e.g. $(x \land \neg x)$
- Positive version:
  - Start with nothing, end with tautology **F**

Resolution

- Frege-like system using CNF clauses only
- Start with original input clauses of CNF **F**
- Resolution rule
  - **(A** Ú **x), (B** Ú Ø**x)** **|** **(A** Ú **B)**
- Goal: derive empty clause L

- Most-popular systems for practical theorem-proving

C-Frege proof systems

- Many circuit complexity classes **C** are defined as follows:
  - **C**={**f**: **f** is computed by polynomial-size circuits with structural property P_C}
  - e.g. non-uniform classes **NC^1**, **AC^0**, **$AC^0[p]$**, **ACC**, **TC^0**, **P/poly**

- Define **C-Frege** to be the p-equivalence class of Frege-style proof systems s.t.
  - each line has structural property P_C
  - finite number of axioms/inference rules
  - complete for circuits with property P_C

Circuit Complexity

- **P/poly** - polysize circuits
- **NC^1** - polysize formulas = $O(\log n)$ depth fan-in 2
- **CNF** - polysize CNF formulas
- **AC^0** - constant-depth unbounded fan-in polysize circuits using and/or/not gates
- **$AC^0[m]$** - also $= 0 \bmod m$ tests
- **TC^0** - threshold instead

What we know in circuit complexity

- **$CNF \subset AC^0 \subset AC^0[p] \subset TC^0$** for **$p$** prime
- **$TC^0 \subseteq NC^1 \subseteq P/poly \subseteq NP/poly$**
- **$AC^0[m] \subset \#P$**

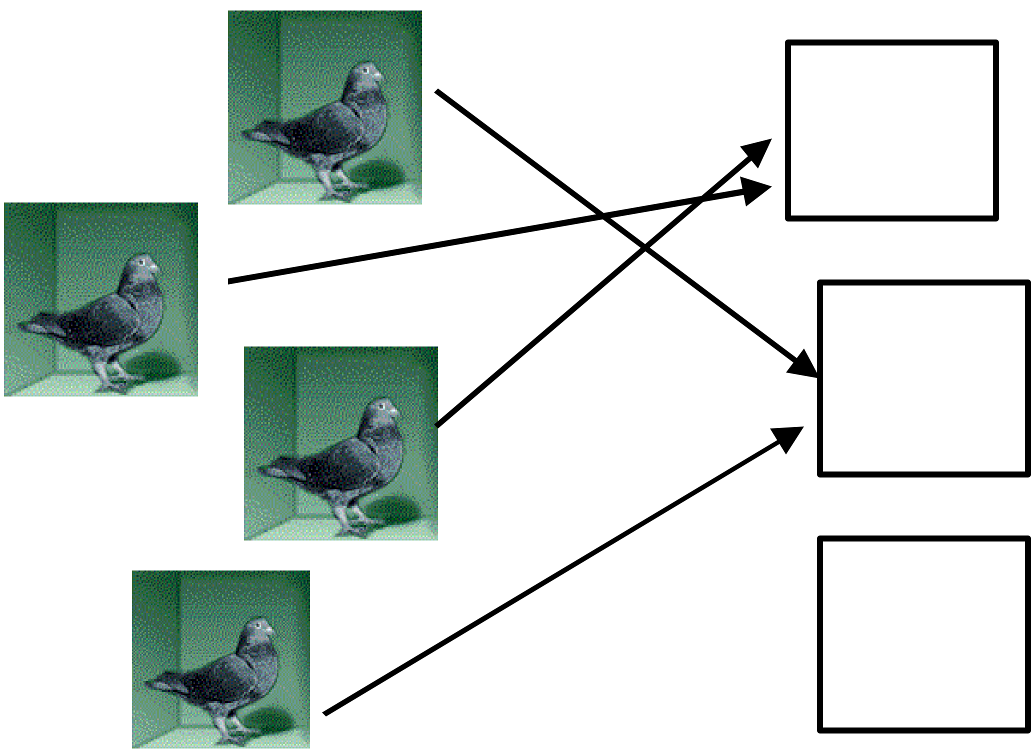
Intuition for hard examples

- A tautology seems likely to be hard to prove in **C-Frege** if the 'natural' proof of it requires concepts that are not computable in circuit complexity class **C**
  - e.g. Majority is not computable in **AC^0[p]** so one might guess something counting-related might be hard for **AC^0[p]-Frege**

- Randomly chosen tautologies/unsatisfiable formulas might be hard to prove because there is no simple good reason to show it.

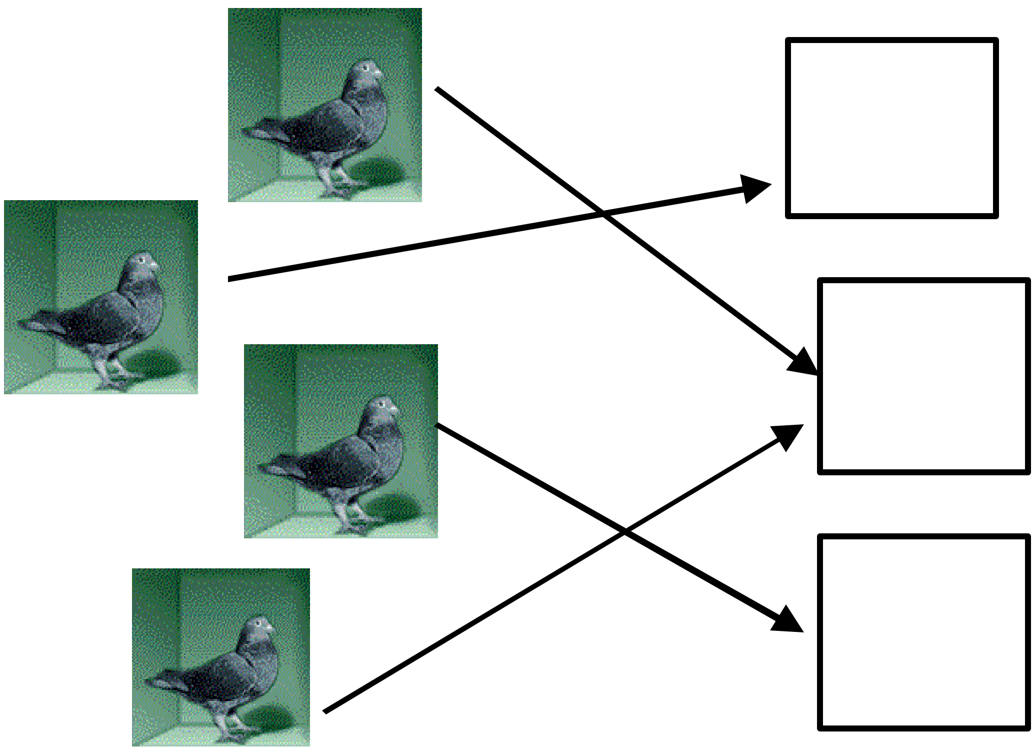
Counting

- Pigeonhole principle **PHP**$^{m \text{®} n}$
  - No **1-1** function from **m** to **n** for **m>n**



Counting

- onto-Pigeonhole principle **ontoPHP**$^{\mathbf{m} \text{®} \mathbf{n}}$
  - No **1-1,onto** function from **m** to **n** for **m>n**



Pigeonhole propositional formulas

Variables

Complete bipartite graph of variables P_{ij} representing $f(i)=j$

Clauses

f is **total**: $(P_{i1} \vee P_{i2} \vee \dots \vee P_{in})$ for $i=1,\dots,m$

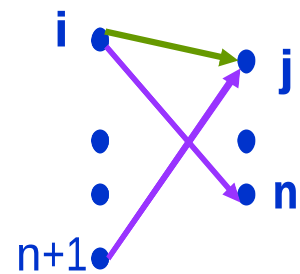
f is **1-1**: (ØP_{ij} Ú ØP_{kj}) for $1 \leq i < k \leq m$, $j=1,\dots,n$

f is **onto**: (P_{1j} Ú P_{2j} Ú... Ú P_{mj}) for $j=1,\dots,n$

f is a **function**: (ØP_{ij} Ú ØP_{ik}) for $i=1,\dots,m$, $1 \leq j < k \leq n$

Note: we usually leave out the **function** clauses.
One can derive the relational form from the functional form by setting $P'_{ij}=P_{ij}$ ÙØP_{i1} Ù ... ÙØ$P_{i(j-1)}$

Usual Proof of PHP$^{n+1\rightarrow n}$



The usual inductive proof of PHP$^{n+1 \circledR n}$

Base: PHP$^{2\rightarrow 1}$ is trivially false

Inductive Step:

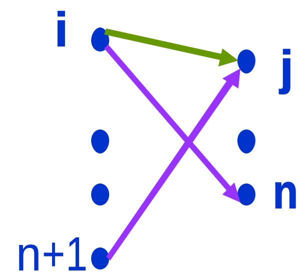
if $f(n+1)=n$ then f on $\{1,\ldots,n\}$ also violates PHP$^{n\rightarrow n-1}$

else define $g:\{1,\ldots,n\} \rightarrow \{1,\ldots, n-1\}$ by

$$g(i)=\begin{cases} f(i) & \text{if } f(i)\neq n \\ f(n+1) & \text{if } f(i)=n \end{cases}$$

g is 1-1/onto **iff** f is

Extended Frege Proof of $PHP^{n+1 \to n}$



The usual inductive proof of $PHP^{n+1 \to n}$

Base: $PHP^{2 \to 1}$ is trivially false

Inductive Step:

if $f(n+1)=n$ then f on $\{1,...,n\}$ also violates $PHP^{n \to n-1}$

else define $g:\{1,...,n\} \to \{1,..., n-1\}$ by

$$g(i)=\begin{cases} f(i) & \text{if } f(i) \neq n \\ f(n+1) & \text{if } f(i)=n \end{cases}$$

g is 1-1/onto **iff** f is

Extended Frege translation: Define new variables

$$Q_{ij} = P_{ij} \vee (\neg P_{(n+1)n} \wedge P_{in} \wedge P_{(n+1)j}) \text{ for } i=1,...n,\ j=1,...,n-1$$

Derive $PHP^{n \to n-1}$ clauses in the Q_{ij} in $O(n^2)$ steps

Cutting Planes Proof of $PHP^{m \to n}$

- Given
  - $P_{i1} + P_{i2} + \ldots + P_{in} \geq 1$ for $i=1,\ldots,m$
  - $P_{ij}+P_{kj} \leq 1$ for $1 \leq i<k \leq m,\ j=1,\ldots,n$
  - $P_{ij} \geq 0$; $P_{ij} \leq 1$ for $i=1,\ldots,m,\ i=1,\ldots,n$
- Derive $P_{1j} + P_{2j} + \ldots + P_{mj} \leq 1$ as follows
  - For $k=3$ to m do
    - Add $(k-2)$ copies of $P_{1j} + P_{2j} + \ldots + P_{(k-1)j} \leq 1$ and one each of $P_{1j}+P_{kj} \leq 1$, ..., $P_{(k-1)j}+P_{kj} \leq 1$ to get $(k-1)P_{1j} + (k-1)P_{2j} + \ldots + (k-1)P_{kj} \leq 2k-3$
    - Apply division rule to get $P_{1j} + P_{2j} + \ldots + P_{kj} \leq 1$
- Compute sum of all P_{ij} in two ways to get $m \leq n$

Resolution and PHP$^{n\to n-1}$

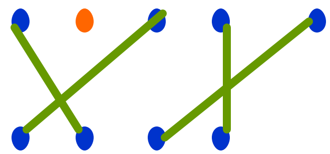
- **Theorem** [Haken 84, Beame-Pitassi 96] Any resolution proof of **PHP$^{n\to n-1}$** requires size at least $2^{n/20}$
  - Applies also to **ontoPHP$^{n\to n-1}$**
- Original proof idea: Bottleneck counting
  - View truth assignments flowing through the proof
    - Assignments start at Λ, flow out towards input clauses
    - A clause in the proof lets only those assignments it **falsifies** flow through it
  - At a 'middle' level in the proof, clauses must talk about lots of pigeons
    - such a clause falsifies few assignments so need lots of them to let all the assignments flow through

Revised proof outline

- $PHP^{n \to n-1}$ lower bound:
  - Show that
    - a partial assignment to the variables, called a restriction can be applied to every small proof so that
      - every large clause disappears and
      - the result is still a $PHP^{n' \to n'-1}$ proof for an good size n'
    - every proof of $PHP^{n' \to n'-1}$ contains a medium complexity clause
    - every medium complexity clause is large

Critical truth assignments for PHP^{n}®$^{n-1}$

- CTAs match all n-1 holes to all but one of the pigeons



  - **1-1, onto clauses (and function clauses) always satisfied**
  - **only input clauses that may not be are clauses**

    C_i=(P_{i1} Ú ... Ú P_{in})

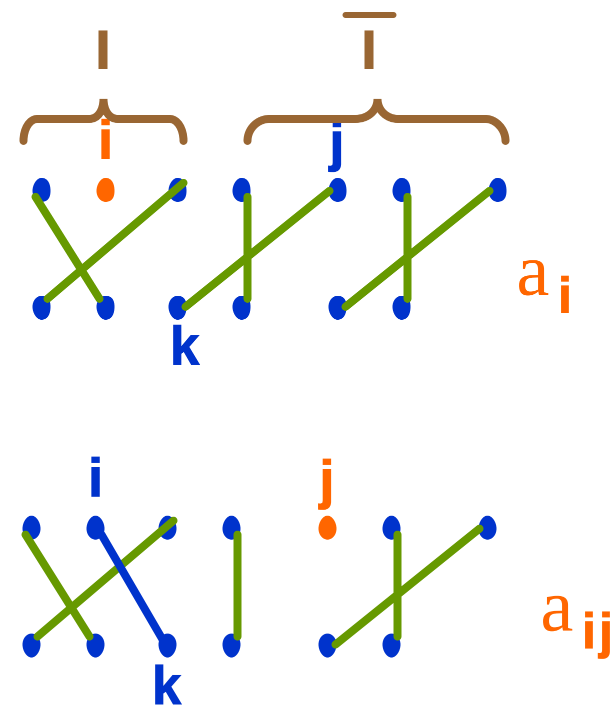
    saying that pigeon i is mapped somewhere
- Modify each of the clauses in the proof
  - **Replace each** ØP_{ij} **by** (P_{1j} Ú ... Ú $P_{(i-1)j}$ Ú $P_{(i+1)j}$ Ú ... Ú P_{nj}) **so all literals are positive**
  - **Lets precisely the same CTAs through**

Any PHP proof has a medium complexity clause

- Given modified clause **C** and **I**$\subseteq\{1,\ldots,n\}$ we say
  - **I** implies **C** iff whenever $\forall i \in I.\ C_i$ is true under some CTA then so is **C**
  - complexity comp(**C**)=**min**{|**I**|: **I** implies **C**}
- Every proof contains a clause of complexity **m** between **n/3** and **2n/3**
  - **Λ has complexity n**
  - **input clauses have complexity $\le$ 1**
  - **if clauses A and B imply C then comp(C) $\le$comp(A)+comp(B)**
  - **walk backwards in proof from Λ, clause complexities decrease but both can't jump over (n/3,2n/3] region**

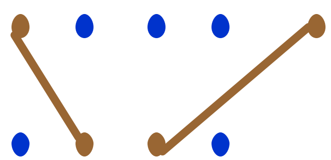
Medium complexity clauses are big

- Suppose I implies C and $|I|=m=\text{comp}(C)$, $n/3 \le m \le 2n/3$
- Since I is minimal, $\forall i \in I$ there is a CTA α_i s.t. $C_i(\alpha_i)=C(\alpha_i)=\text{false}$
- For each $j \notin I$ toggle
  α_i to yield α_{ij}
- Since $C_i(\alpha_{ij})=\text{true}$, $C(\alpha_{ij})=\text{true}$
  thus $P_{ik} \in C$ since it is
  only new true var since α_i
- At least $m(n-m) \ge 2n^2/9$
  total vars in C



Restrictions

- Partial assignments that map certain pigeons to certain holes
  - P_{ij} is set to true and all other P_{ik} or P_{kj} are set to false
  - Reduces $\mathbf{PHP}^{n \rightarrow n-1}$ **to** $\mathbf{PHP}^{n-1 \rightarrow n-2}$
  - More generally, partial matchings



- Restrictions shrink some clauses, satisfy others

Final proof argument

- Call a modified clause **large** iff it has $\geq n^2/10$ vars.
- Assume proof has at most $S < 2^{n/20}$ large clauses.
- On average, restricting a P_{ij} to 1 will satisfy $S/10$ large clauses since large clauses each have $1/10$ of all variables.
- Choose a P_{ij} that satisfies the most large clauses
- Repeat until all large clauses removed:
  - Each time, # of large clauses decreases by a factor of $9/10$
  - Total size of restriction $= \log_{10/9} S < 0.329\ n$
  - Remaining proof proves **PHP** for some n' s.t. $2(n')^2/9 > n^2/10$
  - **Contradiction**

Width of resolution proofs

- If **F** is a set of clauses let
  w(F) = length of longest clause in **F**
- If **P** is a resolution proof
  width(P) = length of longest clause in **P**

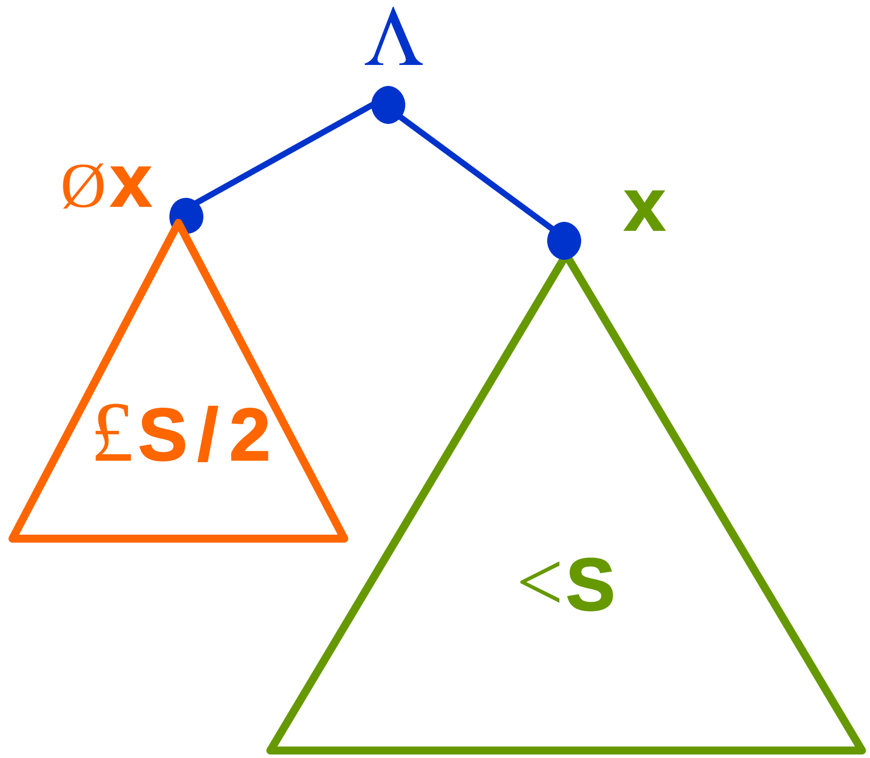
- **Theorem** [BW]: **Every Davis-Putnam (DLL)/tree-like resolution proof of F of size S can be converted to one of width** $\lceil \log_2 \mathbf{S} \rceil + w(\mathbf{F})$

Width of Tree-like Resolution

- Proof: By induction on the size of the proof
  - **Induction Step:**
    Assume that for all sets **F'** of clauses with a tree resolution refutation of size **S'** $<$ **S**, there is a tree-like resolution proof **P'** of **F'** with
    $\mathbf{width(P')} \le \lceil \mathbf{\log_2 S'} \rceil \mathbf{+w(F')}$

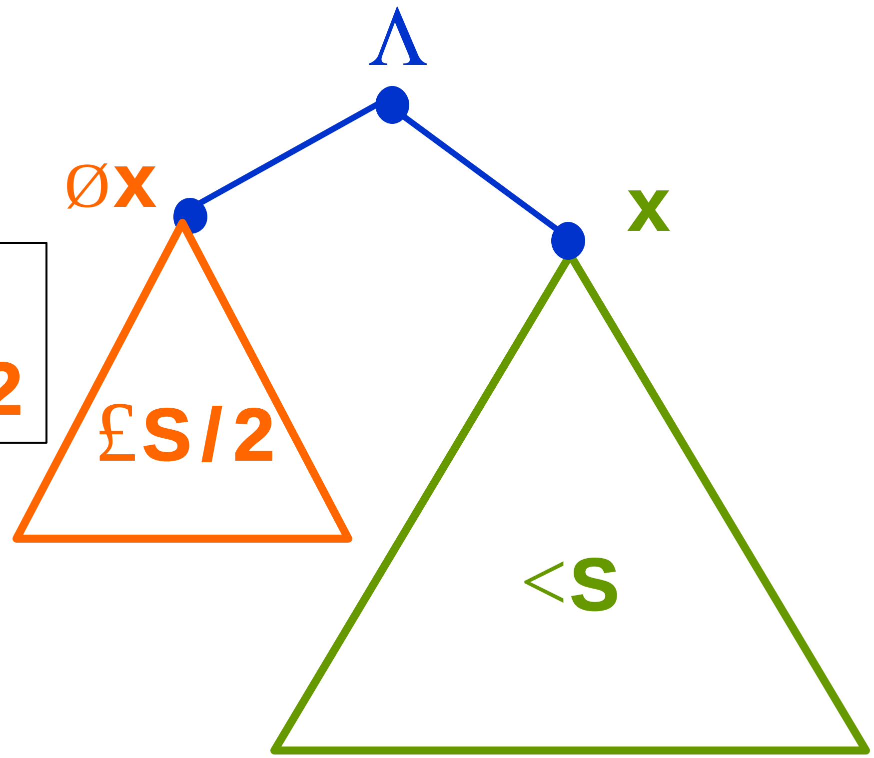
  - Consider a tree resolution refutation of size **S** of a set of clauses **F** and let **x** be the last variable resolved on to derive Λ
  - One of the two subtrees has size at most **S/2** and the other has size strictly smaller than **S**.

Width of Tree-like Resolution

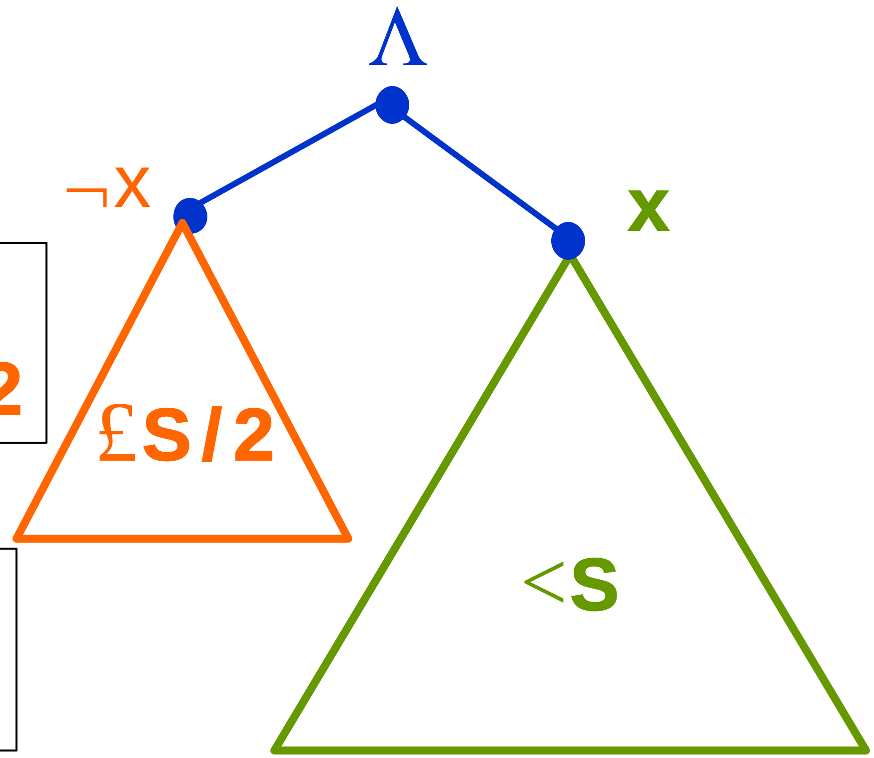


Width of Tree-like Resolution

F proves Ø**x** in size at most **S/2**



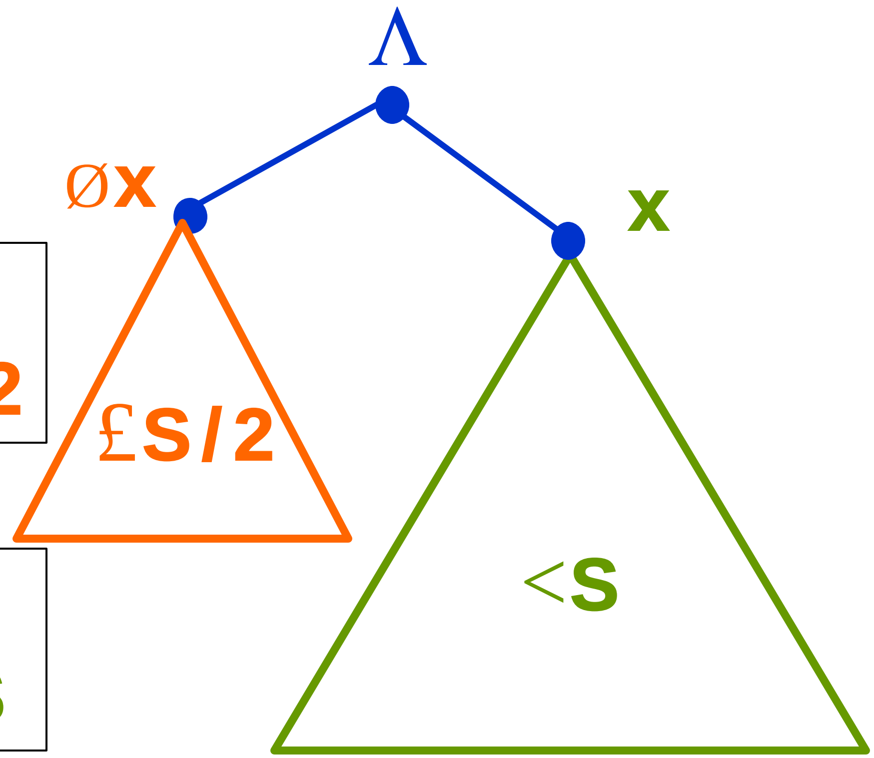
Width of Tree-like Resolution



F proves Ø**x** in size at most **S/2**

F proves **x** in size less than **S**

Width of Tree-like Resolution

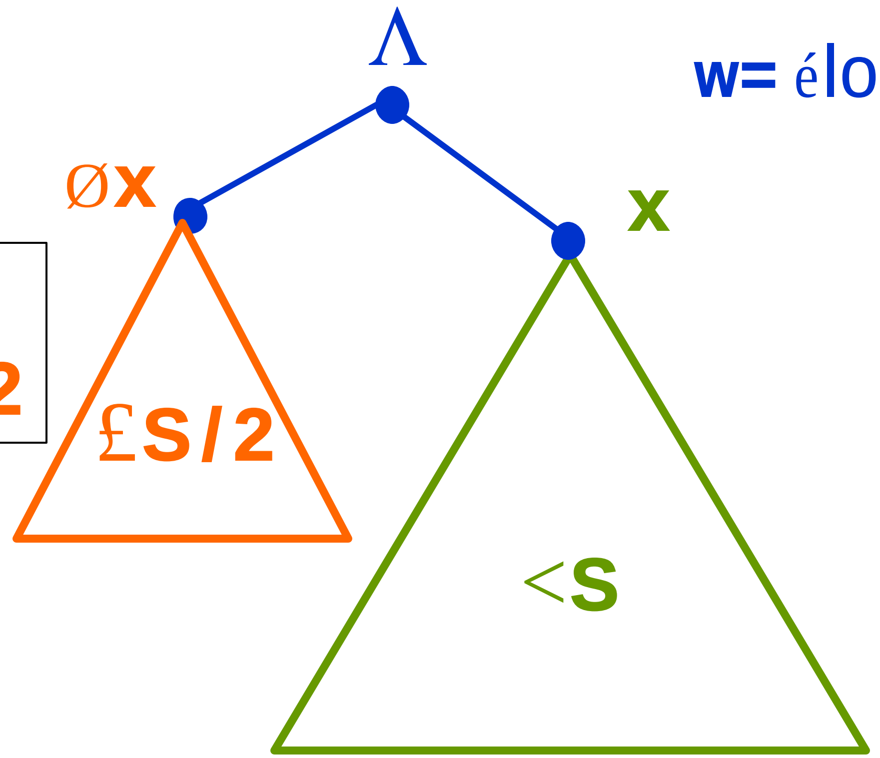


$F_{x \neg 1}$ proves L in size at most **S/2**

$F_{x \neg 0}$ proves L in size less than **S**

Width of Tree-like Resolution

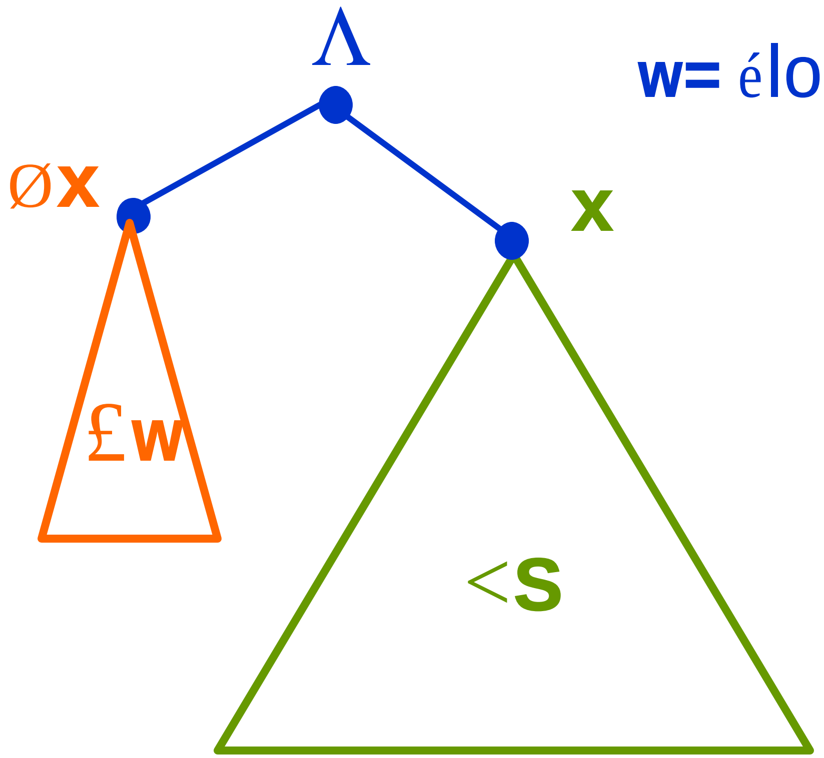
$w = \lceil \log_2 S \rceil + w(F)$



$F_{x \leftarrow 1}$ proves Λ in size at most $S/2$

$F_{x \leftarrow 1}$ proves Λ in width at most $\lceil \log_2 (S/2) \rceil + w(F) = \lceil \log_2 S \rceil + w(F) - 1$

Width of Tree-like Resolution



w= élog$_2$Sù +w(F)

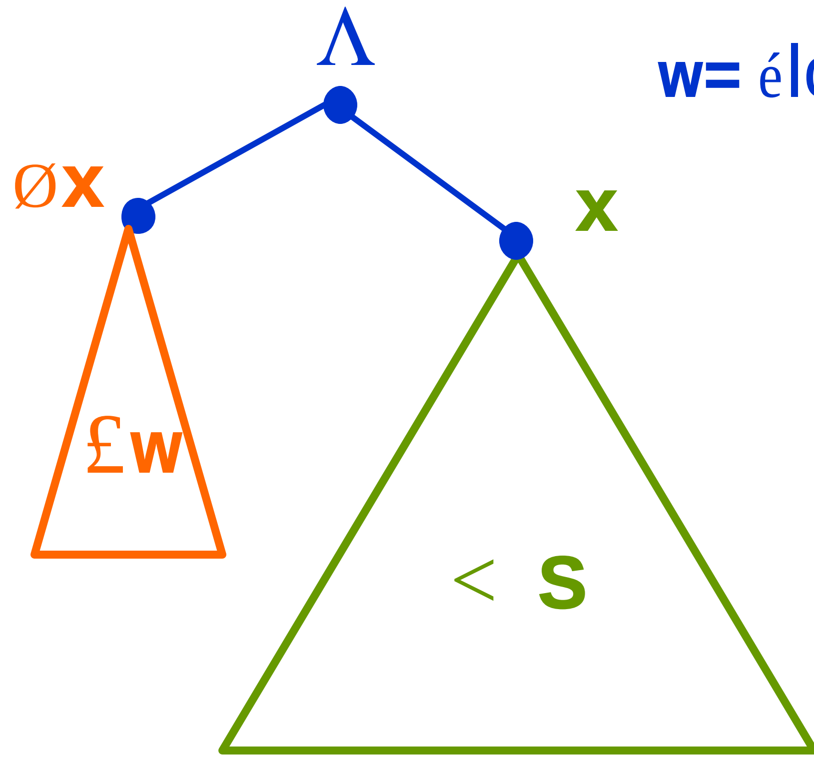
$F_{x\neg 1}$ proves L in size at most S/2

$F_{x\neg 1}$ proves L in width at most
élog$_2$(S/2)ù +w(F)= élog$_2$Sù +w(F)-1

F proves in Øx in width at most élog$_2$Sù+w(F)

Width of Tree-like Resolution

w= é$\log_2$**S**ù +w(F)



$\mathbf{F}_{\mathbf{x}\neg\mathbf{0}}$ proves L in size less than **S**

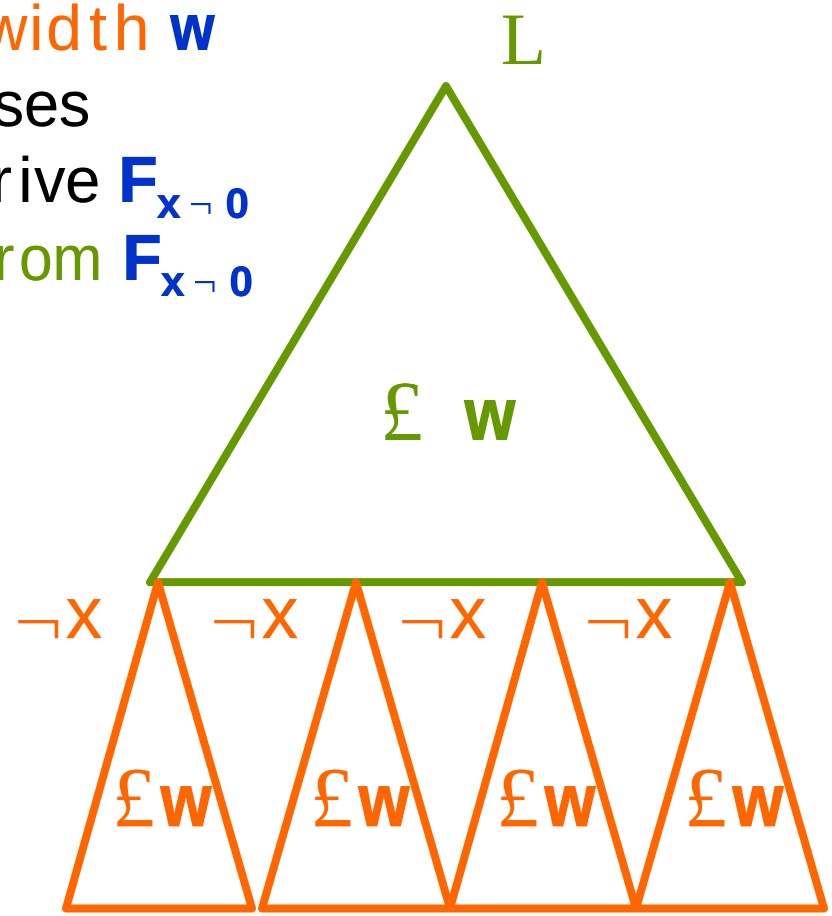
$\mathbf{F}_{\mathbf{x}\neg\mathbf{0}}$ proves L in width at most é$\log_2$**S**ù +w(F)

Width of Tree-like Resolution

w= é$\log_2$**S**ù +w(F)

New Refutation:

1. Derive Ø**x** from **F** in width **w**
2. Resolve Ø**x** with clauses of **F** containing **x** to derive $\mathbf{F}_{\mathbf{x}\,\neg\,\mathbf{0}}$
3. Prove L in width **w** from $\mathbf{F}_{\mathbf{x}\,\neg\,\mathbf{0}}$



Width and Resolution

- **Theorem** [BW] **Every resolution proof of F of size S can be converted to one of width** $O(\sqrt{n \log S}) + w(F)$
- **Proof idea** [CEI] Repeatedly find the most popular literals appearing in large clauses in the proof (like PHP proof)
  - Say a clause is **large** iff it has width ^{3} **W**= $\sqrt{2n \ln S}$
  - There are at most **2n** literals and ^{3} **W** of them per large clause
  - An average literal occurs in ^{3} **W/2n** fraction of large clauses

Proof

- **By induction on n and k: if $(1-W/2n)^k S \le 1$ then any F with at most S large clauses has a proof of width $\le k+w(F)$**
  - Note: W was chosen to be large enough that $(1-W/2n)^W S \le 1$

Proof

- **By induction on n and k: if $(1-W/2n)^k$ S £ 1 then any F with at most S large clauses has a proof of width £ k+w(F)**
    - Note: **W** was chosen to be large enough that $(1-W/2n)^W$ S £ 1
  - Initially at most **S** large clauses
  - Choose literal **x** most frequently occurring in large clauses and set it to **1**, satisfying $\geq$ **(W/2n)** fraction of large clauses
  - Result is a proof of $F_{x \neg 1}$ with £ **S (1-W/2n)** large clauses

Proof

- **By induction on n and k: if $(1-W/2n)^k S \le 1$ then any F with at most S large clauses has a proof of width $\le k+w(F)$**
    - Note: W was chosen to be large enough that $(1-W/2n)^W S \le 1$
  - Initially at most S large clauses
  - Choose literal x most frequently occurring in large clauses and set it to 1, satisfying $\ge (W/2n)$ fraction of large clauses
  - Result is a proof of $F_{x \leftarrow 1}$ with $\le S(1-W/2n)$ large clauses
  - By induction $F_{x \leftarrow 1}$ has a proof of width at most $k-1+w(F)$

Proof

- **By induction on n and k: if $(1-W/2n)^k$ S £ 1 then any F with at most S large clauses has a proof of width £ k+w(F)**
    - Note: **W** was chosen to be large enough that **$(1-W/2n)^W$ S £ 1**
  - Initially at most **S** large clauses
  - Choose literal **x** most frequently occurring in large clauses and set it to **1**, satisfying ≥ **(W/2n)** fraction of large clauses
  - Result is a proof of $F_{x¬1}$ with £ **S (1-W/2n)** large clauses
  - By induction $F_{x¬1}$ has a proof of width at most **k-1 +w(F)**
    - So there is a derivation of Ø**x** from **F** of width **k+w(F)**

Proof

- **By induction on n and k: if (1-W/2n)^{k} S £ 1 then any F with at most S large clauses has a proof of width £ k+w(F)**
    - Note: **W** was chosen to be large enough that **(1-W/2n)^{W} S £ 1**
  - Initially at most **S** large clauses
  - Choose literal **x** most frequently occurring in large clauses and set it to **1**, satisfying ≥ **(W/2n)** fraction of large clauses
  - Result is a proof of **$F_{x \neg 1}$** with £ **S (1-W/2n)** large clauses
  - By induction **$F_{x \neg 1}$** has a proof of width at most **k-1 +w(F)**
    - So there is a derivation of Ø**x** from **F** of width **k+w(F)**
  - By induction there is a proof of **$F_{x \neg 0}$** of width £ **k+w(F)**
    - restrict proof of **F** which has at most **S** large clauses
    - **$F_{x \neg 0}$** has fewer variables

Proof

- **By induction on n and k: if $(1-W/2n)^k S \le 1$ then any F with at most S large clauses has a proof of width $\le k+w(F)$**
    - Note: W was chosen to be large enough that $(1-W/2n)^W S \le 1$
  - Initially at most S large clauses
  - Choose literal x most frequently occurring in large clauses and set it to 1, satisfying $\ge (W/2n)$ fraction of large clauses
  - Result is a proof of $F_{x\leftarrow 1}$ with $\le S(1-W/2n)$ large clauses
  - By induction $F_{x\leftarrow 1}$ has a proof of width at most $k-1+w(F)$
    - So there is a derivation of $\neg x$ from F of width $k+w(F)$
  - By induction there is a proof of $F_{x\leftarrow 0}$ of width $\le k+w(F)$
    - restrict proof of F which has at most S large clauses
    - $F_{x\leftarrow 0}$ has fewer variables
  - New proof: 1) Derive $\neg x$ from F in width $k+w(F)$
    2) Resolve $\neg x$ with F to get $F_{x\leftarrow 0}$ in width $w(F)$
    3) Refute $F_{x\leftarrow 0}$ in width $k+w(F)$

Notes

- Relationship between width and size is roughly optimal for general resolution
  - [Bonet, et al 99] There are tautologies with constant input size and polynomial-size proofs that require width $\Omega(\sqrt{n})$

- Davis-Putnam/DLL/tree-like resolution can require exponentially larger proofs than general resolution [BEGJ 98],[BW 98].
  - Polynomial versus $2^{\Omega(n/\log n)}$ size
  - Uses graph pebbling and width-based lower bound

Width-size relationships

- Let **width(F)** = the minimal width of any resolution proof of **F**

- **Corollary:** Any Davis-Putnam/DLL/tree resolution proof of **F** requires size at least

$$\mathbf{2}^{W(\mathbf{width(F)-w(F)})}$$

- **Corollary:** Any resolution proof of **F** requires size at least

$$\mathbf{2}^{W\left((\text{width}(\mathbf{F})\mathbf{-w}(\mathbf{F}))^{\mathbf{2}}/\mathbf{n}\right)}$$

Resolution lower bound arguments

- $PHP^{n \to n-1}$ lower bound:
  - Show that
    - a restriction can be applied to every small proof so that
      - every large clause disappears and
      - the result is still a $PHP^{n' \to n'-1}$ proof for an good size n'
    - every proof of $PHP^{n' \to n'-1}$ contains a medium complexity clause
    - every medium complexity clause is large
- Width-size relationships:
  - Simply need to show
    - every proof of **F** must contain a large clause relative to # of variables and size of **F**'s input clauses

Minimum unsatisfiable subformula

- **F** - a set of clauses
- **s(F)** - size of minimum subset of **F** that is **unsatisfiable**

Boundary

- **F** - a set of clauses
- **s(F)** - size of minimum subset of **F** that is **unsatisfiable**
- d **F** - **boundary** of **F** - set of variables appearing in exactly one clause of **F**

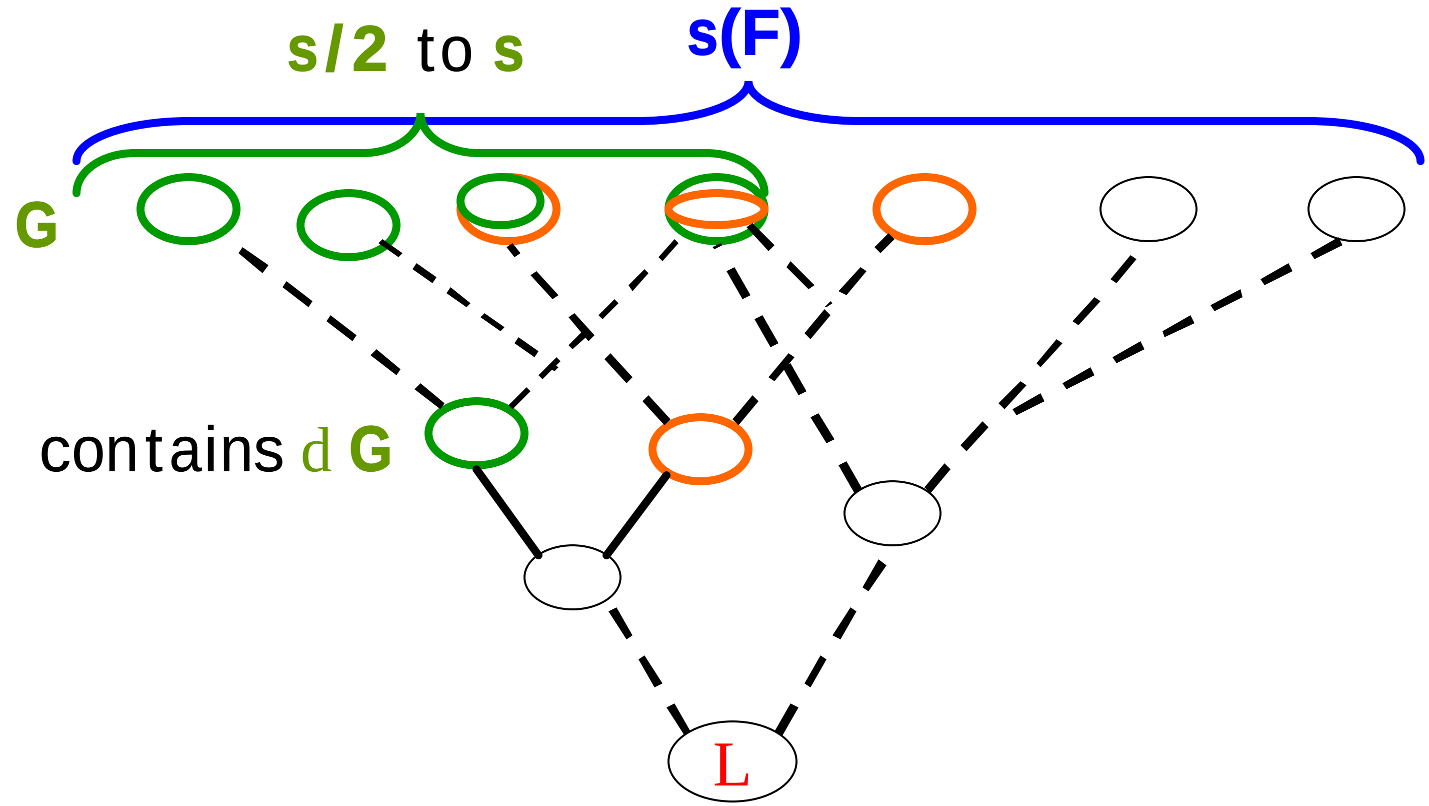
Sub-critical Expansion

- **F** - a set of clauses
- **s(F)** - size of minimum subset of **F** that is **unsatisfiable**
- d **F** - **boundary** of **F** - set of variables appearing in exactly one clause of **F**
- **e(F)** - **sub-critical expansion** of **F** =

$$\max_{s \leq s(F)} \min\{\, |\partial G| : G \subseteq F,\ s/2 \leq |G| < s \,\}$$

Width and expansion

- **Lemma** [CS] : **If P is a resolution proof of F then width(P) $\geq$ e(F).**



Consequences

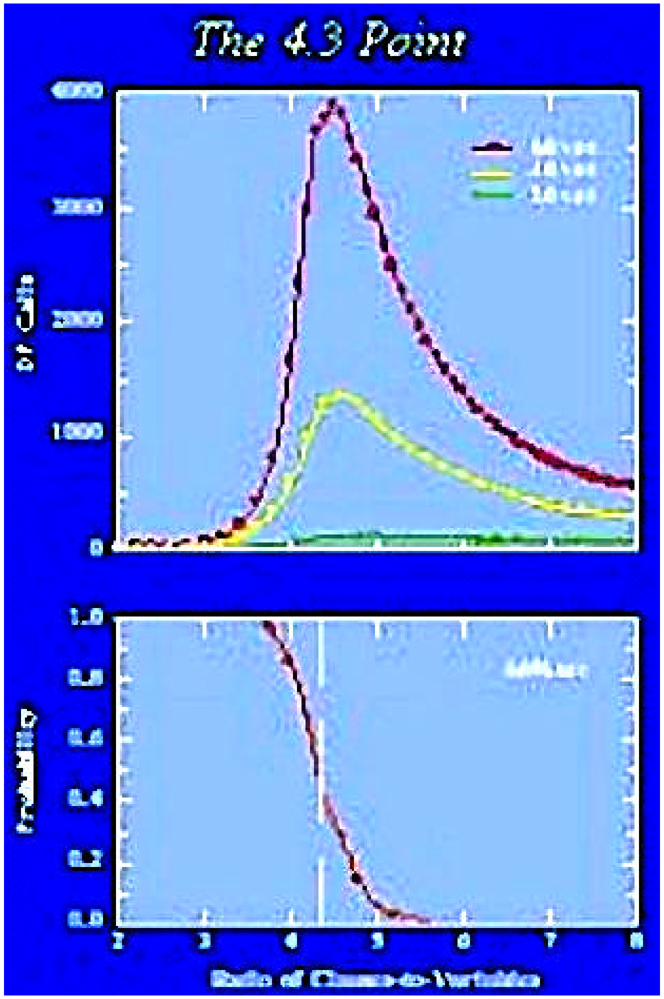
- **Corollaries:**
  - **Any Davis-Putnam (DLL) proof of F requires size at least $2^{e(F)}$**
  - **Any resolution proof of F requires size at least**

$$2^{W\left(e^2(F)/n\right)}$$

Random k-CNF formulas

- Make **m** independent choices of one of the $2^k\binom{n}{k}$ clauses of length **k**
- D **= m/n** is the clause-density of the formula
- Distribution $F^{k}_{n,\Delta}$

Threshold behavior of random k-SAT

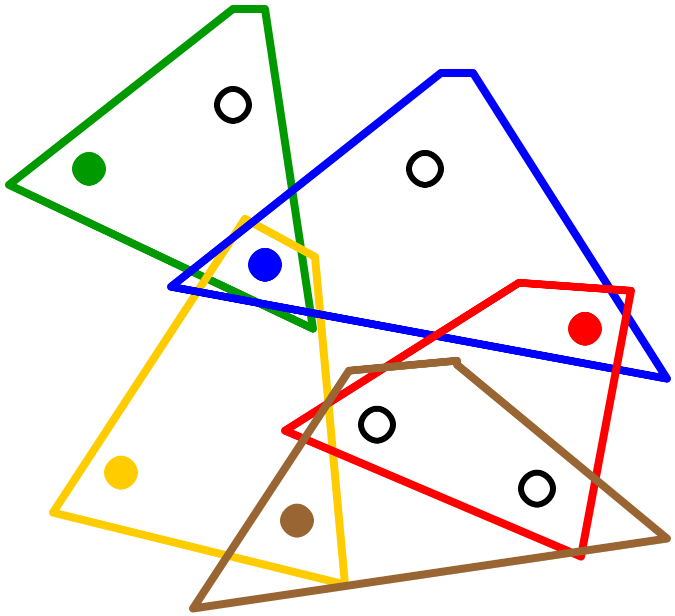


Hypergraph Expansion

- **F** - hypergraph
- ∂ **F** - **boundary** of **F** - set of degree **1** vertices of **F**
- $s_H(F)$ - size of minimum subset of **F** that does not have a **System of Distinct Representatives**
- $e_H(F)$ - **sub-critical expansion** of **F** -

$$\max_{s \le s_H(F)} \min \{ |\delta E| : E \subseteq F,\ s/2 \le |E| < s \}$$

System of Distinct Representatives



variables/nodes

clauses/edges

$s_H(F) \leq s(F)$ so $e_H(F) \leq e(F)$

Density and SDR's

- The **density** of a hypergraph is **#(edges)/#(vertices)**

- **Hall's Theorem:** A hypergraph **F** has a system of distinct representatives iff **every** subgraph has density at most **1**.

s(F) and e(F) for random formulas

- If **F** is a random formula from $\mathbf{F}^{k}_{n,\Delta}$ then
  - **s(F)** is $\Omega\left(n/\Delta^{1/(k-2)}\right)$ almost certainly
  - **e(F)** is $\Omega\left(n/\Delta^{2/(k-2)+\varepsilon}\right)$ almost certainly
- Proved for **Hypergraph** expansion

Density and Boundary

- A **k**-uniform hypergraph of density bounded below **2/k**, say **2/k-**ϵ , has average degree bounded below **2**
  - $\Rightarrow$ constant fraction of nodes are in the boundary

Density of random formulas

- Fix set **S** of vertices/variables of size **r**
  - Probability **p** that a single edge/clause lands in **S** is at most $(r/n)^k$
  - Probability that **S** contains at least **q** edges is at most

$$\Pr\left[B(\Delta n, p) \geq q\right] \leq \left(\frac{e\Delta np}{q}\right)^q \leq \left(\frac{e\Delta r^{k-1}}{n^{k-1}}\right)^q$$

s(F) for random formulas

- Apply for **q=r+1** for all **r** up to **s** using union bound:

$$\sum_{r=k}^{s}\binom{n}{r}\left(\frac{e\Delta r^{k-1}}{n^{k-1}}\right)^{r+1} \le \sum_{r=k}^{s}\left(\frac{ne}{r}\right)^{r}\left(\frac{e\Delta r^{k-1}}{n^{k-1}}\right)^{r+1}$$

$$\le \sum_{r=k}^{s}\frac{r}{en}\left(\frac{e^{2}\Delta r^{k-2}}{n^{k-2}}\right)^{r+1} = o(1)$$

- for $s = O(n/\Delta^{1/(k-2)})$

e(F) for random formulas

- Apply for **q=2r/k** for all **r** between **s/2** and **s** using union bound:

$$\sum_{r=s/2}^{s}\binom{n}{r}\left(\frac{e\Delta r^{k-1}}{n^{k-1}}\right)^{2r/k} \le \sum_{r=s/2}^{s}\left(\frac{ne}{r}\right)^{r}\left(\frac{e\Delta r^{k-1}}{n^{k-1}}\right)^{2r/k}$$

$$\le \sum_{r=s/2}^{s}\left(\frac{e^{1+k/2}\Delta r^{k-1-k/2}}{n^{k-1-k/2}}\right)^{2r/k} = o(1)$$

- for **s = Θ(n/$\Delta^{2/(k-2)}$)**

Lower bounds

- **For random k-CNF chosen from $F^{k}_{n,\Delta}$ almost certainly for any $\epsilon > 0$:**
  - **Any Davis-Putnam proof requires size**

$$2^{n/\Delta^{2/(k-2)+\epsilon}}$$

  - **Any resolution proof requires size**

$$2^{n/\Delta^{4/(k-2)+\epsilon}}$$

A digression: Upper Bound

- **Theorem** [BKPS]: **For F chosen from $F^{k}_{n,\Delta}$ and D above the threshold, the simple Davis-Putnam (DLL) algorithm almost certainly finds a refutation of size**

$$2^{O\left(n/?^{1/(k-2)}\right)} n^{O(1)}$$

- **and this is a tight bound...**

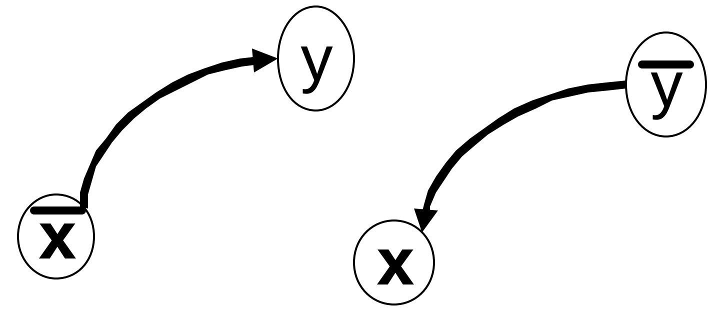
Simple Davis-Putnam Algorithm

- Refute(**F**)
  - While (**F** contains a clause of size **1**)
    - set variable to make that clause true
    - simplify all clauses using this assignment
  - If **F** has no clauses then
    - output "**F** is satisfiable" and HALT
  - If F does not contain an empty clause then
    - Choose smallest-numbered unset variable **x**
    - Run Refute($\mathbf{F}_{\mathbf{x} \neg \mathbf{0}}$)
    - Run Refute($\mathbf{F}_{\mathbf{x} \neg \mathbf{1}}$)

splitting rule

Idea of proof

- 2-clause digraph
  - **$(x \vee y)$**



- Contradictory cycle: contains both **x** and **$\bar{x}$**
- After setting **$O(n/\Delta^{1/(k-2)})$** variables, $\geq 1/2$ the variables are almost certainly in contradictory cycles of the 2-clause digraph
  - a few splitting steps will pick one almost certainly
  - setting clauses of size 1 will finish things off

Implications

- Random k-CNF formulas are **provably** hard for the most common proof search procedures.

- This hardness extends well beyond the phase transition.
  - Even at clause ratio D**$=n^{1/3}$**, current algorithms on random 3-CNF formulas have asymptotically the same running time as the best factoring algorithms.

Random graph k-colorability

- Random graph **G(n,p)** where each edge occurs independently with probability **p**
  - **Sharp threshold** for whether or not graph is **k-colorable, e.g.** **p ~ 4.6/n** for **k=3**

Lower Bound

- **Theorem [BCM 99]: Non-k-colourability** requires exponentially large resolution proofs for random graphs

- Basic proof idea:
  - same outline as before
  - notion of **boundary** of a sub-graph
    - set of vertices of degree $< k$
  - **s(G)** smallest non-k-colourable sub-graph

Nullstellensatz proof system

- Clause $\overbrace{(x_1 \vee \neg x_2 \vee x_3)}^{C}$ becomes equation $\underbrace{(1-x_1)x_2(1-x_3)}_{Q_C}=0$
- Add equations $x_i^2 - x_i = 0$ for each variable
  - Guarantees only 0-1 solutions
- A **proof** is polynomials $P_1, \ldots, P_{m+n}$ proving unsatisfiability: i.e. such that

$$\sum_{j=1}^{m} P_j Q_{C_j} + \sum_{i=1}^{n} P_{m+i}(x^2 - x) \equiv 1$$

Polynomial Calculus

- Similar to Nullstellensatz except:
  - Begin with $Q_1, \ldots, Q_{m+n}$ as before
  - Given polynomials **R** and **S** can infer
    - $a \bullet R + b \bullet S$ for any **a, b** in **K**
    - $x_i \bullet R$
  - Derive constant polynomial **1**
  - Degree = maximum degree of polynomial appearing in the proof
  - Can find proof of **degree d** in time $n^{O(d)}$ using Groebner basis-like algorithm (linear algebra)

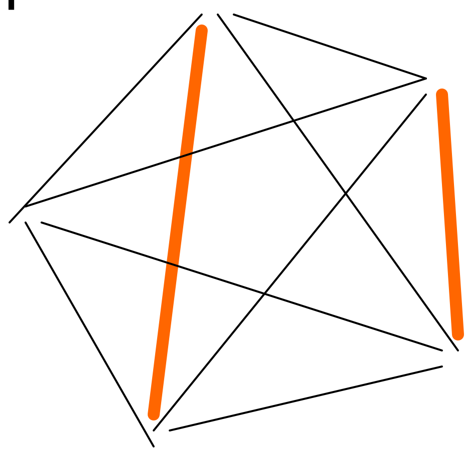
- Special case of **$AC^0[p]$-Frege** if **K=GF(p)** (depth 1)

Natural polynomials for onto$PHP^{m\rightarrow n}$

- f is **total**: $P_{i1}+P_{i2}+...+P_{in}-1=0$ for $i=1,...,m$
- f is **1-1**: $P_{ij}\ P_{kj}=0$ for $1\le i<k\le m$, $j=1,...,n$
- f is **onto**: $P_{1j}+P_{2j}+...+P_{mj}-1=0$ for $j=1,...,n$
  - If $m=n+1$ can simply sum up the **total** polynomials and subtract the **onto** polynomials to get $0=1$, degree 1 Nullstellensatz proof
- Facts:
  - [BR] **If $m=n+p^k$ and $n>p^{2k}$, need degree 2^k Nullstellensatz proofs over $GF(p)$ but easy over $GF(q)$**
  - [R] **Without onto clauses requires PC proofs of degree $n/2$ for any m and any field**

Counting again

- Counting mod 2
  - cannot pair up an odd size set $\mathbf{Count}_2^{2n+1}$



- Counting mod r
  - no perfect r-partition if r doesn't divide n

$$\mathbf{Count}_r^n \quad n \neq 0 \bmod r$$

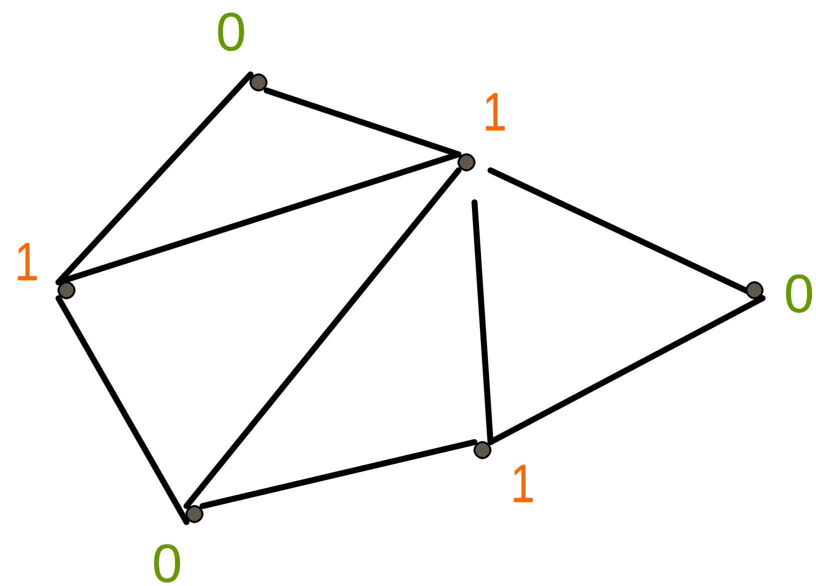
Polynomials for $Count^{m|r}$

- Let $E=[m]^{(r)}$ be the set of all size r subsets of $\{1,...m\}$
  i.e. complete r-uniform hypergraph
- Variables x_e such that $e \in E$
- Equations
  - Every point is **covered:**
    - $1 - \Sigma_{e, i \in e} x_e = 0$ for $i=1,...,m$
  - Edges are **disjoint**:
    - $x_e \, x_f = 0$ for all $e \neq f \in E$ s.t. $e \cap f \neq \emptyset$

- Exercise: $Count^{m|r}$ is easy to refute over Z_r

Tseitin tautologies - odd-charged graphs

- Given a low degree graph **G(V,E)** with 0-1 **charges** on it nodes s.t. total is **odd**



- One variable x_e per edge $e \in E$
  - Clauses saying parity of edges touching **v** is **charge(v)**
  - If degree is large, add extension variables to compute parity at each vertex
- Unsatisfiable

Polynomials in Fourier basis (char(K) $\neq$ 2)

- Interpret atom **x** over **{1,-1}** instead of **{0,1}**; i.e., $y=(-1)^x$
  - linear transform $y=1-2x$
- Variables are {1,-1}
  - $y^2 - 1 = 0$ instead of $x^2 - x = 0$
- Contradiction is **1=-1**
- Convenient for expressing parity
  - $x_1 \oplus \ldots \oplus x_k = 0$ becomes $y_1 y_2 \ldots y_k = 1$

- **Exercise** Since transformation is linear and invertible it preserves degrees of proofs

Tseitin tautologies in Fourier basis

- variables are in $\{1,-1\}$
  - $(y_e)^2 = 1$ for every $e \in E$
- parity of edges equal charge
  - $\prod_{e,v\in e} y_e = (-1)^{\mathrm{charge}(v)}$ for every $v \in V$

- Degree of polynomials equals degree of graph

- **Theorem:** There is a constant degree graph G s.t. a Tseitin tautology for G with all charges 1 requires
  - **degree $\Omega(n)$ to prove in Nullstellensatz** [Grigoriev]
  - **degree $\Omega(n)$ to prove in Polynomial Calculus** [BGIP]

Expander graphs

- **Defn:** Let **G=(V,E)** be a graph. **G** has **expansion** ε iff every subset **S** of $\leq$ **|V|/2** vertices has

  $$|N(S)| \geq (1+\varepsilon)|S|$$

- **Fact:** [Margulis, Gabber-Galil] Constant degree regular bipartite graphs with constant expansion $\varepsilon > 0$ exist.
  - Many applications in complexity
  - Originally considered for regular resolution lower bounds

- Let $E(S) \subseteq E$ be those edges with one endpoint in **S** and one outside **S**. Expansion ε implies $E(S) \geq \varepsilon S > 0$ for all sets **S** of size at most **n/2**.

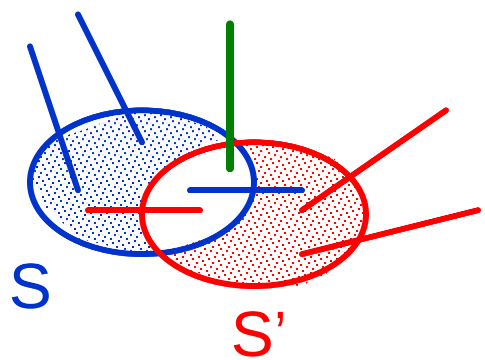
- Degree lower bound is $\varepsilon n/8$

Proof idea: binomial equations

- Every input polynomial has two terms so can think of it as an equivalence for monomials
  - Can one rewrite 1 and -1 to equal each other?
  - Every monomial corresponds to a parity of a subset of edges (and a sign)
  - Each equivalence corresponds to the parity of the set of edges leaving a small non-empty set of vertices
    - **initially just a single vertex v**
  - Might as well think of summation equations mod 2 in the original variables and derive 0=1 rather than use products since they represent the same thing

Parity Reasoning

- Given S, let Σ_S denote the sum of the original edge variables leaving a set S. Every equation is of the form $\Sigma_S = |S| \pmod 2$.
  - Initially $S=\{v\}$ and all charges are 1
  - If we add two equations $\Sigma_S = |S| \pmod 2$ and $\Sigma_{S'} = |S'| \pmod 2$ we get $\Sigma_{S \Delta S'} = |S \Delta S'| \pmod 2$



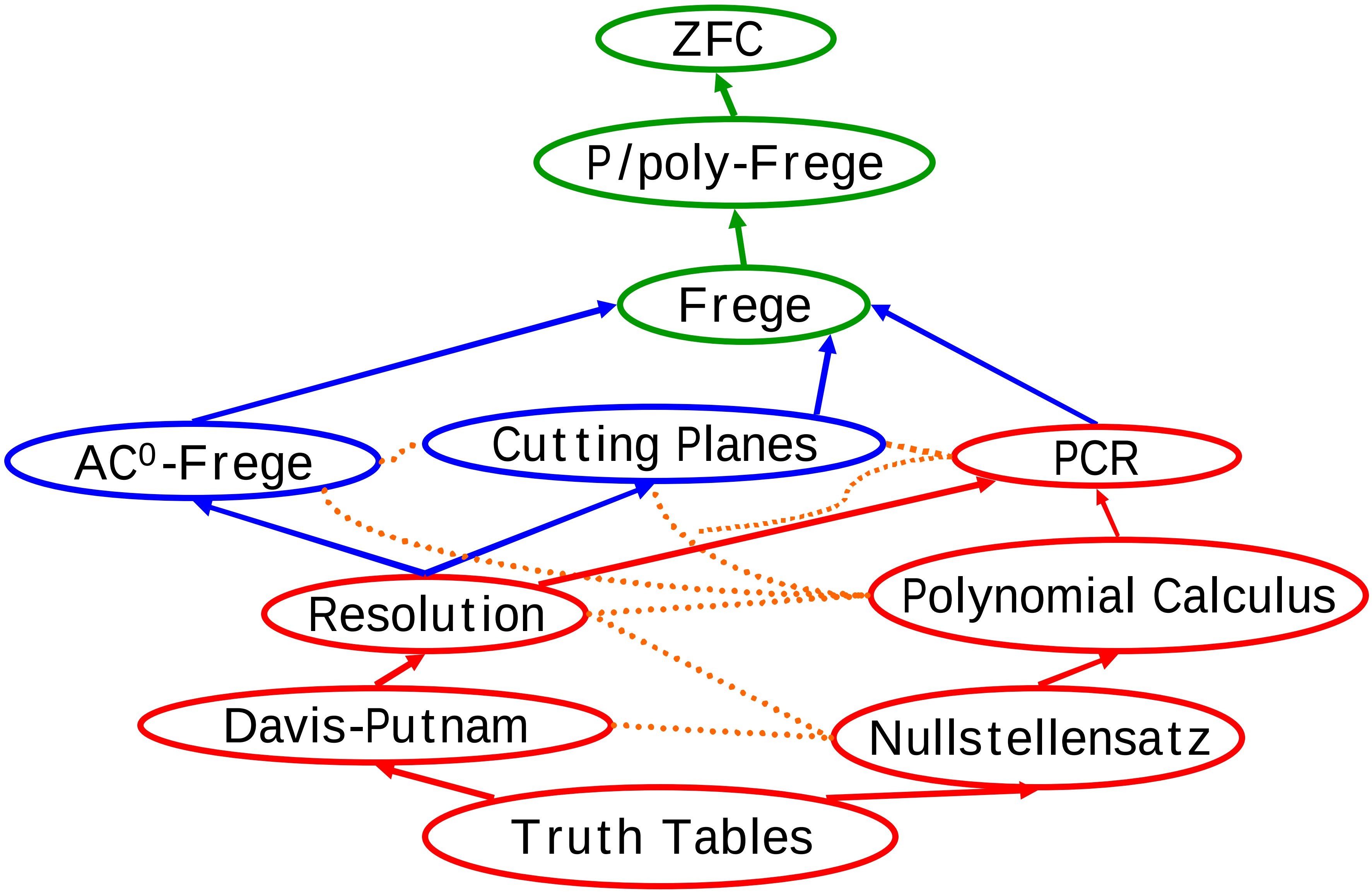
Relation to degree

- No contradiction can be reached if always have $|S\Delta S'|\leq n/2$ since $|E(S\Delta S')|>0$
  - If sets started of size at most **n/4** then this won't happen
  - By expansion, sets of size more than **n/4** have at least ϵ**n/4** edges leaving them so if one is working with sums of fewer than ϵ**n/4** terms one won't see such sets.
  - Each binomial corresponds to a parity summation equation with some portion of the equation in each monomial
    - No contradiction if monomials have degree at most ϵ**n/8**

Implications for $Count^{n|r}$

- Can reduce Tseitin to $Count^{2n+1|2}$
  - Implies $\Omega(n)$ degree lower bounds for $Count^{2n+1|2}$ for all fields K with $char(K) \neq 2$
- Can generalize Tseitin tautologies to arbitrary characteristics $Tseitin(p)$
  - encode in extension fields having p^{th} roots of unity instead of using the Fourier basis
  - similar binomial degree lower bounds if $char(K) \neq p$
- Can reduce $Tseitin(p)$ to $Count^{n|p}$
  - Implies $\Omega(n)$ degree lower bounds for $Count^{pn+1|p}$ for all fields K with $char(K) \neq p$

Some Proof System Relationships



Polynomials in Fourier basis (char(K) $\neq$ 2)

- Interpret atom **x** over **{1,-1}** instead of **{0,1}**; i.e., $y=(-1)^x$
  - linear transform $y=1-2x$
- Variables are {1,-1}
  - $y^2 - 1 = 0$ instead of $x^2 - x = 0$
- Contradiction is $1=-1$
- Convenient for expressing parity
  - $x_1 \oplus \ldots \oplus x_k = 0$ becomes $y_1 y_2 \ldots y_k = 1$

- **Exercise** Since transformation is linear and invertible it preserves degrees of proofs

Binomial equations

- If every input polynomial has two terms so can think of it as an equivalence for monomials
  - $y_{i_1} \ldots y_{i_k} = y_{j_1} \ldots y_{j_l}$ or $y_{i_1} \ldots y_{i_k} = -y_{j_1} \ldots y_{j_l}$
- Might as well think of summation equations mod 2 in the original variables and derive **0=1** rather than use products since they represent the same thing
  - $x_{i_1} + \ldots + x_{i_k} + x_{j_1} + \ldots + x_{j_l} = 0 \pmod{2}$ or
  - $x_{i_1} + \ldots + x_{i_k} + x_{j_1} + \ldots + x_{j_l} = 1 \pmod{2}$

PCR = PC + Resolution

- Two variables x and x' for each atomic proposition x
  - x' stands for $\neg x$
  - include equations $x+x'-1=0$, $x^2-x=0$, and $(x')^2-x'=0$
- Translate $(x_1 \vee \neg x_2 \vee x_3)$ as $(1-x_1)x_2(1-x_3)=0$ or as $x'_1\, x_2\, x'_3=0$
- Same proof rules as polynomial calculus
- **Exercises:**
  - **Show how PCR simulates resolution with degree $\leq$ width and no increase in size**
  - **Show how the resolution relationships between size and width apply to PCR using size and degree**
  - **Binomial equations work just as in PC if char(K) $\neq$ 2**

Hypergraph Expansion

- **F** - hypergraph
- δ **F** - **boundary** of **F** - set of degree **1** vertices of **F**
- $\mathbf{s_H(F)}$ - size of minimum subset of **F** that does not have a **System of Distinct Representatives**
- $\mathbf{e_H(F)}$ - **sub-critical expansion** of **F** -

$$\max_{s \le s_H(F)} \min\{\, |\delta E| : E \subseteq F,\ s/2 \le |E| < s \,\}$$

Hypergraph Expansion and Polynomial Calculus

- **Theorem** [BI]: **The degree of any PCR, polynomial calculus or Nullstellensatz proof of unsatisfiability of F is at least $e_H(F)/2$ if the characteristic is not 2.**

- ⇨ Groebner basis algorithm bound is only $n^{O(e_H(F))}$

k-CNF and parity equations

- Clause $\mathbf{(x_1 \vee \neg x_2 \vee x_3)}$
  is implied by $\mathbf{x_1+(x_2+1)+x_3 = 1 \pmod 2}$
  i.e. $\mathbf{x_1+x_2+x_3 = 0 \pmod 2}$
- Derive contradiction $\mathbf{0 = 1 \pmod 2}$ by adding collections of equations
- # of variables in longest line is at least $\mathbf{e_H(F)}$

Parity equations and polynomial calculus

- Given equations of form
  - $x_1+x_2+x_3 = 0 \pmod 2$
- Represent in the Fourier basis
  - Polynomial equation $y_i^2-1=0$ for each variable
    - $y_i = 1-2x_i$
  - Polynomial equation $y_1 y_2 y_3-1=0$
    - would be $y_1 y_2 y_3+1=0$ if RHS were 1
- Imply the usual equations for original clauses in degree k if $\mathrm{char}(K)$ is not 2

Relationship of equations

- We have 3 forms
  - Original clause $\mathbf{(x_1 \vee \neg x_2 \vee x_3)}$
  - Usual {0,1} polynomials $(1-x_1)x_2(1-x_3)=0,\ x_i^2-x_i=0$
  - **Stronger parity equation** $x_1+x_2+x_3 = 0 \pmod 2$
  - **Fourier basis polynomials** $y_1\, y_2\, y_3-1=0,\ y_i^2-1=0$ where $y_i=1-2x_i$
- $y_i^2-1=0$ and $y_i=1-2x_i$ imply $x_i^2-x_i=0$
- Each equation only involves **k** variables so we use our standard degree upper bound on Nullstellensatz to get usual {0,1} polynomials since the transformed polynomials are stronger

Lower bound

- **For random k-CNF chosen from $F^k_{n,\Delta}$ almost certainly for any $\epsilon>0$:**
  - **Any Nullstellensatz, Polynomial Calculus or PCR refutation over a field K with $\mathrm{char}(K) \neq 2$ requires degree at least**

    $$n/\Delta^{2/(k-2)+\epsilon}$$

    and size at least

    $$2^{c_\epsilon n/\Delta^{4/(k-2)+\epsilon}}$$

Sources

- [Chvatal, Szemeredi 89]
- [Mitchell, Selman, Levesque 93]
- [Clegg, Edmonds, Impagliazzo 96]
- [Beame, Pitassi 97]
- [Razborov 97]
- [Beame, Riis 98]
- [Beame, Karp, Pitassi, Saks 98]
- [Ben-Sasson, Wigderson 99]
- [Ben-Sasson, Impagliazzo 99]
- [Buss, Grigoriev, Impagliazzo, Pitassi 99]
- [Impagliazzo, Pudlak, Sgall 99]
- [Beame, Culberson, Mitchell 00]