Lecture 10

Last time: Cutting Planes stronger than resolution!

(Short proofs of $PHP_n^{n+1}$ - polynomial size and $O(n)$ depth)

$\Rightarrow O(\log n)$

Thm Cutting Planes can efficiently simulate Resolution.

Pf. Reminder: $x_1 \lor \overline{x}_2 \lor x_3 \Rightarrow x_1 + (1-x_2) + x_3 > 1$

Let $F$ be an unsat. CNF formula. Show how to simulate resolution rule with CP rules.

\[
\begin{align*}
  x_1 \lor \overline{x}_2 & \quad x_3 \lor x_2 \quad \Rightarrow \quad x_1 + (1-x_2) > 1 \quad x_3 + x_2 > 1 \\
  x_1 \lor x_3 & \quad \Rightarrow \quad x_1 + x_3 > 1 \quad \checkmark \\
  x_1 \lor \overline{x}_2 & \quad x_2 \lor x_1 \quad \Rightarrow \quad x_1 + (1-x_2) > 1 \quad x_2 + x_1 > 1 \\
  & \quad \Rightarrow \quad 2x_1 > 1 \\
  & \quad \Rightarrow \quad x_1 \geq \lceil \frac{1}{2} \rceil = 1 \quad \checkmark \\
  x_1 \lor \overline{x}_2 & \quad x_2 \lor x_3 \lor x_1 \quad \Rightarrow \quad x_1 + (1-x_2) > 1 \quad x_1 + x_2 + x_3 > 1 \\
  x_1 \lor x_3 & \quad \Rightarrow \quad x_1 + x_3 > 1 \\
  2x_1 + x_3 & \quad \Rightarrow \quad 2x_1 + 2x_3 > 1 \\
  & \quad \Rightarrow \quad x_1 + x_3 \geq 1 \\
\end{align*}
\]

In CP assume $0 \leq x_i \leq 1$ are present for every $i$.

- CP can simulate resolution w/ $O(n)$ overhead
- CP is exp. stronger for some formulas.

Next

Lower bounds for Cutting Planes

Unlike Resolution — there is no known "combinatorial parameter" (like width in resolution) that controls the complexity of CP proofs.

(Vague) Open Problem: Find one? ("Natural generalization of width is not good")

Despite this, we have one proof technique for lower bounds.

Feasible Interpolation

Basic idea: reduce proving lower bounds on proofs to proving lower bounds on computations.

Let $F$ be an unsatisfiable CNF of a particular form:

$$F = A(x,y) \land B(x,z)$$

on three disjoint sets of vars $x, y, z$.

(2)

E.g. Let $x \in \{0,1,2\}$ encodes edges in some n-vertex graph.

$$A(x,y) = 1 \text{ iff } y \text{ encodes a } K\text{-clique in } x.$$  

$$B(x,z) = 1 \text{ iff } z \text{ encodes a proper } (K-1)\text{-colouring of the vertices of } x.$$  

$A \land B$ is unsat!  

(If two vertices $uv$ have $x_{uv}=1$ then colours of $u$ and $v$ are different.)
There is a boolean function \( f_F \) on \( x \)-vars here:

\[
f_F(x) := \begin{cases} 
1 & \text{if } A(x, y) \text{ is satisfiable} \\
0 & \text{o/w}
\end{cases}
\]

Feasible Interpolation

Complexity of refuting \( F = A(x,y) \land B(z, z) \) \( \sim \) Complexity of computing \( f_F \)

Name comes from

Craig Interpolation Theorem \[Craig \ 53\]

If \( \varphi \rightarrow \psi \) is true (\( \varphi, \psi \) are prop. formulas) then there is another formula \( \rho \) s.t.

\[ \varphi \rightarrow \rho \rightarrow \psi \]

and further \( \operatorname{vars}(\rho) \subseteq \operatorname{vars}(\varphi) \cap \operatorname{vars}(\psi) \).

\( f_F \) is just a "\( \rho \)"!

\[ A \land B \text{ unsat } \iff A \lor \overline{B} \text{ is a tautology} \]

\[ \Rightarrow B \Rightarrow A \quad \therefore \quad \emptyset \text{ s.t. } B \Rightarrow \emptyset \Rightarrow A \]

[Krajíček 94] Observed that the proof of Craig's Interpolation gives an efficient alg.

for \( \emptyset \).

Goal:

\[ \text{Thm } [Pudlák \ 97] \quad \text{The Clique-Colouring tautologies require exponential length Cutting planes refutations.} \]
(1) Given a Cutting Planes proof of $F = A(x,y) \land B(x,z)$, extract a real monotone boolean circuit computing the interpolant function $f_F$.

(2) Prove real monotone circuit lower bounds for $f_F$. (will not see)

First prove a "baby version":

**Thm** Let $F = A(x,y) \land B(x,z)$ be an unsatisfiable CNF formula then show the size of the smallest boolean formula computes $f_F$ is at most $O(\text{poly}(S_{\text{Res}}(F)))$.

**Pf.** Uses communication complexity.

**Defn** Let $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$, we consider computing $f$ by the following 2-player game.

2 Players: Alice and Bob — Alice get $x \in \{0,1\}^n$, Bob $y \in \{0,1\}^n$. Their goal is to compute $f(x,y)$.

They do this by means of a communication protocol.

Formally, a protocol $\Pi$ is a tree $T$, where each internal node is labelled with $A$ or $B$, and a function $g : \{0,1\}^n \rightarrow \{0,1\}$. Each outgoing edge is labelled with 0 or 1.
The leaves of the tree will be labelled with bits: \( \{0, 1\} \).

Given input \((x, y)\), the players trace a path down the tree in the natural way: when they reach a node with 'A', Alice sends a bit to Bob and they go to the child. (Same for Bob if node labelled 'B').

Protocol computes \( f \) if the leaf they arrive at is labelled with \( f(x, y) \).

**Defn** The Karchmer-Wigderson game for a boolean function \( h: \{0, 1\}^n \to \{0, 1\} \) is the following 2-player communication problem:

- Alice get \( x \in h^{-1}(1) \) (i.e. \( h(x) = 1 \))
- Bob get \( y \in h^{-1}(0) \) (i.e. \( h(y) = 0 \))

Their goal is to find \( i \in [n] \) s.t. \( x_i \neq y_i \).

The communication complexity of the KW game to be the number of bits that must be communicated in any protocol solving the game.

The communication size complexity of the KW game is the minimum size of any protocol tree solving the game.

**Punchline** communication depth complexity of KW(h) is depth of the shallowest circuit computing \( h \).
communication size complexity of Kw(h)

size of the smallest boolean formula computing h

Communication protocol $\leftrightarrow$ Prover-Delayer game

boolean formula $\leftrightarrow$ Resolution Proof