1 Proof Complexity

We discussed earlier what a proof system is.

Definition 1 A proof system for a nonempty set $T$ of theorems is a computable onto function $f : \Delta^* \rightarrow T$.

Intuitively, $f$ checks an alleged proof. It either outputs the theorem that the proof does prove, or a default element $t_o \in T$.

Next we added a requirement of feasibility:

Definition 2 A feasible proof system for a nonempty set $T$ of theorems is a polytime onto function $f : \Delta^* \rightarrow T$.

This requires that the checking be done in polytime.

If we have such a function $f$, then the proof of a theorem $t$ is the word $w \in \Delta^*$ such that $f(w) = t$. In general, there is no a priori bound on the length of the proof of a theorem. This motivates the following definition:

Definition 3 A polybounded proof system for a nonempty set $T$ of theorems is a polytime onto function $f : \Delta^* \rightarrow T$ where there is a polynomial $p$ such that if $t \in T$ then there is $w \in \Delta^{\leq p(|t|)}$ where $f(w) = t$.

Some reflection shows that this definition is essentially equivalent to the definition of a language being in NP.

Theorem 1 $T$ has a polybounded proof system iff $T$ is in NP.

Corollary 1 VALID has a polybounded proof system iff NP=co-NP.

Because we do not believe that NP=co-NP, we do not believe that VALID has a polybounded proof system. Can we prove it? Note that proving that NP!=co-NP implies that P≠NP, so proving that VALID does not a polybounded proof system is expected to be difficult.

Earlier we also formalized a more specific notion of a “step-by-step” proofs, with respect to a deductive system $\Gamma$, consisting of inference rules.

Definition 4 A $\Gamma$-deduction is a sequence $\varphi_1, ..., \varphi_n$ such that for every $\varphi_i$, there are $i_1 < i_2 < ... < i_{k-1} < i$ such that $(\varphi_{i_1}, \varphi_{i_2}, ..., \varphi_{i_{k-1}}, \varphi_i) \in \Gamma$.

$Deductions(\Gamma) = \{(\varphi_1, ..., \varphi_n) \mid (\varphi_1, ..., \varphi_n) \text{ is a } \Gamma \text{-deduction}\}$
Now we define polynomial feasibility of $\Gamma$ as follows.

Definition 5 $\Gamma$ is polynomially feasible if $\text{Deductions}(\Gamma) \in \text{PTIME}$.

How long is the proof going to be? We want reasonably short proofs.

Definition 6 $\Gamma$ is a polynomially bounded system iff there is polynomial $\rho$ s.t. whenever $\vdash \Gamma \varphi$, there is a deduction $\varphi_1, \ldots, \varphi_n \vdash \varphi$ in $\Gamma$, such that

- $n \leq \rho(|\varphi|)$
- $|\varphi_i| \leq \rho(|\varphi|)$

The definition says that a proof is polynomially bounded iff it has polynomial lines and each line has polynomial length.

We list the following desiderata for $\Gamma$:

1. Soundness and completeness
2. Polynomial feasibility
3. Polynomial boundedness

Theorem 2 If $\Gamma$ is a sound and complete deductive system that is polynomially feasible and bounded, $\text{VALID} \in \text{NP}$.

The intuition behind this theorem is as follows. Suppose we want to know if $\vdash \varphi$ is a tautology. $\vdash \varphi$ iff $\vdash \Gamma \varphi$ means validity is in NP and therefore NP = co-NP. This is because of the relatively short proof. NP = co-NP is widely believed not to be the case.

Corollary 2 We don’t think we can find $\Gamma$, that is sound, complete, polynomially feasible and bounded.

Resolution is sound and complete, but it is not polynomial bounded. Recall that, in Resolution Theorem Proving, we iteratively construct increasingly larger sets of formulas until a contradiction is reached.

\[
\begin{align*}
C_0 &= \varphi \\
C_1 &= R_{p_1}(C_0) \\
C_2 &= R_{p_2}(C_1) \\
&\vdots \\
C_n &= R_{p_n}(C_{n-1})
\end{align*}
\]

While each set $C_i$ is only quadratically bigger than $C_{i-1}$, the overall growth can be exponential. Put otherwise, the length of a resolution refutation can be
exponential. We now focus on the length of the refutations. We will give an example formula whose refutation is exponential.

Recall the Pigeon-Hole Principle: If we have \( f : \{1, \ldots, n\} \mapsto \{1, \ldots, n-1\} \), then for some \( 1 \leq i \leq j \leq n \) we have \( f(i) = f(j) \). It can be written as: \( R \subseteq \{1, \ldots, n\} \times \{1, \ldots, n-1\} \), for every pigeon \( i \) there is some hole \( k \) such that \( R(i, k) \). There exists \( 1 \leq i < j \leq n \) and \( 1 \leq k \leq n-1 \), such that both \( R(i, k) \) and \( R(j, k) \) hold.

We formulate this (for a fixed \( n \in \mathbb{N} \)) in propositional logic. To do so, we will use the following atomic propositions, for each \( i, j \) such that \( 1 \leq i \leq j \leq n \):

\[
P_{ij} \overset{\text{def}}{=} R(i, j)
\]

\[
PHP_n \left( \bigwedge_{1 \leq i \leq n} \bigvee_{1 \leq k \leq n-1} P_{ik} \right) \rightarrow \left( \bigvee_{1 \leq i < j \leq n} \bigvee_{1 \leq k < n} (P_{ik} \land P_{jk}) \right)
\]

So, this formula is valid iff the Pigeon-Hole Principle (for a fixed \( n \)) holds.

**Theorem 3 (Haken, 1975)*** Every resolution refutation of \( \neg PHP_n \) has length \( 2^{\Omega(n)} \).

So, Resolution Theorem Proving is not polynomially-bounded: there is no polynomial function of the length of a theorem that places an upper bound on the length of the smallest proof of that theorem.

Conceivably, resolution have long refutations because it is such a weak system. So an active research area is establishing lower bounds for more powerful proof systems.