Algebraic Proof Systems

Pavel Pudlák

Mathematical Institute, Academy of Sciences, Prague
and
Charles University, Prague

Fall School of Logic, Prague, 2009

Overview

1. a survey of proof systems
2. a lower bound for an algebraic proof system
3. on lower bounds for ILP proof systems

Propositional Proof Systems

The idea of a general propositional proof system

1. it is sound;
2. it is complete;
3. the relation *'D is a proof of tautology* ϕ*"* is decidable in *polynomial time.*

Definition (Cook, 1975)

Let *TAUT* be a set of tautologies. A *proof system* for *TAUT* is any polynomial time computable function f that maps the set of all binary strings $\{0,1\}^*$ onto *TAUT*.

Meaning:
Every string is a proof.
$f(\bar{a})$ is the formula of which $\bar{a}$ is a proof.

Propositional Proof Systems

The idea of a general propositional proof system

1. it is sound;
2. it is complete;
3. the relation *'D is a proof of tautology* ϕ*"* is decidable in *polynomial time.*

Definition (Cook, 1975)

Let *TAUT* be a set of tautologies. A *proof system* for *TAUT* is any polynomial time computable function f that maps the set of all binary strings $\{0,1\}^*$ onto *TAUT*.

Meaning:
Every string is a proof.
$f(\bar{a})$ is the formula of which $\bar{a}$ is a proof.

Say that *a proof system is polynomially bounded*, if every tautology has a proof of polynomial length.

Fact

There exists a polynomially bounded proof system iff $\mathbf{NP} = \mathbf{coNP}$*.*

Meaning:
Every string is a proof.
$f(\bar{a})$ is the formula of which $\bar{a}$ is a proof.

Definition

A proof system f_1 *polynomially simulates* a proof system f_2, if there exists a polynomial time computable function g such that for all $\bar{a} \in \{0,1\}^*$, $f_1(g(\bar{a})) = f_2(\bar{a})$.

Meaning:
Given a proof $\bar{a}$ of $f_2(\bar{a})$ in the second system, we can construct a proof $g(\bar{a})$ of the same tautology in the first system in polynomial time.

Frege Proof Systems

propositional variables $p_1, p_2, \ldots$
any complete finite set of connectives.
any complete finite set of rules.
a Frege proof is a string of formulas (tautologies) that are axioms or derived from previous ones using rules

Example. [Hilbert and Ackermann]
Connectives $\neg, \vee$.
Axiom schemas

1. $\neg(A \vee A) \vee A$
2. $\neg A \vee (A \vee B)$
3. $\neg(A \vee B) \vee (B \vee A)$
4. $\neg(\neg A \vee B) \vee (\neg(C \vee A) \vee (C \vee B))$

Rule

- *From A and $\neg A \vee B$ derive B.*

Frege Proof Systems

propositional variables $p_1, p_2, \ldots$
any complete finite set of connectives.
any complete finite set of rules.
a Frege proof is a string of formulas (tautologies) that are axioms or derived from previous ones using rules

Example. [Hilbert and Ackermann]
Connectives $\neg, \vee$.
Axiom schemas

1. $\neg(A \vee A) \vee A$
2. $\neg A \vee (A \vee B)$
3. $\neg(A \vee B) \vee (B \vee A)$
4. $\neg(\neg A \vee B) \vee (\neg(C \vee A) \vee (C \vee B))$

Rule

- *From A and $\neg A \vee B$ derive B.*

Theorem (Cook-Reckhow)

Frege systems polynomially simulate each other.

Lower bounds for prop. proof systems

Exponential lower bounds imply:

- separations of some fragments of bounded arithmetic,
- impossibility of efficient algorithms of certain types,
- exp. lower bounds on all systems would prove $\mathbf{NP} \neq \mathbf{coNP}$.

But we are able to prove lower bounds only for very restricted subsystems of Frege proofs: where the depth of all formulas in the proof is bounded by a constant, *Bounded Depth Frege proof systems*.

Algebraic proof systems

Two types

1. proving unsolvability of systems of equations
2. proving polynomial identities

F a field
$F[x_1, \ldots, x_n]$ the ring of polynomials
algebraic circuits

Proving unsolvablity of equations

Theorem (Hilbert's Nullstellensatz)

A system of equations

$$f_1(x_1, \ldots, x_n) = 0, \ldots, f_m(x_1, \ldots, x_n) = 0$$

does not have a solution in the algebraic closure of F, iff there exist polynomials $g_1, \ldots, g_m$ such that

$$\sum_{i=1}^{m} f_i g_i = 1.$$

Note that

1. the "if" part is trivial;
2. the condition is equivalent to:
   polynomials $f_1, \ldots, f_m$ generate the ideal of all polynomials.

Nullstellensatz as a proof system

Call $(g_1, \ldots, g_m)$ such that $\sum_{i=1}^{m} f_i g_i = 1$ a proof of the unsolvability of $f_1 = 0, \ldots, f_m = 0$.

Measures of the complexity of such a proof:

1. $\max_i \deg g_i$;
2. the number of monomials in $g_1, \ldots, g_m$;
3. the size of formulas/circuits computing $g_1, \ldots, g_m$.

Nullstellensatz as a proof system

Call $(g_1, \ldots, g_m)$ such that $\sum_{i=1}^{m} f_i g_i = 1$ a proof of the unsolvability of $f_1 = 0, \ldots, f_m = 0$.

Measures of the complexity of such a proof:

1. $\max_i \deg g_i$;
2. the number of monomials in $g_1, \ldots, g_m$;
3. the size of formulas/circuits computing $g_1, \ldots, g_m$.

What if we are only interested in 0-1 solutions? Add equations

$$x_1^2 = x_1, \ldots, x_m^2 = x_m.$$

Such a proof system is a propositional proof system.

Polynomial Calculus

We can derive $\sum_{i=1}^{m} f_i g_i = 1$ sequentially.
Recall that $\emptyset \neq I \subseteq F[x_1, \ldots, x_n]$ is an ideal, iff

1. $g, h \in I \Rightarrow g + h \in I$ and
2. $g \in F[x_1, \ldots, x_n] \wedge h \in I \Rightarrow gh \in I$.

Polynomial Calculus

We can derive $\sum_{i=1}^{m} f_i g_i = 1$ sequentially.
Recall that $\emptyset \neq I \subseteq F[x_1, \ldots, x_n]$ is an ideal, iff

1. $g, h \in I \Rightarrow g + h \in I$ and
2. $g \in F[x_1, \ldots, x_n] \wedge h \in I \Rightarrow gh \in I$.

The rules of the Polynomial Calculus

1. from g and h derive $g + h$
2. from h derive gh (where g is any polynomial)

A proof is

$$(f_1, \ldots, f_m, h_1, \ldots, h_\ell, 1),$$

where ...

Proving equations - equational calculus

Axioms

1. $x = x$;
2. 0 is zero, 1 is one, associativity and commutativity of $\times$ and $+$, distributivity.

Rules

1. reflexivity of $=$;
2. $=$ is a congruence reaction w.r.t. $+$ and $\times$.

(Horn formulas translate into rules.)

ILP proof systems

Integer Linear Programing problem is given by

- a rational matrix $\{a_{ij}\}$ and
- a rational vector $\vec{B}$.

The task is to find an integral solution to the set of inequalities inequalities (or to determine if it exists)

$$\sum_j a_{ij}x_j \leq B_i$$

Fact

The decision version of ILP is **NP***-complete.*

ILP proof systems

Integer Linear Programing problem is given by

- a rational matrix $\{a_{ij}\}$ and
- a rational vector $\vec{B}$.

The task is to find an integral solution to the set of inequalities inequalities (or to determine if it exists)

$$\sum_j a_{ij} x_j \leq B_i$$

Fact

The decision version of ILP is **NP***-complete.*

By adding inequalities $0 \leq x_j \leq 1$ we may restrict the set of solutions to 0s and 1s.

Cutting planes proof system

A proof line is an inequality

$$\sum_k c_k x_k \geq C,$$

where c_k and C are integers.

The *axioms* are $\sum_j a_{ij} x_j \leq B_i$

The *rules* are

1. **addition:** from $\sum_k c_k x_k \geq C$ and $\sum_k d_k x_k \geq D$ derive $\sum_k (c_k + d_k) x_k \geq C + D$;
2. **multiplication:** from $\sum_k c_k x_k \geq C$ derive $\sum_k d c_k x_k \geq dC$, where d is an arbitrary positive integer;
3. **division:** from $\sum_k c_k x_k \geq C$ derive $\sum_k \frac{c_k}{d} x_k \geq \left\lceil \frac{C}{d} \right\rceil$, provided that $d > 0$ is an integer which divides each c_k.

To prove the unsatisfiability of the inequalities we need to derive

$$0 \geq 1$$

Lovász Schrijver system

We want to prove unsatisfiability of linear inequalities in integers by deriving the contradiction $-1 \geq 0$

Proof lines are linear and quadratic inequalities.

Axioms

1. the given inequalities and
2. $x_i^2 - x_i = 0$, for all variables x_i.

Rules

1. positive linear combinations;
2. from linear inequality $\sum_k c_k x_k + C \geq 0$ derive $x_i(\sum_k c_k x_k + C) \geq 0$;
3. from linear inequality $\sum_k c_k x_k + C \geq 0.$ derive $(1 - x_i)(\sum_k c_k x_k + C) \geq 0$

Note that one has to get rid of quadratic terms before applying rules (2) and (3).

Hybrid systems

Bounded depth Frege system with the parity gate.

Although exponential lower bounds for bounded depth circuits with parity gates are known since 1986, for this proof system we do not have lower bounds. Only the first step has been done: lower bounds on the Polynomial Calculus.

Lecture 2: A lower bound on the degree for Polynomial Calculus proof of the Pigeon-Hole Principle.[1][2]

Lecture 3: A lower bound on the size of Cutting-Plane proofs of the Clique-Coloring tautology.[3]

---

[1] A.A. Razborov: Lower bounds for the polynomial calculus.

[2] R. Impagliazzo, P. Pudlak, J. Sgall: Lower bounds for the polynomial calculus and the Groebner basis algorithm.

[3] P. Pudlak: Lower bounds for resolution and cutting planes proofs and monotone computations.