Course ”Propositional Proof Complexity“, JASS’09

Polynomial Calculus

Maximilian Schlund

May 2, 2009

Contents

1 Motivation

What is ”Polynomial Calculus” good for?

- a proof system for refuting systems of polynomial equations
- ”strong” proof system (e.g. compared to resolution)
- quite efficient algorithms for automatic proof search exist (Groebner Bases - Buchberger’s Algorithm)

We will consider two types of algebraic proof systems:

- Nullstellensatz proof system (NS)
- Polynomial calculus (PC) - stronger than NS

Both systems try to prove that a system of polynomial equations $g(x) = 0$ has no solution.

2 Preliminaries

2.1 Polynomials and Propositional Logic

There is a direct connection to Propositional Logic: We can easily translate a propositional formula into a system of equations $g(x) = 0$ that is satisfiable if and only if the formula is satisfiable. One possibility to do this is to use the following (recursive) translation Φ:

X	$\Phi(X)$
$\top$	$0 = 0$
$\bot$	$1 = 0$
x_i	$(1 - x_i) = 0$
$\neg A$	$1 - \Phi(A) = 0$
$A \vee B$	$\Phi(A) \cdot \Phi(B) = 0$

For each variable x_i we add the equation "$x_i^2 - x_i = 0$" (expresses $x_i \in \{0, 1\}$) (note that normally we ommit the "$= 0$", and use the words "polynomial" and "equation" interchangeably) As an Example we look at a simple translation of a formula:

$$x \vee y \rightarrow z \rightsquigarrow [1 - (1 - x)(1 - y)]z \rightsquigarrow xz + yz - xyz$$

Note that the "$\wedge$"-operation could be expressed by $\neg$ and $\vee$ but it is more effective to translate the operands seperately to two eqations and add them to the set of all equations.

2.2 Nullstellensatz

A very important theorem from algebraic geometry that is the foundation of algebraic proof systems is the following

Theorem 1 (Hilbert's (weak) Nullstellensatz)**.** *Let F be an algebraically closed field and $f_1, \ldots, f_n$ be a system of polynomials over F. This system of polynomials is unsatisfiable if and only if 1 is in the ideal generated by the $f_1, \ldots, f_n$.*

$$\nexists x \in F^m.\ \forall 1 \leq i \leq n.\ f_i(x) = 0 \Leftrightarrow \exists g_1, \ldots, g_n : \sum_{i=1}^{n} g_i f_i = 1$$

The proof can be found in any textbook about about algebraic geometry or commutative algebra.

Nullstellensatz proof system A proof in the NS proof system of the unsatisfiability of $p_1, \ldots, p_n$ is a system $q_1, \ldots, q_n$ such that

$$\sum_{i=1}^{n} p_i q_i = 1$$

A measure for the size of a NS proof is $\max_i(\deg(q_i))$.

Note that the complexity of algebraic proof systems depends heavily on the representation of the polynomials involved. For example it can make a huge difference if the polynomials are presented in a dense (as a list of all coefficients) or in a sparse representation (as a list of only non-zero coefficients). This fact also makes it difficult to compare the power and the efficiency of algebraic proof systems to other proof systems like Resolution or Frege systems.

2.3 Polynomial calculus

Polynomial calculus We start with a system of polynomials and try to prove the constant polynomial 1 (i.e. the unsatisfiable equation $1 = 0$) using the following inference rules:

$$\frac{P \quad Q}{aP + bQ} \quad (with\ a, b \in F)$$

$$\frac{P}{xP} \quad (with\ x \in \{x_1, \ldots, x_n\})$$

Axioms

$$x_i^2 - x_i \quad (for\ all\ Variables\ x_i)$$

These axioms force the variables to take only boolean values. By moving all calculations to the quotient ring $K[x_1, \ldots, x_n]/I$, where I is the ideal generated by the axiom polynomials we can get rid of stating and using the axioms explicitly.

The size of a PC proof is measured as the maximum degree over all polynomials appearing in the proof.

We write $p_1, \ldots, p_n \vdash_d q$ if q has a PC proof from the p_i with size at most d

A proof $p_1, \ldots, p_n \vdash_d q$ in PC can be expressed as a list of polynomials $r_1, \ldots, r_k, q$ where each r_i is either an axiom (i.e. $x^2 - x$), an assumption (one of the p_j) or it is derived from some previous (i.e. some r_j with $j < i$) polynomials in the proof.

3 Properties of PC and Relation to other Proof systems

3.1 Simple Properties

Because of the axioms $x_i^2 - x_i$ (more explicit: $x_i^2 = x_i$) or more formally by looking at the quotient ring $K[x_1, \ldots, x_n]/I$ (with I the ideal generated by the $x_i^2 - x_i$), we can restrict ourselves to to multilinear polynomials (i.e. each variable has an exponent of at most 1) appearing in the proof. For example

$$x^2y^2z \rightsquigarrow xy^2z \rightsquigarrow xyz$$

$$\frac{x^2y^2z \qquad \dfrac{x^2 - x}{x^2y^2z - xy^2z}}{xy^2z}$$

It is obvious that the space of all multi-linear polynomials of degree at most d over F is a vector space.

Let $m(p)$ denote the mapping that maps every polynomial to the corresponding multilinear polynomial (i.e. replaces every x^n with x). So $m(p)$ is just the canonical (surjective) quotient map from $K[x_1, \ldots, x_n]$ to $K[x_1, \ldots, x_n]/I$.

Definition 2. Let $V_d(p_1, \ldots, p_n)$ denote the smallest subspace V of this space that

1) includes all p_i and

2) if $p \in V$ and $deg(p) \leq d-1$ then $m(xp) \in V$

We now arrive at a Vector-space characterization of formulas that are provable via bounded degree PC proofs.

Theorem 3. *Let $p_1, \ldots, p_n, q$ be multi-linear polynomials of degree at most d then:*

$$p_1, \ldots, p_n \vdash_d q \Leftrightarrow q \in V_d(p_1, \ldots, p_n)$$

Proof. Define $V := \{q \mid q\ multi-linear,\ p_1, \ldots, p_n \vdash_d q\}$. We have to show that $V_d(p_1, \ldots, p_n) = V$

" $\Leftarrow$ " : prove $V_d(p_1, \ldots, p_n) \subseteq V$ by showing that V has all the properties of $V_d(p_1, \ldots, p_n)$.

" $\Rightarrow$ " : Assume there is a $q \in V - V_d(p_1, \ldots, p_n)$. Then q has a degree d proof in PC $r_1, \ldots, r_m$. Let r_i be the first line with $m(r_i) \notin V_d(p_1, \ldots, p_n)$. Distinguish cases for r_i and derive contradiction.

Cases:

- r_i cannot be one of the p_i and neither an axiom $x^2 - x$ as $m(x^2 - x) = 0$.
- r_i cannot be of the form $aP + bQ$ of previous lines, because $V_d(p1, \ldots, p_n)$ is a vector space.
- r_i cannot be xP for a previous line P, since $deg(P) \leq d-1$ and then $m(xP) \in V_d(p1, \ldots, p_n)$

All cases yield a contradiction! Therefore $q \in V_d(p1, \ldots, p_n)$ and so $V_d(p1, \ldots, p_n) = V$. □

This result also yields an algorithm for determining if q is provable from $p_1, \ldots, p_n$ by a degree d PC proof: Compute a basis for $V_d(p_1, \ldots, p_n)$ and then check if q lies in the vector space. A simple algorithms achieving this is presented in [CEI96] having a runtime of $\mathcal{O}(n^{3d})$.

Now some simple technical results that are helpful when working with PC proofs.

Lemma 4. *Let x be a variable and $p, p_1, \ldots, p_k, q, q'$ be multilinear polynomials of degree at most d*

1. *If* $p_1, \ldots, p_k, x \vdash_d 1$ *then* $p_1, \ldots, p_k \vdash_{d+1} 1 - x$
2. *If* $p_1, \ldots, p_k, 1 - x \vdash_d 1$ *then* $p_1, \ldots, p_k \vdash_{d+1} x$
3. $p, x \vdash_d p|_{x=0}$
4. $p, 1 - x \vdash_d p|_{x=1}$
5. *If* $p_1, \ldots, p_k \vdash_d q$ *and* $p_1, \ldots, p_k, q \vdash_d q'$ *then* $p_1, \ldots, p_k \vdash_d q'$
6. *If* $p_1|_{x=0}, \ldots, p_k|_{x=0} \vdash_d 1$ *and* $p_1|_{x=1}, \ldots, p_k|_{x=1} \vdash_{d+1} 1$ *then* $p_1, \ldots, p_k \vdash_{d+1} 1$
7. *If* $p_1|_{x=1}, \ldots, p_k|_{x=1} \vdash_d 1$ *and* $p_1|_{x=0}, \ldots, p_k|_{x=0} \vdash_{d+1} 1$ *then* $p_1, \ldots, p_k \vdash_{d+1} 1$

Part 1 If $p_1, \ldots, p_k, x \vdash_d 1$ then $p_1, \ldots, p_k \vdash_{d+1} 1 - x$

Proof. Let

$$p_1, \ldots, p_k, x, r_1, \ldots, r_k, 1$$

be a PC refutation of $p_1, \ldots, p_k, x$ with degree d.

Then

$$p_1, \ldots, p_k, p_1(1-x), \ldots, p_k(1-x), x(1-x), r_1(1-x), \ldots, r_k(1-x), (1-x)$$

is a degree $d + 1$ PC proof of $1 - x$.

Explanation: $p_i(1 - x)$ can be derived from p_i, $x(1 - x)$ is an axiom, so it can be trivially derived and $r_i(1 - x)$ can be proved like r_i in the original refutation:

$$\frac{q_j \quad q_l}{aq_j + bq_l = r_i} \quad \rightsquigarrow \frac{(1-x)q_j \quad (1-x)q_l}{(1-x)(aq_j + bq_l) = (1-x)r_i}$$

What if e.g. q_l is x? We do not have x as an assumption anymore... $\rightsquigarrow$ but it turns into an axiom!

$$\frac{q_j \quad x}{aq_j + bx = r_i} \quad \rightsquigarrow \frac{(1-x)q_j \quad (1-x)x}{(1-x)(aq_j + bx) = (1-x)r_i}$$

□

Part 2 If $p_1, \ldots, p_k, 1 - x \vdash_d 1$ then $p_1, \ldots, p_k \vdash_{d+1} x$

Proof. Essentially same proof as 1. □

Part 3 $p, x \vdash_d p|_{x=0}$

Proof. Multiply x by appropriate variables and then subtract from p to cancel out all terms in p that contain x. □

Part 4 $p, (1-x) \vdash_d p|_{x=1}$

Proof. Essentially same proof as 3. □

Part 5 If $p_1, \ldots, p_k \vdash_d q$ and $p_1, \ldots, p_k, q \vdash_d q'$ then $p_1, \ldots, p_k \vdash_d q'$

Proof. Concatenate the proofs. □

Part 6 If $p_1|_{x=0}, \ldots, p_k|_{x=0} \vdash_d 1$ and $p_1|_{x=1}, \ldots, p_k|_{x=1} \vdash_{d+1} 1$ then $p_1, \ldots, p_k \vdash_{d+1} 1$

Proof. With Part 3 we get

$$p_1, \ldots, p_k, x \vdash_d p_1|_{x=0}, \ldots, p_k|_{x=0} \vdash_d 1$$

And by Part 1 we get:

$$p_1, \ldots, p_k \vdash_{d+1} 1 - x$$

Since $p_1|_{x=1}, \ldots, p_k|_{x=1} \vdash_{d+1} 1$ we obtain $p_1, \ldots, p_k, 1-x \vdash_{d+1} 1$ and by Part 5 we end up with $p_1, \ldots, p_k, \vdash_{d+1} 1$ by concatenating the proofs. □

Part 7 If $p_1|_{x=1}, \ldots, p_k|_{x=1} \vdash_d 1$ and $p_1|_{x=0}, \ldots, p_k|_{x=0} \vdash_{d+1} 1$ then $p_1, \ldots, p_k \vdash_{d+1} 1$

Proof. Essentially same proof as 6. □

3.2 Relation to other proof systems

We now want to compare PC with other proof systems for propositional logic. At first we state that PC can (quasi-polynomial) simulate tree-like Resolution proofs:

Theorem 5. *If the set of Clauses $C_1, \ldots, C_n$ of size at most k has a tree-like resolution proof with S lines, then the corresponding polynomials have a PC refutation of degree $k + \log_2 S$.*

Proof. Induction on S. Let $p_1, \ldots, p_n$ be the direct translations of the C_i into polynomials. The maximum degree of the p_i is k. The last line of the resolution refutation is of course $\emptyset$.

Base case: If $\emptyset = C_i$ for a i, then the corresponding translation is $p_i = 1$. This is a trivial degree 0 PC refutation with 1 lines.

Ind.-step: x was resolved with $\neg x$ for some varible x. Then x has a (tree-like) resolution derivation of S_1 lines and $\neg x$ has a derivation of S_2 lines, s.t. $S_1 + S_2 = S - 1$.

Setting $x = 0$ in the proof with S_1 lines gives a resolution refutation from the $C_i[0/x]$ so by induction we have $p_1|_{x=0}, \ldots, p_k|_{x=0} \vdash_{m+\log_2 S_1} 1$. (Note that the translation of $C_i[0/x]$ is $p_i|_{x=0}$). Similarly by setting $x = 1$ in the proof with S_2 lines we get a refutation from the $C_i[1/x]$ so by induction we have $p_1|_{x=1}, \ldots, p_k|_{x=1} \vdash_{m+\log_2 S_2} 1$. If $S_1 \leq S/2$ then applying Part 6 of the previous Lemma with $d = m + \log_2 S - 1 \geq m + \log_2 S_1$ we get $p_1, \ldots, p_k \vdash_{m+log_2 S} 1$. This works symmetrically for $S_2 \leq S/2$ and applying Part 7 instead.

□

4 Lower bounds

4.1 Seperation of NS and PC

We will now prove a lower bound on NS refutations using a modified version of the PHP called "House sitting principle" (HSP). Note that an upper bound on NS refutations is n if we have n variables and the equations "$x_i^2 - x_i = 0$" are in the refutation set. Then we can assume the g_i to be multi-linear in $\sum_i f_i g_i = 1$

- n+1 pigeons, n houses ordered by attractivity
- Pigeon i owns house i for $1 \leq i \leq n$
- Pigeon 0 owns no home.
- All pigeons must stay at their own or at a house nicer than their own
- At most 1 pigeon per house allowed

We will show that the HSP has a degree 2 PC refutation but requires a proof of degree n in NS.

The easy part first - the PC refuation. Informal proof of the HSP first: Using induction "backwards".

Base Pigeon n has the nicest house and must live somewhere, so it is at home.

Step Assume that pigeons $[i + 1..n]$ are all at home.

- Because all the houses $[i + 1..n]$ are occupied, pigeon i has to take its own house to live.
- We conclude that pigeon 0 is at home, but it is homeless! $\leadsto$ Contradiction!

We will mimic this informal proof formally.

Therefore, first translate the HSP into a system of equations.

- $\forall i \in [0..n], j \in [1..n]$, we introduce variables $x_{(i,j)}$ - meaning pigeon i is in house j
- $\forall i \in [0..n],\ j \in [1..n]\ Q'_{(i,j)} := x^2_{(i,j)} - x_{(i,j)} = 0$ - forces the variables to take 0/1-values.

- $\forall i \in [0..n] : Q_i := (\sum_{j\in[i..n]} x_{(i,j)}) - 1 = 0$ - pigeon i is in one hole that is at least as nice as its own.
- $Q := x_{(0,0)} = 0$ - Pigeon 0 is homeless.
- $\forall i \in [0..n],\ j \in [i+1..n]\ Q_{(i,j)} := x_{(i,j)}x_{(j,j)} = 0$ - pigeon i cannot go to house j if pigeon j is at home.
- $\forall i \in [0..n],\ j,k \in [1..n]\ Q_{(i,j,k)} := x_{(i,j)}x_{(i,k)} = 0$ - a pigeon cannot be in more than one house.

First we start with the assumption $Q_{(n,n)} = x_{(n,n)} - 1$ (i.e. pigeon n is at home). From this (and the other assumptions) we derive $x_{(n-1,n)}$ and $x_{(n-1,n-1)} - 1$ (i.e. pigeon $n-1$ is not in house n and is at home) and so on...

So we construct the proof inductively ("backward" Induction on i):

- For $i = n$ we get $Q_{(n,n)} = x_{(n,n)} - 1$ directly from the assumptions
- Assume we have derived the equations $x_{(i+1,i+1)} - 1, \dots, x_{(n,n)} - 1$
- $\forall j \in [i+1..n]$ derive $x_{(i,j)} = -x_{(i,j)} \cdot (x_{(j,j)} - 1) + Q_{(i,j)}$
- from this derive $x_{(i,i)} = Q_i - \sum_{j\in[i+1..n]} x_{(i,j)}$
- Finally we derive $x_{(0,0)}$ and $Q - x_{(0,0)} = 1$ gives us the derivation of 1 and therefore completes the refuation.

Now a sketch of the proof for the claim that every NS proof (over $\mathbb{Z}_2$) of the HSP requires degree n. Assume we have a NS proof of degree $n-1$. We show that this implies the non-existence of a structrue called a n-design, but these structures exist so we get a contradiction. Suppose we have Polynomials P of degree at most $n-1$ so that:

$$\sum_{i\in[0..n]} P_iQ_i + \sum_{i\in[0..n],j,k\in[1..n]} P_{(i,j,k)}Q_{(i,j,k)} +$$
$$\sum_{i\in[0..n],j\in[i+1..n]} P_{(i,j)}Q_{(i,j)} + PQ + \sum_{i\in[0..n],j\in[1..n]} P'_{(i,j)}Q'_{(i,j)} = 1$$
$$\Leftrightarrow \sum_{i\in[0..n]} P_iQ_i \equiv 1\ (mod Q_{(i,j,k)}, Q_{(i,j)}, Q, Q'_{(i,j)})$$

We simplified the equation by moving to the quotient ring given by the above modulus.

By multiplying out the identity $\sum_{i\in[0..n]} P_iQ_i \equiv 1$ and equating coefficients on boths sides we obtain a system of linear equations for the coefficients of the P_i. One can then prove that this equations have a solution iff a structure called $n-design$ does not exist. But such a structure can be constructed (see for example [Bus98] for a general construction) and therefore we get a contradiction.

There are also results for linear lower bounds on PC proofs, like:

Theorem 6. *There is a graph G with constant degree s.t. a Tseitin tautology for G with all charges 1 requires degree $\Omega(n)$ to prove in PC.*

The proof in [BGIP99] is quite well explained and readable.

References

[Bea] P. Beame. Proof complexity. Lecture notes about Proof Complexity, URL: www.cs.toronto.edu/~toni/Courses/Proofcomplexity/Papers/paul-lectures.ps.

[BGIP99] Sam Buss, Dima Grigoriev, Russell Impagliazzo, and Toniann Pitassi. Linear gaps between degrees for the polynomial calculus modulo distinct primes. In *STOC '99: Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 547–556, 1999.

[BIK+94] P. Beame, R. Impagliazzo, J. Krajicek, T. Pitassi, and P. Pudlak. Lower bounds on hilbert's nullstellensatz and propositional proofs. In *SFCS '94: Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 794–806, 1994.

[Bus98] Samuel R. Buss. Lower bounds on nullstellensatz proofs via designs. In *in Proof Complexity and Feasible Arithmetics, P. Beame and S. Buss, eds., American Mathematical Society*, pages 59–71. American Math. Soc, 1998.

[CEI96] Matthew Clegg, Jeffery Edmonds, and Russell Impagliazzo. Using the groebner basis algorithm to find proofs of unsatisfiability. In *STOC '96: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 174–183, 1996.

[Gri98] D. Grigoriev. Tseitin's tautologies and lower bounds for nullstellensatz proofs. In *FOCS '98: Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, page 648, 1998.

[Raz98] Alexander A. Razborov. Lower bounds for the polynomial calculus. *Comput. Complex.*, 7(4):291–324, 1998.