Exercises on Algebraic Proof Complexity
CSCI 6114 Fall 2021

Joshua A. Grochow

October 14, 2021

Recall that a Nullstellensatz proof that a system of polynomial equations $f_1(\vec{x}) = \cdots = f_m(\vec{x}) = 0$ is unsatisfiable is a list of polynomials g_i such that

$$\sum_{i=1}^{m} f_i(\vec{x}) g_i(\vec{x}) = 1.$$

The typical complexity measure used for Nullstellensatz proofs is $\max_i \deg(f_i g_i)$.

1. Consider the n-th "induction principle":

   $$(x_1) \wedge (\neg x_1 \vee x_2) \wedge (\neg x_2 \vee x_3) \wedge \cdots \wedge (\neg x_{n-1} \vee x_n) \wedge (\neg x_n)$$

   (a) Show that it is unsatisfiable.

   (b) What is the size of a resolution refutation of the n-th induction principle?

   (c) What is the smallest-degree Polynomial Calculus refutation you can find?

   (d) What is the smallest-degree Nullstellensatz refutation you can find?

2. Show that Polynomial Calculus p-simulates Nullstellensatz.

3. Show that a Polynomial Calculus proof in which at most m monomials appear in the entire proof can be verified in time polynomial in m and n (the number of variables).

4. Show that a Polynomial Calculus proof in which the maximum degree appearing is d can be verified in $n^{O(d)}$ time.

5. Show that, if there exists a PC proof of maximum degree d, then it can be *found* in $n^{O(d)}$ time.

6. Show that a Nullstellensatz proof of degree d can be verified in $n^{O(d)}$ time.

7. Show that a Nullstellensatz proof of degree d can be found in $n^{O(d)}$ time.

8. The DPLL (Davis–Putnam–Logemann–Loveland) family of algorithms for SAT works as follows. Given a Boolean formula φ, it somehow chooses a variable x_i, and sequentially tries setting $x_i = 1$ and $x_i = 0$. It then simplifies the formula before iterating. When all variables have been assigned but φ is not satisfied (or if it can tell φ is not satisfied by the current partial assignment), it backtracks; if it ever finds a satisfying assignment it stops. If it gets to the end of its (potentially exponentially long) search without finding a satisfying assignment, it returns UNSATISFIABLE. DPLL is really a family of algorithms, depending on how the next variable is chosen, and whether it attempts setting the variable to 1 or 0 first, and what kinds of simplifications it does to the formula. Show that if φ is unsatisfiable, then for any DPLL algorithm, from its computation history on input φ one can extract a resolution refutation of φ.

   Conclude that resolution lower bounds imply lower bounds on the runtime of all algorithms in the DPLL family.

Resources

- Pitassi–Tzameret survey on algebraic proof complexity
- Fleming–Kothari–Pitassi monograph on semi-algebraic proof systems (preprint available on ECCC)
- Atserias & Maneva ITCS '12: show equivalence between WL and Sherali–Adams for Graph (non)Isomorphism.