CS 2429 – Fall 2002
Location: WB 119
Time: 10-12

Instructor: Toniann Pitassi
Office: 2305A Sandford Fleming Building
email: toni@cs.toronto.edu
Office hours: by appointment

Course Web Site: http://www.cs.toronto.edu/ toni/Courses/Proofcomplexity/CS2429.html
Refer to this site periodically for important announcements and other information. All handouts will be available on the site in postscript form.

Course Materials: There is no book for this course. Each lecture will have supplemental reading material such as a paper or lecture notes, available on the website.

Course Description

This is a topics course in propositional proof complexity. The only prerequisite for this course is the equivalent of CS364 (undergraduate complexity theory).

Proof complexity, the study of the lengths of proofs in propositional logic, is an area of study that is fundamentally connected both to major open questions of computational complexity theory and to practical properties of automated theorem provers and reasoning systems. In the last decade, there have been a number of significant advances in proof complexity. Moreover, new connections between proof complexity and circuit complexity have been uncovered, and the interplay between these two areas has become quite rich. In addition, concepts and techniques in proof complexity have borrowed from and contributed to several other areas of computer science, including: crytography, automated reasoning and AI, complexity of search classes within NP, the analysis of heuristics and algorithms for NP-hard problems, and bounded arithmetic. In this course, we will cover the fundamentals of propositional proof complexity, and some of the connections with these other areas.

Here is an outline of topics that I hope to cover in this course. Please let me know if there is a specific topic (listed or otherwise) that you are particularly interested in, as we are likely to not get through all of the material.

(1.) Introduction to proof complexity, propositional proof complexity, basic concepts and definitions, motivation, connections to complexity theory and logic.

(2.) Define standard proof systems: Resolution, Frege systems, bounded-depth Frege systems, Extended Frege, Substitution Frege, Cutting Planes, algebraic proof systems. Complexity measures: size, width, number of lines, tree versus dag form. Polynomial-simulations and relative complexity of these proof systems. Lower bounds for a propositional proof system P show that algorithms based on P cannot solve SAT in polynomial time.

(3.) Resolution. Hard examples for resolution, including the pigeonhole principle and random formulas. Lower bounds and the width method. Lower bounds for PHP and random formulas, and implications for satisfiability algorithms. Lower bounds for the P versus NP statement and implications for complexity theory.

(4.) The interpolation method. Craig's interpolation, feasible interpolation, and connection to complexity theory. Prove that Resolution and Cutting Planes have feasible interpolation, and how this implies superpolynomial lower bounds. Limitations of the method–negative results for Frege systems under cryptographic assumptions. The tradeoff between proof search and proof length: automatizability.

(5.) Frege systems. The sequent calculus. Equivalence between tree and dag form. Hard examples for Frege including consistency statements, Avigad's formulas, the P versus NP statement, statements from linear algebra. Subsystems of Frege including: bounded arithmetic, LA, LAP. Connections with bounded arithmetic. The best known lower bounds for unrestricted Frege.

(6.) Bounded depth Frege systems. The restriction method and lower bounds for bounded-depth Frege systems. Limitations of the method.

(7.) Algebraic proof systems. Algebraic proofs, nullstellensatz proofs, polynomial calculus proofs. Simulations and connections to standard (boolean) proof systems. Lower bounds.

(8.) Extended Frege. Equivalence of Extended Frege (EF), substitution Frege, Hajos Calculus. Hard examples. Best known lower bounds.

(9.) Beyond EF: Is there a strongest (super) proof system?

(10.) Beyond SAT: Proof systems for other coNP complete problems. Proof systems for QSAT, SSAT, etc, and connections to problems in probabilistic reasoning, bayes nets and probabilistic planning.

(11.) Connections to NP search classes.

Grading and Assignments

Grading will be based on 4 assignments which will be handed out during the semester. You will have at least one week to turn in each assignment. Extra challenging questions will be marked with a (*). The work you submit must be your own. You may discuss problems with each other; however you should prepare written solutions alone. Class attendence is manditory and you are encouraged to ask many questions in class.