Term Project Information CS 764 — Spring 2014

Purpose of the project

A major intent of the term project is that you learn about the way research is done. You will find references, you will read, correlate and criticize previous work, and you will have some ideas of your own.

Although you may start from a single paper, you should follow up references and discuss the focal work in a somewhat wider context. You should use whatever sources of information you can think of. Web resources can help a lot:

- Google Scholar (http://scholar.google.com) provides the usual Google capabilities but limits searches to scholarly articles, which greatly reduces spurious hits.
- The DBLP database (http://www.dagstuhl.de/db) provides a searchable index of most publications in computer science. You can track authors, journals, etc.
- CiteSeer (http://citeseer.psu.edu) gathers citation data. Useful for questions like "what work has been done since this paper?"

Also, talk to people who might know something.

You may find that you end up with a project different than the one you had in mind at the start. If you found something interesting, great!

Project outlines

Hand in an outline of your project by Wednesday, July 9. Describe in three or four *double-spaced* pages (about 700–1000 words) what you plan to do for your project. Be as specific as you can; give the topics you will cover and explain how you will treat them. Include a preliminary bibliography.

I will check your outlines for suitability of the project, for style, and for grammar and spelling. I will return unsatisfactory outlines for revision. I will not accept your final paper unless you previously submitted a satisfactory outline.

Class presentations

We will have project presentations during the last week or two of the term—likely two presentations per class period. The exact schedule will be set once we know the actual slate of projects.

Ten percent of your project mark will be based on the class presentation.

Prepare your presentations carefully. You will find it impossible to present everything; you will need to pick and choose a subset that you can explain to your fellow students. Give yourself a "practice" talk (to friends, to your bathroom mirror, or whatever) so that you can estimate your actual time required.

Final papers

Submit your papers by noon on Monday, Aug. 11. A typical paper will be approximately fifteen to twenty-five *double-spaced* pages in length. I will look for content and presentation much more than length.

When reading your papers, I will hope to find some contribution from you. Originality can take many forms—choose from my list or do your own thing.

- Survey a collection of related work. Provide an overall view—not just a listing of the results.
- Present results in a different way than originally done. Usually, the first presentation of a result is not the simplest possible. Also, important ideas may develop over time rather than appearing in any one place.
- Fix errors or inconsistencies in published work. Or fill in deliberate gaps, which often occur in conference publications with page limitations.
- Compare and contrast two or more different approaches. Can you prove a result by a different method than the original proof?
- Suggest new approaches or possible avenues of investigation. If you discover that your ideas don't work, report on that.
- Do something new. Feel free to speculate on what might be possible, but make sure to differentiate between conjecture and proof.

Note that quoting or paraphrasing a previous work does none of the above. Quoting and paraphrasing do have their appropriate uses, but they don't count toward originality.

Summary of dates

Outlines due: Wednesday, July 9, in class. Presentations: July 23–30 (details tba). Papers due: Monday, Aug. 11, at noon.

Some Possible Starting Points

Note: None of these papers is the last word on its subject; some are not even the final version of the paper. Make sure you look for other work!

Turing machines

What is the precise relationship between time complexity and space complexity? What has been discovered since these 30-year-old papers?

- J. Hopcroft, W. Paul, L. Valiant, "On time versus space," J. Assoc. Comput. Mach. 24 (1977) 332–337.
- W.J. Paul, N. Pippenger, E. Szemerédi, W.T. Trotter, "On determinism versus nondeterminism and related problems," *IEEE Conference on Foundations of Computer Science*, 24th annual, 1983, pp. 429–438.

How hard is SAT, anyway?

- L. Fortnow, "Time-space tradeoffs for satisfiability," J. Computer System Sciences 60 (2000).
- R. Impagliazzo, R. Paturi, "On the Complexity of k-SAT," J. Computer and System Sciences, 62 (2001) 367–375.
- R. Williams, "Improving Exhaustive Search Implies Superpolynomial Lower Bounds," ACM Symposium on Theory of Computing, 42nd annual, 2010, pp. 231–240.

Proof complexity

How does the size of the proof of a property relate to the complexity of computing it?

E. Ben Sasson, A. Wigderson, "Short Proofs are Narrow—Resolution Made Simple," J. Assoc. Comput. Mach. 48 (2001) 149–169.

For an overview of the topic, see Chapter 15 of Arora/Barak.

Descriptive Complexity

Relating the complexity of describing a problem to the difficulty of computing solutions.

- N. Immerman, "Languages that capture complexity classes," SIAM J. Computing 16 (1987) 760-778.
- N. Immerman, J. Buss, D. Mix Barrington, "Number of variables is equivalent to space," J. Symbolic Logic 66, 2001.

Kolmogorov complexity

Some systems (or strings) exhibit subtle, "deep" structure. Can we measure the subtlety of a system?

L. Antunes, L. Fortnow, D. van Melkebeek, "Computational Depth," Proc. Sixteenth Ann. IEEE Conf. Computational Complexity, IEEE Computer Society, 2001, pp. 266–273.

"Application" areas

Survey how the study of computational complexity impacts or illuminates an area of application. Some possible areas:

- Robotics. (E.g.: many problems are complete for *PSPACE*.)
- Software engineering. (E.g.: how does computational complexity limit the use of "formal methods"—or not?)
- •

You should consult with the instructor before deciding on a project of this type, to determine the suitability of your intended work for credit in CS764.

Serendipity

Browse through recent proceedings of a complexity conference, and find something interesting! Possible conferences (held annually) include

"CCC": IEEE Conference on Computational Complexity

"SToC": ACM Symposium on Theory of Computing

"FoCS": IEEE Conference on Foundations of Computer Science

As with the previous group, consult with the instructor to determine the suitability of your proposal.