Term Projects

This year each student in 6.852 is required to carry out a term project related to distributed algorithms. This handout describes what we would like you to do.

Students should work in project teams of two or three. More than three is too many since that makes it hard for everyone to get a chance to contribute. Working alone isn’t so much fun, but let us know if you really want to do that.

You are required to write a report, due on the final regularly-scheduled day of classes. Your report should be no more than 15 pages long (in 11 pt font). We will have presentations based on the reports on the final day of classes (which is a Tuesday) and probably one overflow session on the following Thursday, same time.

1 Kinds of projects

There are at least three kinds of projects that are acceptable. (If you have another idea, ask us.) Keep in mind that this is a theory course, so anything you do should be mostly about theory.

Reading project: In this type of project, you choose a topic of current interest in the distributed algorithms research community, learn about it, and write a report to explain it to others.

That is, read and understand several key papers on your topic and write an expository report explaining the important theoretical ideas. You should write for an audience at the level of the students in our class. If you do this well, your explanations will be easier for the class to understand than the original papers, which are generally written for specialists on the research topic.

Theoretical research project: In this type of project, you find a theoretical research problem you are interested in, and devise your own new algorithms or prove your own new lower bound results (or both). Write a theoretical research report about it. The style should be like that of the conference papers you have seen in this course, but you should try to make the results understandable to students like those in our class.

We suggest that you consider a currently-active research topic, since it’s harder to get new results about older, well-studied problems.

To work on a new problem, you will need to do background reading on the topic. That will provide you with a convenient fall-back: in case your attempts at new results don’t work out as you planned, you can simply turn the project into a reading project.

Experimental research project: In this type of project, you select known distributed algorithms from the literature, and perhaps devise some variations. Think of clear questions to ask about the algorithms that might be answered by means of experiments. For example, you might ask how the algorithms behave under a variety of assumptions about parameter values, about inputs, etc.

Simulate the algorithms to answer your questions. (We don’t recommend trying to run your algorithms in real distributed settings, but ask us if you really want to do that.) Write an experimental research report to describe the questions, experiments, and results. The report should contain clear questions that your experiments are designed to answer, clear descriptions of your experiments and their results, and clear conclusions.
2 Sources

Typical sources for current research on distributed computing theory are various conferences devoted to this area, such as the Principles of Distributed Computing conference (PODC), the International Symposium on Distributed Computing (DISC), the Symposium on Parallelism in Algorithms and Architectures (SPAA), the International Conference on Principles of Distributed Systems (OPODIS), etc.

Other conferences, such as ICDCS, have special “tracks” for distributed algorithms papers. More general theoretical conferences, such as FOCS, STOC, and SODA, publish distributed algorithms papers. If you are interested in basic concurrency theory, CONCUR publishes papers on this topic.

You can also find ideas in journal papers. The journal Distributed Computing is entirely devoted to distributed computing theory papers. Many broader journals such as JACM, Information and Computation, etc., also publish papers on distributed computing theory. The Morgan Claypool Synthesis series of monographs on distributed computing theory can also provide ideas. However, note that journal papers and monographs are generally more polished and complete than conference papers, so they may leave less for you to work on.

3 Suggestions for topics

This is a partial list of interesting topics that have been studied in the past few years. It is not intended to be complete in any sense, just ideas that we thought of quickly.


- **Asynchronous shared memory algorithms:** Randomized consensus algorithms and lower bounds. Transactional memory.

- **Asynchronous network algorithms:** Cooperative task-oriented computing. Quorum-based data-management algorithms. Failure detectors.

- **Timing-based algorithms:** Using Paxos to implement reliable data storage. Gradient clock synchronization.

- **Flexible, robust, adaptable distributed algorithms.** Computing functions in dynamic graph networks. Self-stabilizing algorithms. Computing in the presence of ongoing changes. Reconfigurable atomic memory algorithms.


- **Biologically-inspired distributed algorithms:** Insect colony algorithms for problems such as foraging, task allocation, or cooperative construction. Algorithms for patterning among cells in developing embryos. Distributed algorithms for learning and memory in brains.

4 Requirements

**Project proposal:** A one-page project proposal is due on Thursday, October 31. This will give us a chance to consider your project idea and make suggestions.

State clearly what you are planning to do, and which type of project (of the three listed above) you are carrying out. Include relevant references.
**Final report:** Your final report is due at the last regular class, on Tuesday, Dec. 10. It should be at most 15 pages long (in 11 point font). It should include relevant references.

We suggest that, before handing your paper in, you ask other students in the class (students not in your project group) to read your paper and comment about what is not clear. Revise based on their suggestions.

**Presentations:** Students will present summaries of their projects at the last regularly scheduled class, on Tuesday, Dec. 10. If this is not enough time, we will finish up at the same time on Thursday, Dec. 12. Food will be provided.

Attendance at the extra class meeting is optional, and if you cannot attend it will not affect your grade. The main purpose of the presentations is to give you a chance to tell everyone what you worked on and get some feedback. Of course, it will also make it easier for us to understand what you have accomplished.