CSCI 432 Course Project

Instructor: Brittany Terese Fasy

August 29, 2016

Abstract

1 Overview

In this project, groups of two to five students will collaborate to investigate an algorithm from a recent (past 30 years or so) conference or journal. Each group will choose the topic on their own, and become experts on this algorithm. In addition, they will take the paper that they found one step further in the plus-one (+1) element of their project.

2 Deadlines

All deadlines are by 23:59 Mountain time on the date given (except for the in-class presentations and discussion). Although not mandatory, I encourage all teams to schedule a time to meet with me or a TA 1-1 between the P-1 and P-3 deadlines.

(P-1) due 9 Sept, 1 point Form teams of two to five people. Note: it is expected that these teams will be fixed for the semester; however, changes can be made if needed.

(P-2) due 5 Oct, 9 points Initial investigation: Investigate 3 algorithms. For each algorithm, you must identify the problem being solved in a 1-2 page write-up.

(P-3) due 19 Oct, 10 points Selection of algorithm: You must select the algorithm that you will be investigating. This does not need to be one of the algorithms discussed in P-2, but is likely to be one of those. For this stage of the project, the deliverable is a 1-2 page write-up identifying the problem being solved and providing pseudo-code for the algorithm itself. Not all steps need to be fully explained, but the general essence of the algorithm should be clear.

(P-4) due 9 Nov, 10 points Identification of what the +1 element of your project will be: Suggestions of ways to do a +1: improve the algorithm in some way (speed, clarity), explain the analysis of the algorithm in a different way than provided in the paper, provide a runtime analysis if there is none, compare it against 2-3 other algorithms, walking through an example or two, explain the algorithm more clearly/concisely than the original publication, thoroughly explaining one of their proofs, etc. You are free to be as creative as you want for your +1 element, but I highly suggest you discuss this with myself or a TA before this deadline. Your +1 is going to be the focus of your video, so choose wisely.

(P-5) due 23 Nov, 10 points Demonstration of progress: Prepare a 1 page (2 if you REALLY need it) document outlining (a) what you have done so far and (b) what is left to be done. The grade for this component of the project will be 5 points for significant progress and 5 points for clarity and completeness of the document.

(P-6) due TBA, 50 points Video presentations: Each group will present their video. Each student will fill out a peer evaluation and hand that in. Each group member will turn in a one-page summary detailing their contributions, the strengths / weaknesses of their project, and explain one thing that would be done differently if you could do it again.

(P-7)) Final, 10 points Peer evaluation: We will discuss the peer-evaluations that were filled out during the video presentations. You will not be graded on how well your peers evaluate your project, but rather, how well you evaluate their projects. Mock PC's (to be assigned / formed later in the semester) will bring a summary of all peer evaluations to the final meeting time.

3 Extra Credit

Up to ten points of extra credit can be earned for using version control of LaTex documents for the non-video deliverables of this project. The deliverables should be either .txt or .tex files maintained on a mercurial or git repository hosted on bitbucket or github. Microsoft word documents do not play nicely with version control. (Note: getting 10 points means that you are an expert in LaTex and either mercurial or git, and that youve demonstrated that. While expertise is a high standard, at least 1 point will be given for all who at least attempt to use version control and LaTex.)

4 Project Suggestions

List of possible papers from which to base a project:

1. Ahmed, Fasy, Wenk. Path-Based Distance for Street Map Comparison. ACM TSAS, 2015.

2. Aloupis, Fevens, Langerman, Matsui, Mesa, Nunez, Rappaport, and Toussaint. Algorithms for Computing Geometric Measures of Melodic Similarity. Computer Music Journal, 2006.

3. Bethea and Reiter. Data Structures with Unpredictable Timing. ESORICS, 2009.

4. Chambers, de Verdire, Erickson, Lazard, Lazarus, and Thite. Homotopic Frchet Distance between Curves or, Walking your Dog in the Woods in Polynomial Time. Computational Geometry, 2010.

5. Chan. Optimal output-sensitive convex hull algorithms in two and three dimensions. Discrete and Computational Geometry, 1996.

6. Chen and Kerber. Persistent Homology Computation with a Twist. EuroCG, 2011.

7. Chazal, Fasy, Lecci, Rinaldo, Singh, and Wasserman. On the Bootstrap for Persistence Diagrams and Landscapes. Modeling and Analysis of Information Systems, 2013. CSCI 432 COURSE PROJECT 3

8. Edelsbrunner and Guibas. Topologically Sweeping an Arrangement. Journal of Computer and System Sciences, 1989.

9. Alon, Babai, and Itai. A Fast and Simple Randomized Parallel Algorithm for the Maximal Independent Set Problem/. Journal of Algorithms, 1986.

10. Guibas, and Oudot. Reconstruction Using Witness Complexes. DCG, 2008.

11. Guttman. R-Trees: A Dynamic Index Structure for Spatial Searching. SIGMOD, 1984.

12. Li and Svensson. Approximating k-Median via Pseudo-Approximation. STOC, 2013.

13. Liota. Low Degree Algorithms for Computing and Checking Gabriel Graphs, Brown University Technical Report, 1996.

14. Miller and Sheehy. Approximate Centerpoints with Proofs. CGTA, 2010.

15. Millman, Love, Chan, and Snoeyink. Computing the Nearest Neighbor Transform Exactly with only Double Precision. ISVD, 2012.

16. Milosavljevic, Morozov, and Skraba. Zigzag Persistent Homology in Matrix Multiplication Time. SoCG, 2011.

17. Pagh, Rodler, and Friche. Cuckoo Hashing. ESA, 2001

18. Shonkwiler. An Image Algorithm for Computing the Hausdorff Distance Efficiently in Linear Time. Information Processing Letters, 1989.

19. Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM Journal on Computing, 1997.

20. Oostrum and Veltkamp. Parametric Search Made Practical. SoCG 2002.

21. Wang, Wang, and Li. Efficient Map Reconstruction and Augmentation via Topological Methods. ACM SIGSPATIAL GIS, 2015.

22. Zheng, Jestes, Phillips, Li. Quality and Efficiency in Kernel Density Estimates for Large Data. ACM Conference on the Management of Data (SIGMOD), 2013.

The list of papers above is only a sprinkling of recent developments in algorithms. I encourage teams to peruse the following (non-exhaustive) list of venue for finding other papers:

- ESA: European Symposium on Algorithms.
- EuroCG: European Workshop on Computational Geometry.
- GD: International Symposium on Graph Drawing.
- ISAAC: International Symposiums on Algorithms and Computation.
- LATIN: Latin American Theoretical INformatics.
- SIGSPATIAL GIS
- SoCG; Symposium on Computational Geometry.
- SODA: Symposium on Discrete Algorithms.
- STOC: ACM Symposium on Theory of Computing.
- SWAT: Scandinavian Workshop on Algorithm Theory.
- WADS: Algorithms and Data Structures Symposium.