CMSC 858F: Algorithmic Lower bounds: Fun with Hardness Proofs

Prof: Mohammad T. Hagiaghagi "hardness made easy"

http: www.cs.Umd.edu/~hagiagha/ALB14/ALB.html

Sister Course by Erik Demaine at MIT (Lectures are taught separately)

See the course Agenda first

what is this class?

- Practical guide to proving computational problems are formally hard
- Not a complexity course but we use/refer to needed materials intractable
- (anti) algorithmic perspective.
- another name for the class can be Assumptions & Reductions
- Lots of researchers design algorithms for different settings such as Counting Algorithms, Approximation Algorithms, Fixed Parameter Algorithms, streaming Algorithms, parallel Algorithms Algorithmic Game theory, Geometric problems, online Algorithms, puzzle Algorithms However sometime we cannot design algorithms with certain efficiency in time, spac, approximation. In this course we want to understand why there are such limits.
- We also consider the common techniques to prove hardness of Algorithms for different settings.
- We believe every CS grad student esp. in theory should have a basic Knowledge about hardness of algorithms at the level of this course
- In short, while in typical Advanced Algorithm courses, we learn design of algorithms for different settings, in this course we learn why we cannot design algorithms with certain guarantees in different settings

why take this course?

- know the limits in algorithm design
- master techniques for proving hardness < Key problems & Assumptions; Key Reductions < Proof styles, Gadgets
- Cool connections between problems
- fun problems too solve (even some puzzle problems → potentially publishable papers

Background: algorithms, asymptotics, combinatorics, some complexity, but not much background

Requirements: Passing 451 or equivalent course before. Having passed a grad course in Algorithms is a plus

3-4 sets of Assignments
scribe notes
class participation (esp. attend lectures)
one exam
Research-based projects (survey and new theory)
class presentation

Topics: (on the webpage as well)
- NP-completeness (3SAT, 3-Partition, Hamiltonicity, Geometry, 3-coloring)
- PSPACE, EXPTIME
- Inapproximability (PCP, APX, set cover, Label cover, UGC, ...)
- Fixed parameter intractability (W, clique, ...)
- 3SUM (toward n^2) and All pairs shortest path (toward n^3)
- Counting (#P) & uniqueness (ASP)
- Algorithmic game theory (PPAD)
- Existential theory of reals, undecidability (if time)
- streaming algorithms lower bounds (indexing, set disjointness
- online algorithms lower bounds (matching, set cover, K-server)

References: There is no text book for this class, but there are some references on the website

→ We often focus on graph-like problems in this course