0368.416701 Sublinear Time Algorithms	October 19, 2009

Course Information

Lecturer: Ronitt Rubinfeld

Lecture: Mon 13:00-16:00, Room to be determined.

Instructor: Ronitt Rubinfeld, `ronitt@cs.tau.ac.il`, Schreiber 326.

Course topics In this course we will study techniques for designing algorithms that output a solution by only considering a very small portion of the data, in particular, algorithms which run in sublinear time. The study of sublinear time algorithms has been applied to problems from a wide range of areas, and many beautiful techniques have been applied to analyzing such algorithms, including the Szemeredi Regularity Lemma, low rank approximations of matrices, Fourier analysis, and other tools from number theory, combinatorics, linear algebra, optimization theory and probability theory.

For most problems, sublinear time algorithms must necessarily return an approximate answer. We will consider both classical approximation algorithms, as well as property testing algorithms. We will also consider natural extensions of these notions to estimating quantities associated with distributions over large domains. A detailed list of topics to be covered is given below. The list is subject to change due to my personal whims, class interest, timing and other constraints.

- Combinatorial objects
  - Graphs: Characterizing dense graph properties that are testable in constant time. Szemeredi Regularity lemma. Algorithms for sparse graph properties: connectivity, low diameter, triangle-freeness. Estimating vertex cover and matching size. Estimating graph parameters. Estimating minimum spanning tree weight.
  - Testing of clusterability and other metric properties of points
  - Strings: Estimating edit distance, compressibility.
- Testing properties of functions
  - Monotonicity: over linearly ordered, hypercube and general poset domains.
  - Convexity
  - Algebraic properties: Linearity and low total degree polynomial testing. Locally testable codes.
  - Dependence on few variables: Singletons, Juntas
  - Linear Threshold Functions
- Estimating parameters of distributions (given by samples): Testing uniformity and closeness of two distributions. Testing monotonicity, independence. Estimating entropy.

Course Requirements Approximately 4 homework sets (60%). Scribe notes(25%). Class participation (15%). As part of class participation, students will be asked to help with grading of assignments and writing solution sets.

Prerequisites An undergraduate algorithms course.