CMPSCI 690RA: Randomized Algorithms

Lecture 1: Introduction

Andrew McGregor

Last Compiled: January 28, 2020

Outline

Introduction to Randomized Algorithms and Examples

Course Outline and Administrivia

Randomized Algorithms?

- A randomized algorithm is an algorithm whose steps are based both on the input and the flips of a coin (a.k.a., a string of random bits).
- What's great about randomized algorithms?
  - *Simplicity:* Algorithms and analysis are often simple and elegant, e.g., randomized quick-sort...
  - *Speed:* Some randomized algorithms are faster than the best known deterministic algorithms, e.g., checking if a multivariate polynomial is the zero-polynomial...
  - *Defeating Adversaries!* Imagine playing rock, paper, scissors without randomization... The "adversary" might be the future in the case of dynamic, streaming, or online algorithms.
- What's not so great about randomized algorithms?
  - *Errors:* May return the wrong answer with small probability.
  - *Running Time:* Sometimes we only know the expected running time and the algorithm may not terminate
  - *Debugging:* Bugs might be hard to reproduce.

Course Topics

1. Classic Randomized Algorithms Topics: Randomized Rounding of Linear Programs, Probabilistic Method and Lovasz Local Lemma, Monte Carlo Simulations and MCMC, Finger Printing and Pattern Matching, Derandomization and Randomness Extraction.
2. Probability Topics: Tail bounds, Markov Chains, Martingales
3. Randomized Topics in Big Data: Hashing and Load Balancing, Sub-linear Time Algorithms and Property Testing, Linear Sketches and Data Streams, Distributed Algorithms
4. Let's see some representative examples...

2-SAT and Random Walks

- An algorithm for 2-SAT:
  1. Pick arbitrary assignment.
  2. Pick an unsatisfied clause: randomly flip the value assigned to one of the two variables.
  3. Repeat Step 2 until there are no unsatisfied clauses.
- How long until we terminate?
- Ideas for Analysis:
  - Let $x^{(t)}$ be the assignment at time t.
  - Fix some satisfying assignment y and define random variable $X^{(t)}$ be the number of values for which y and $x^{(t)}$ agree.
  - $X^{(t+1)} = X^{(t)} \pm 1$ and

$$\mathbb{P}\left[X^{(t+1)} = X^{(t)} + 1\right] \geq 1/2$$

  - Can analyze time until $X^{(t)} = n$ via Markov Chains where n is the number of variables.
- Answer turns out to be ... $O(n^2)$ rounds.

k-SAT, Probabilistic Method, and Lovasz Local Lemma

- Consider an instance ϕ of k-SAT: There are m clauses and each is the OR of k literals.
- If $m < 2^k$ we can show ϕ must be satisfiable via the union bound: if we randomly assign variables, then probability there exists an unsatisfied clause is $\leq m/2^k < 1$
- Suppose each clause shares variables with at most d other clauses. Can show via the Lovasz Local Lemma that if $e(d+1) \leq 2^k$ then there is a satisfying assignment.

Uniformity Testing and Distributional Property Testing

- Suppose you have access to samples from unknown distribution p on

$$\{1, 2, \ldots, n\}$$

- Design an algorithm with low sample complexity such that:
  - If $p = (1/n, 1/n, \ldots, 1/n)$, then algorithm accepts with prob. $\geq 3/4$.
  - If $\sum_i |p_i - 1/n| \geq \epsilon$, then algorithm rejects with prob. $\geq 3/4$.
- Best result is $O(\sqrt{n}/\epsilon^2)$. This is a lot fewer than the $\Omega(n/\epsilon^2)$ that are required to learn p up to sufficient accuracy.

Sublinear Time Algorithms

- Let G be a graph with n nodes, max degree d, and every edge has a weight in range $\{1, 2, \ldots, w\}$.
- It's possible to approximate the weight of the min spanning tree up to a factor $1 + \epsilon$ in $O(dw\epsilon^{-2} \log(dw/\epsilon))$ time.
- This is much less time than reading the entire input!
- Related results for a range of other graph problems including vertex cover, set cover, matching... Techniques are based on sampling and exploit connections to distributed algorithms.

Communication Complexity

- Suppose Alice has n bit number x and Bob has n bit number y.
- How many bits to need to be communicated to determine if $x = y$?
  - If no randomness is allowed $\Omega(n)$ bits is required.
  - If randomness is allowed, $O(\log n)$ bits suffice: Alice randomly picks one of the first $2n$ primes p and sends $(x \bmod p)$.
- Other Questions How many bits need to be communicated to determine $x > y$ or other functions of x and y?

Load Balancing and Hashing

- Suppose there are n bins and n balls.
- If you throw each ball into a random bin, with high probability the number of balls in the heaviest bin is

$$(1 + o(1))\frac{\log n}{\log \log n}$$

- Suppose before each ball you pick two random bins and throw the ball into the lightest of the two bins. Then the heaviest throw each ball into a random bin, then the number of balls in the heaviest bin is

$$\log \log n + O(1)$$

- How can we design hashing schemes that harness "the power of two choices" phenomena? For example, we'll explore cuckoo hashing and associated random graph analysis.

Outline

Introduction to Randomized Algorithms and Examples

Course Outline and Administrivia

Basic Stuff

Lectures: Tuesday and Thursday, 10am to 11.15am in CMPS 140.

Lecturer: Professor Andrew McGregor

- Email: mcgregor@cs.umass.edu
- Office: CMPS 334
- Office hours: Tuesday 11:30 - 12:30, or by appointment.

Textbooks and Materials

Optional Textbooks:

- R. Motwani and P. Raghavan, Randomized Algorithms. Cambridge University Press, 1995.
- M. Mitzenmacher and E. Upfal, Probability and Computing: Randomized Algorithms and Probabilistic Analysis. 2nd Edition. Cambridge University Press, 2017.

Other materials, including lecture slides, will be posted at:

`https://people.cs.umass.edu/~mcgregor/CS690RA20/index.html` .

Discussion on Piazza and homework submissions to Gradescope.

Assessment

- *Homeworks:* Three assignments will contribute 30% to grade. Collaboration is allowed in groups of at most three.
- *Exams:* There will be two exams. No collaboration is permitted. The exams will each contribute 25% to grade.
- *Participation:* Remaining 20% of the grade will be based on class participation, contributions in Piazza, and a short project.