Randomized Algorithms

- Deterministic and (Las Vegas & Monte Carlo) Randomized Algorithms
- Probability Review
- Probabilistic Analysis of deterministic QUICK-SORT Algorithm
- RANDOMIZED-SELECT and RANDOMIZED-QUICK-SORT
- Max-Cut
- Min-Cut
- MAX-3-SAT and Derandomization
- Closest pair
- Hashing, Bloom filters, Streams, Sampling, Reservoir sampling, Sketch

IMDAD ULLAH KHAN

The MAX-3-SAT Problem

- Given n Boolean variables $x_1, \ldots, x_n$
- Each can take a value of $0/1$ (true/false)
- A literal is a variable appearing in some formula as x_i or $\bar{x}_i$
- A clause of size 3 is an OR of three literals
- A 3-CNF formula is AND of one or more clauses of size ≤ 3
- A formula is satisfiable if there is an assignment of $0/1$ values to the variables such that the formula evaluates to 1 (or true)

3-SAT(f) problem: Is there a satisfying assignment for 3-CNF formula f?

MAX-3-SAT(f) problem: Find an assignment for 3-CNF formula f that satisfies the maximum number of clauses

MAX-3-SAT

MAX-3-SAT(f) problem: Find an assignment for 3-CNF formula f that satisfies the maximum number of clauses

- The problem is NP-HARD
- Brute Force: Try all 2^n possible assignments in $\mathcal{O}(m2^n)$

$\triangleright$ m is the number of clauses

MAX-3-SAT(f) problem: Find an assignment for 3-CNF formula f that satisfies the maximum number of clauses

Randomized Algorithm

- Simple Idea: Toss a coin, and independently set each variable true with probability $1/2$
- What is the expected number of clauses satisfied by a random assignment?

MAX-3-SAT

A random assignment to variables satisfies in expectation $^{7m}/_8$ clauses of a 3-CNF formula f with m clauses

Let random variable $Z_j = \begin{cases} 1 & \text{if clause } C_j \text{ is satisfied} \\ 0 & \text{otherwise} \end{cases}$

$E[Z_j] = Pr[C_j \text{ is satisfied}] = 1 - Pr[C_j \text{ is not satisfied}]$

C_j is not satisfied when all literals in C_j are set to FALSE (independently)

Thus, $Pr[C_j \text{ is not satisfied}] = (^1/_2)^3 = {}^1/_8$ $\triangleright$ $E[Z_j] = {}^7/_8$

Let Z be the number of clauses satisfied by random assignment

$$E[Z] = \sum_{j=1}^{m} E[Z_j] = \sum_{j=1}^{m} \frac{7}{8} = \frac{7m}{8}$$ $\triangleright$ linearity of expectation

MAX-3-SAT Las Vegas $^7/_8$-Approximation

For any instance of MAX-3-SAT with m clauses, there exists a truth assignment which satisfies at least $^{7m}/_8$ clauses

- There is a non-zero probability that a random variable takes the value of its expectation
- $Pr[Z \geq E[Z] = {}^{7k}/_8] > 0$

Probabilistic Method:

Prove the existence of a non-obvious property by showing that a random construction produces it with positive probability

MAX-3-SAT Las Vegas ($7/8$)-Approximation

- Is there a $7/8$ Las Vegas approximation algorithm for MAX-3-SAT?
  - guaranteed to find an assignment satisfying at least $7m/8$ clauses
  - expected runtime is polynomial
- Standard trick: Repeatedly generate a random assignment A to variables until A satisfies at least $7m/8$ clauses
- Suppose $Pr\left[A \text{ satisfies} \geq 7m/8 \text{ clauses}\right] \geq p$
- Then, expected number of trials to find this assignment is $1/p$

  ▷ Expectation of geometric random variable
- If p is polynomial, then expected running time is polynomial

MAX-3-SAT Las Vegas ($7/8$)-Approximation

Probability p that a random assignment satisfies $\geq 7m/8$ clauses is $\geq 1/8m$

p_j : probability that the random assignment satisfies exactly j clauses

$\triangleright\ j = 1, 2, \cdots, m$

Lower bound on p using $E[Z] = 7m/8$

$$E[Z] = \sum_{j=0}^{m} j\,p_j = \sum_{j<7m/8} j\,p_j + \sum_{j\geq 7m/8} j\,p_j \leq \frac{7m-1}{8} \sum_{j<7m/8} p_j + m \sum_{j\geq 7m/8} p_j$$

$$\implies E[Z] \leq {7m-1}/8 \cdot 1 + m\cdot p \implies 7m/8 \leq {7m-1}/8 + mp \implies p \geq 1/8m$$

MAX-3-SAT cannot be approximated in polynomial time to within a ratio greater than $7/8$, unless P=NP $\triangleright$ [Håstad 1997]

MAX-3-SAT: Derandomization

- Random choices by an algorithm sometimes happen to be 'good' i.e.
  - The randomized algorithm output is close to the optimal
- Can these 'good' choices be made deterministically?
- Derandomization: Transforming a randomized algorithm into a deterministic algorithm
- Can the $7/8$-approx Las Vegas Algorithm for MAX-3-SAT be derandomized?
- How do we know which set of choices for variable assignments is 'good'? i.e. satisfies greater number of clauses
- Idea: Consider the choice for each variable (True/False) one by one

MAX-3-SAT : Derandomization

- Let Z be the number of clauses satisfied

Given assignments for the "first i" variables $x_1 = a_1 \cdots , x_i = a_i$, the expected value of Z with random assignment of the unassigned variables $x_{i+1}, \cdots , x_n$ can be computed in polynomial time

- Given assignment to a variable, for each clause C_j if the corresponding literal evaluates to
  - FALSE, then remove it from C_j
  - TRUE, then ignore the clause as it is satisfied
- Conditional expectation of Z is the unconditional expectation of Z in the reduced set of clauses <u>plus the number of already satisfied clauses</u>
- This yields a polynomial time deterministic algorithm for MAX-3-SAT

MAX-3-SAT : Derandomization

- Let Z be the number of clauses satisfied
- Fix an order of variables $x_1, x_2, \cdots, x_n$
- For $i = 1$ to n
  - If $E\left[Z | x_1 = a_1, \cdots, x_{i-1} = a_{i-1}, x_i = \text{TRUE}\right]$
    $> E\left[Z | x_1 = a_1, \cdots, x_{i-1} = a_{i-1}, x_i = \text{FALSE}\right]$,
    - then set x_i to TRUE
    - else set x_i to FALSE
- Since $E\left[Z | x_1 = a_1, \cdots, x_i = a_i\right] \ \geq \ E\left[Z\right] \quad$ for $\quad 1 \leq i \leq n$
- And $E\left[Z\right] = {}^{7m}/_{8}$
- Thus, $E\left[Z | x_1 = a_1, \cdots, x_i = a_i\right] \geq {}^{7m}/_{8}$
- Derandomized algorithm satisfies at least ${}^{7m}/_{8}$ clauses