Fall 2014 Randomized Algorithms Deadline: Nov 12, 2014

Problem Set 3

Prof. Friedrich Eisenbrand *Assistant: Chidambaram*

Note: Problem 6 in this set is a bonus problem. The points for this problem, if you attempt it, will be added to your final score over all problem sets. You may also skip it as the maximum points for this problem set is only 24 points.

1. Is the Schwartz-Zippel lemma tight? Justify your answer. **[1 pt]**

2. (a) Let A, B, and C be matrices in $\mathbb{R}^{n\times n}$. To check if $AB \stackrel{?}{=} C$ we perform the following test: We pick a random $r \in \{0,1\}^n$ and test if $A(Br) \stackrel{?}{=} Cr$. Show that this check succeeds in detecting an incorrect matrix product with probability at least $1/2$. **[3 pts]**

   (b) Construct a multivariate polynomial Q such that $Q \equiv 0$ if and only if $AB = C$ and use polynomial identity testing to derive a result similar to the first part of this problem. **[2 pts]**

3. Two $n \times n$ matrices A and B over $\mathbb{Z}_2$ are said to be similar if there exists a non-singular T such that $TAT^{-1} = B$. Devise a randomized algorithm for testing the similarity of the matrices A and B. **[6 pts]**

   Hint: View the entries of T as variables and write the constraints as a system of homogenous linear equations in the variables. Then reduce the problem to a polynomial identity testing problem.

4. Prove that the rank of the Tutte matrix is twice the size of a maximum matching in a graph (the "rank" here refers to largest r such that there is a $r \times r$ submatrix whose determinant is not the zero polynomial). **[4 pts]**

   Hint: Let A be an $n \times n$ skew symmetric matrix of rank r. For any two setes $S, T \subseteq [n]$ we denote by A_{ST} the submatrix of A indexed by rows S and columns T. For any two sets S, T of size r show

   $$\det(A_{ST})\det(A_{TS}) = \det(A_{TT})\det(A_{SS}).$$

   In class we saw how to obtain a randomized $O(n^{\omega+2}\log n)$ algorithm to find a maximum matching in a graph with high probability. The result of this problem generalizes Tutte's theorem and also saves a $O(\log n)$ factor in the time complexity this algorithm. (Note: To compute determinants, inverses and solutions of linear systems in $O(n^\omega)$ time we use the fact that the LU-decomposition of a matrix can be computed in matrix multiplication time [BH74]).

5. The Rabin-Vazirani algorithm uses additional properties of the Tutte matrix to find a perfect matching in $O(n^{\omega+1})$ time. Let A be the Tutte matrix associated with the input graph $G = (V, E)$. Recall the determinant formula in terms of the minors of the first row of $A \in \mathbb{R}^{n \times n}$,

$$\det(A) = \sum_{j=1}^{n} (-1)^{1+j} A_{1,j} A[\{1\}, \{j\}].$$

We use the notation $A[X, Y]$ to denote the submatrix of A obtained by removing rows and columns indexed by $X \subseteq [n]$ and $Y \subseteq [n]$ respectively.

(a) If $\det(A) \not\equiv 0$ then there is a $j \in [n]$ so that $A_{1,j} A[\{1\}, \{j\}] \not\equiv 0$ and thus $\{1, j\} \in E$ and $G \setminus \{1, j\}$ also contains a perfect matching. The main observation of the Rabin-Vazirani algorithm is that in this case $\det(A[\{1, j\}, \{1, j\}]) \not\equiv 0$ so that we have a perfect matching in $G - \{1\} - \{j\}$ also. Prove this. **[4 pts]**

(b) Since the (i, j)-th entry of the inverse is given by

$$A^{-1}_{i,j} = \frac{(-1)^{i+j}}{\det(A)} \det(A[\{i\}, \{j\}]),$$

$\det(A[\{i\}, \{j\}]) = 0 \iff A^{-1}_{i,j} = 0$. This leads to the following natural algorithm for computing a perfect matching of G:

Algorithm 1 The Rabin-Vazirani algorithm

---

procedure FINDPERFECTMATCHING(G)
Sample s_{ij} from $[0, n^2]$ for every $\{i, j\} \in E$.
Sample a prime p uniformly at random from the first $\lceil 3n^3 \log_2 n \rceil$ primes
$B \leftarrow A_G(\{s_{ij}\}_{\{i,j\} \in E}) \mod p$.
if $\det(B) \equiv 0 \mod p$ **then return** No perfect matching exists.

$M \leftarrow \emptyset$.
while $|M| < n/2$ **do**
Compute $B^{-1} \mod p$ in $O(n^\omega)$ time.
Find j such that $B^{-1}_{1j} \not\equiv 0 \mod p$ and $\{1, j\} \in E$.
$M \leftarrow M \cup \{1, j\}$.
$B \leftarrow B[\{1, j\}, \{1, j\}]$.

We make all computations modulo some prime p to ensure our algorithms run in polynomial time. Obtain a crude upper bound on the random variables $|B^{-1}_{1j}|$. How many prime factors can it have? Complete the analysis of the algorithm by showing that it succeeds with high probability. **[4 pts]**

6. **(Bonus Problem)** In this problem we will construct a randomized *parallel* algorithm to find a perfect matching in a graph using similar ideas.

First we need to prove a useful lemma called the isolating lemma which ensures that all the parallel processors work towards computing the same perfect matching without having to coordinate explicitly.

(a) Suppose you are given a family $\mathcal{F}$ of subsets of $\{1, \ldots, m\}$ and the weights of the elements are chosen uniformly at random from $\{1, \ldots, 2m\}$. Show that the weight of smallest weight subset in $\mathcal{F}$ is unique with probability at least $1/2$. **[4 pts]**

Hint: For $x \in \{1, \ldots, m\}$ consider the function

$$\alpha(x) = \min_{S \in \mathcal{F}: x \notin S} w(S) - \min_{S \in \mathcal{F}: x \in S} w(S \setminus \{x\}).$$

(b) Thinking of the elements are edges in G and the set of $\mathcal{F}$ to be all the perfect matchings in G, we can apply the isolating lemma to guarantee that that there is a unique minimum weight perfect matching in G with probability at least $1/2$ in the following algorithm.

Algorithm 2 The Mulmuley-Vazirani-Vazirani algorithm

procedure ParallelPerfectMatching(G)
  Sample w_{ij} from $[0, 2n^2]$ for every $\{i, j\} \in E$.
  $B \leftarrow A_G(\{2^{w_{ij}}\}_{\{i,j\} \in E})$.

  Calculate 2^w, the largest power of 2 that divides $\det(B)$
  $M \leftarrow \emptyset$
  for $\{i, j\} \in E$ **do**
    Compute in parallel $t_{ij} = \det(B[\{i\}, \{j\}]) \frac{2^{w_{ij}}}{2^w}$.
    if $t_{ij} \equiv 1 \mod 2$ **then**
      $M \leftarrow M \cup \{i, j\}$
  if M is a perfect matching **then return** M
  else
    Fail.

Assuming that the minimum weight perfect matching M_0 (according to the weights w) is unique, prove that an edge $\{i, j\} \in M_0$ if and only if t_{ij} is odd. Note that we replace x_{ij} in the Tutte matrix of G by $2^{w_{ij}}$ for an edge $\{i, j\} \in E$. **[4 pts]**

References

[BH74] James R Bunch and John E Hopcroft. Triangular factorization and inversion by fast matrix multiplication. *Mathematics of Computation*, 28(125):231–236, 1974.