Prof. Friedrich Eisenbrand
Alfonso Cevallos

Discussions from: March 31, 2015

Computer Algebra

Spring 2015

Assignment Sheet 4

Exercises marked with a $\star$ can be handed in for bonus points. Due date is April 14.

Exercise 1
Let $A \in \mathbb{Q}^{n\times n}$, and denote its columns by $a_1, \cdots, a_n$, and let B be an upper bound on the absolute values of the entries of A.

1. Give a formal proof of the Hadamard bound $|\det(A)| \leq \prod_{j=1}^{n} |a_j|_2$, where $|\cdot|_2$ is the Euclidean norm. (Hint: use the Gram-Schmidt orthogonalization process.) Derive from this that $|\det(A)| \leq n^{n/2} B^n$.

2. Prove Leibniz's bound $|\det A| \leq B^n n!$. How does it comparte to Hadamard bound?

Exercise 2
Prove that for a matrix $A \in \mathbb{Z}^{n\times n}$, where all the entries are bounded in absolute value by a constant B, we have $\log|\det(A)| = O(n\log n + n\log B)$. Then, use this fact to prove that the bit complexity of performing Gaussian elimination is polynomial in the bit size of the input.

Exercise 3 ($\star$)

$$A = \begin{pmatrix} 1 & 0 & -2 \\ 2 & -1 & 1 \\ 0 & 2 & 2 \end{pmatrix}$$

For the matrix A above, use Gaussian elimination modulo p to compute $\det(A) \mod p$, for $p = 3, 5, 7$. Then, use Leibniz bound to obtain an upper bound on the quantity $2|\det(A)| + 1$, and use this to conclude that you can directly obtain $\det(A)$ from the previous results.

Exercise 4
Show that, using Gaussian elimination, one can compute a solution to the system $Ax = b$, $A \in \mathbb{Q}^{m\times n}, b \in \mathbb{Q}^m$, or assert that none exists, in time polynomial in the bit size of A and b.

Exercise 5 ($\star$)

Let $T(G)$ be the Tutte matrix of a graph G, and $\nu(G)$ the cardinality of the maximum matching of G.

1. For a given a graph G, show that there exists a subgraph H of G with a perfect matching such that $2\nu(G) = 2\nu(H) = \text{rank}(T(H)) = \text{rank}(T(G))$.[1]

2. Describe (and prove the correctness of) an efficient randomized algorithm for computing $\nu(G)$, that outputs the correct answer with probability at least $1/2$.

Exercise 6 ($\star$)

Implement an algorithm that takes as input the adjacency matrix of a graph, and then uses the Tutte matrix and the Schwartz-Zippel lemma to find a perfect matching, or assert that none exists.

[1] Note that this does not immediately imply that we can compute $\nu(G)$, because we still have to show how to compute the rank of a matrix with indeterminate entries. You can assume however that you can efficiently find the rank of a matrix with numerical entries.