Algorithmic Combinatorics WS 2014 / 2015
Tibor Szabó
Shagnik Das

Exercise Sheet 13

Due date: Feb 6th, 2:00 PM, tutor box of Shagnik Das
Late submissions will be snubbed worse than Leonardo DiCaprio at the Oscars.

You should try to solve and write up all the exercises. You are welcome to submit **at most** three neatly written exercises for correction each week. You are encouraged to submit in pairs, but please indicate the author of each solution. Each problem is worth 10 points.

Exercise 1. Let $\mathcal{J}$ be a family of d-intervals not containing three pairwise-disjoint d-intervals; that is, there are no $J_1, J_2, J_3 \in \mathcal{J}$ with $J_i \cap J_j = \emptyset$ for every $1 \leq i < j \leq 3$. Show that $\mathcal{J}$ has a transversal of size $4d^2$.

Exercise 2. Let $k \geq 1$ be some integer, and let A and B be two $2^k \times 2^k$ matrices. We wish to efficiently compute the product $C = AB$. We express these as block matrices:

$$A = \begin{pmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{pmatrix}, B = \begin{pmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{pmatrix}, \text{ and } C = \begin{pmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{pmatrix}.$$

We now define some new matrices:

$$\begin{aligned} &M_1 = (A_{1,1} + A_{2,2})(B_{1,1} + B_{2,2}), && M_2 = (A_{2,1} + A_{2,2})B_{1,1}, && M_3 = A_{1,1}(B_{1,2} - B_{2,2}), \\ &M_4 = A_{2,2}(B_{2,1} - B_{1,1}), && M_5 = (A_{1,1} + A_{1,2})B_{2,2}, && M_6 = (A_{2,1} - A_{1,1})(B_{1,1} + B_{1,2}), \\ & && \text{and} && M_7 = (A_{1,2} - A_{2,2})(B_{2,1} + B_{2,2}). \end{aligned}$$

(i) Express the blocks $C_{i,j}$ in terms of the blocks $A_{i,j}$ and $B_{i,j}$.

(ii) Verify the following identities:

$$C_{1,1} = M_1 + M_4 - M_5 + M_7, \; C_{1,2} = M_3 + M_5, \; C_{2,1} = M_2 + M_4, \text{ and } C_{2,2} = M_1 - M_2 + M_3 + M_6.$$

(iii) One can reuse these identities to calculate the products in the definition of the matrices M_i, leading to a recursive algorithm for computing the product $C = AB$. Estimate the running time (in terms of the number of arithmetic operations) of this algorithm.

(iv) For general integers $n \geq 1$, how can this algorithm be applied to $n \times n$ matrices?

Exercise 3. In this exercise, $\mathbb{F}$ is an arbitrary field, but you may assume $\mathbb{F} = \mathbb{Q}$ if you like.

(i) Show that the Schwartz–Zippel theorem can be tight. That is, show that for any $S \subseteq \mathbb{F}$ with $|S| \geq d$, there is a polynomial in n variables of degree d with exactly $d\,|S|^{n-1}$ roots $(r_1, \ldots, r_n) \in S^n$.

(ii) Prove the following generalisation of the Schwartz–Zippel theorem: For any non-zero polynomial $p \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of degree at most d, and subsets $S_1, S_2, \ldots, S_n \subset \mathbb{F}$, each of size s, the number of roots

$$\{(r_1, r_2, \ldots, r_n) \in S_1 \times S_2 \times \ldots \times S_n : p(r_1, r_2, \ldots, r_n) = 0\}$$

is at most ds^{n-1}.

Exercise 4. You give your friend two $n \times n$ matrices A and B to multiply, and he tells you the answer is C. To check that he is correct, you run the randomised verification algorithm, multiplying both C and AB by a random vector $\vec{x} \in \{0,1\}^n$.

(i) How many times do you have to run the algorithm to have at least 95% confidence in the outcome?

Suppose you run the algorithm, and find that $C\vec{x} \neq AB\vec{x}$. This proves the *existence* of a mistake. However, to complain to your friend, you would like to explicitly *find* a mistake; that is, find some i and j such that $C_{ij} \neq (AB)_{ij}$.

(ii) How many more arithmetic operations will this take?

To save time, instead of choosing a random vector $\vec{x} \in \{0,1\}^n$, you take a random vector $\vec{x} \in \{0, 1, \ldots, N\}^n$ instead.

(iii) If indeed $C \neq AB$, what now is the probability of your algorithm accepting C as being correct? What are the drawbacks to this approach?

Exercise 5. Suppose we have some univariate polynomial $p \in \mathbb{F}[x]$ of degree at most d that we only have oracle access to, so that for any input $y \in \mathbb{F}$, we are told the value $p(y)$.

(i) Explain how we can determine the coefficient of y^k with at most $d+1$ oracle queries.

Suppose now we have a bipartite graph $G = (U \cup V, E)$ with $|U| = |V| = n$, and suppose further that the edges E are coloured red or blue.

(ii) Using part (i), explain how we can extend the randomised perfect matching testing algorithm to test whether or not there is a perfect matching of G with exactly k red edges.

Exercise 6. This exercise will show you how to extend our randomised perfect matching algorithm for bipartite graphs to general graphs. Let $G = ([n], E)$ be a graph on n vertices.

We introduce a new variable x_{ij} for each edge $\{i, j\} \in E$. The *Tutte matrix* A of the graph G is defined as $A = (a_{ij})_{i,j \in [n]}$, where

$$a_{ij} = \begin{cases} +x_{ij} & \text{if } i < j \text{ and } \{i, j\} \in E, \\ -x_{ij} & \text{if } i > j \text{ and } \{i, j\} \in E, \\ 0 & \text{otherwise.} \end{cases}$$

(i) Find both the Tutte matrix A and its determinant $\det(A)$ when $G = K_3$ and $G = C_4$.

(ii) Show $\det(A) \not\equiv 0$ if G has a perfect matching.

We can think of a permutation $\pi \in S_n$ in terms of its cycle structure[1], which allows us to define $\operatorname{sgn}(\pi) = (-1)^{\#\text{ even cycles in }\pi}$. We then have $\det(A) = \sum_{\pi \in S_n} \operatorname{sgn}(\pi) \prod_{i \in [n]} a_{i\pi(i)}$.

(iii) If we think of isolated edges as cycles of length two, show that any nonzero monomial in this expansion of $\det(A)$ corresponds to a partition of $[n]$ into vertex-disjoint cycles in G.

(iv) By reversing the direction of an odd cycle, show that if $\det(A) \not\equiv 0$, then there is some partition of $[n]$ into vertex-disjoint cycles in G, all of which have even length.

(v) Deduce that $\det(A) \not\equiv 0$ if and only if G has a perfect matching, and give a randomised algorithm for testing for the existence of a perfect matching in G.

[1]For example, if $n = 6$ and $\pi(1) = 2$, $\pi(2) = 5$, $\pi(3) = 4$, $\pi(4) = 3$, $\pi(5) = 1$ and $\pi(6) = 6$, then π has cycle structure (1 2 5) (3 4) (6).