Graph Theory Applications | EPFL, Spring 2014

# Final Exam

Date: 27.06.2014

**Rules:**

- This exam is closed book. No electronic items are allowed. You are only allowed to have two handwritten single-sided A4 pages of notes. Place all your personal items on the floor. Leave only a pen and your ID on the desk. If you need extra scratch paper, please ask for it by raising your hand.
- Please do not cheat. We will be forced to report any such occurrence to the president of EPFL. This is not how you want to meet him. :-(
- The exam starts at 8:15 and lasts till 11:00 in INM10.
- If a question is not completely clear to you, don't waste time and ask us for clarification right away.
- It is not necessarily expected that you solve all problems. Don't get stuck. Start with the problem which seems the easiest to you and try to collect as many points as you can.

Name : ........................................................................

Sciper : ....................

**Problem 1** ... / 20

**Problem 2** ... / 20

**Problem 3** ... / 20

**Problem 4** ... / 20

**Problem 5** ... / 20

**Problem 6 – Bonus** ... / 10

**TOTAL** ... / 100+10

**Problem 1.** [*20pts*] Let $G = (V, E)$ be a graph with $m$ edges. Prove that there exists a subset of $E$, call it $F$, of cardinality at least $m/2$ so that the induced subgraph $G(F)$ is bipartite.

*Hint.* Probabilistic method.

*Solution.* Let $G = (V, E)$ and choose a random subset $T \subseteq V$ by inserting every vertex into $T$ independently with probability $1/2$. For a given edge $e = (u, v)$, let $X_e$ denote the indicator random variable of the event that exactly one of the vertices of $e$ is in $T$. Then, we have

$$\mathbb{E}(X_e) = \mathbb{P}(u \in T, v \notin T) + \mathbb{P}(u \notin T, v \in T) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}.$$

If $X$ denotes the number of edges having exactly one vertex in $T$, then

$$\mathbb{E}(X) = \sum_{e \in E} \mathbb{E}(X_e) = \frac{m}{2}.$$

Thus, for some $T \subseteq V$, there are at least $m/2$ edges crossing between $T$ and $V \setminus T$ forming a bipartite graph.

**Problem 2.** [*20pts*] Consider an $(l, r)$-regular bipartite graph $G$ with $2 \leq l \leq r$ and $n$ variable nodes representing an error correcting code. Assume that the graph is an expander with expansion exceeding $1/2$ for all sets of variable nodes of size up to $\alpha n$, $0 < \alpha \leq 1$. In class we showed that the corresponding code has a minimum distance of at least $\alpha n$, i.e., any non-zero codeword of this code has Hamming weight at least $\alpha n$.

Let us define a *stopping set*, call it $\mathcal{S}$, as a subset of the set of variables so that in the induced subgraph $G(\mathcal{S})$ there is no check node of degree 1. To compare, one can think of a codeword as a subset of the set of variable nodes, call it $\mathcal{S}$, so that in the induced subgraph $G(\mathcal{S})$ all check nodes have even degree.

Prove that in such a graph any non-zero stopping set must have size at least $\alpha n$, i.e., there are no small stopping sets.

*Solution.* The proof is essentially identical to the proof regarding the minimum distance and we proceed by contradiction. Assume therefore that there is a non-zero subset of the set of variable nodes which forms a stopping set, call it $\mathcal{S}$, and let it be of cardinality $S$. Let $\mathcal{C}$ be the set of neighbors of $\mathcal{S}$ and let it be of cardinality $C$. If $S \leq \alpha n$ then by the expansion property we must have $C > 1/2lS$. Further, since $\mathcal{S}$ is a stopping set, any element of $\mathcal{C}$ must be connected into $\mathcal{S}$ at least twice. Therefore, $lS \geq 2C$, contradicting the previous inequality.

**Problem 3.** [*20pts*] Suppose that several families want to go to a picnic by car sharing. Unfortunately, the members of each family constantly quarrel among themselves! To ensure a relaxing trip, the coordinator wants to make sure that no two members of the same family travel in a car together. Given a set of $n$ families with $F_1, F_2, \ldots, F_n$ members and a set of $m$ cars with capacity $C_1, C_2, \ldots, C_m$, write this problem as a graph problem and suggest an efficient algorithm to solve it.

*Solution.* Let us phrase the problem as a network flow problem. We have a source $s$, a sink $t$, and in addition a bipartite graph with $n$ nodes on one side (representing the families) and $m$ nodes on the other side (representing the cars). Connect the source to the node representing family $i$ with an edge of capacity $F_i$. Further, connect the node representing the car $j$ to the sink $t$ with an edge of capacity $C_j$. Finally, connect each node connecting a family to each node representing a car with an edge of capacity $1$. Clearly, there exists a car sharing arrangement if and only if there exists a feasible flow of value $\sum_{i=1}^{n} F_i$ between $s$ and $t$.

Run the Ford-Fulkerson algorithm to find the maximum flow of the network. If and only if such a flow has capacity $\sum_{i=1}^{n} F_i$, then a relaxing car is ensured.

**Problem 4.** [*20pts*] Take an $8 \times 8$ chessboard and remove from it the two squares with coordinates $(1, 1)$ and $(8, 8)$ (these are two squares opposite to each other along a diagonal).

Show that it is not possible to cover the remaining area consisting of 62 squares with dominoes of size $1 \times 2$ (the dominoes can be put horizontally or vertically).

*Hint.* Phrase this as a matching problem and show that no appropriate matching exists.

*Solution.* Note that if we remove the two squares with coordinates $(1, 1)$ and $(8, 8)$, then there are 30 squares of one color and 32 squares of the other color. Each domino piece covers exactly one white and one black square. Hence, it is not possible to cover the whole area with dominoes.

**Problem 5.** [*20pts*] Consider the complete graph on 6 vertices. Show that, for any coloring of the edges with blue and red, there exists always either a red or a blue triangle.

*Solution.* Pick a node $v$. Since the graph is complete $v$ has $5$ neighbors, and hence $5$ outgoing edges. Out of these $5$ edges, there are at least $3$ of them, which are painted in the same color. Without loss of generality that there are at least 3 red edges.

These 3 red edges connect $v$ to 3 nodes, call them $v_1$, $v_2$ and $v_3$. The nodes $v_1$, $v_2$ and $v_3$ are connected by a triple of edges, call them $e_1$, $e_2$ and $e_3$. Now, there are just two possibilities:

1. The edges $e_1$, $e_2$ and $e_3$ are all painted blue.

2. At least one of the edges $e_1$, $e_2$ and $e_3$ is painted red.

In the first case, there is a blue triangle. In the second case, there is a red triangle, since the edges connecting $v$ with $v_1$, $v_2$ and $v_3$ are red.

**Problem 6.** *[Bonus – 10points] Déjà vu.* Consider the graph $G(V, E)$ with adjacency matrix

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

(Note that this graph one vertex has a self loop.)

This matrix has a maximum eigenvalue of $\lambda = 1 + \sqrt{2}$ and an associated eigenvector $x^T = (1, 1/\sqrt{2}, 1/\sqrt{2})$.

Give a good upper and lower bound on the number of walks of length $n$ from vertex 1 back to vertex 1.

Note: *Good* here means that the ratio of the two bounds is $\Theta(1)$, i.e., it stays constant as $n$ grows large.

*Solution.*

Let $a_{i,j}^{(n)}$ be the number of walks of length $n$ from vertex $i$ to vertex $j$ for $i, j \in \{1, 2, 3\}$. Then, as we have seen in Problem Set 2, $a_{i,j}^{(n)}$ is the $(i, j)$-th entry of $A^n$. Our goal is to estimate $a_{1,1}^{(n)}$.

Now, by definition of eigenvalue and eigenvector,

$$A^n x = \lambda^n x.$$

Hence,

$$a_{1,1}^{(n)} + \frac{a_{1,2}^{(n)}}{\sqrt{2}} + \frac{a_{1,3}^{(n)}}{\sqrt{2}} = \lambda^n,$$

which, as the elements of $A^n$ are all positive, gives us the upper bound

$$a_{1,1}^{(n)} \leq \lambda^n.$$

Similarly we have that

$$a_{1,1}^{(n-1)} + \frac{a_{1,2}^{(n-1)}}{\sqrt{2}} + \frac{a_{1,3}^{(n-1)}}{\sqrt{2}} = \lambda^{n-1}.$$

This implies that

$$a_{1,1}^{(n-1)} + a_{1,2}^{(n-1)} + a_{1,3}^{(n-1)} \geq \lambda^{n-1}.$$

Observe that there is an edge from vertex 1 to any other vertex (included vertex 1 itself), which means that any walk of length $n$ from vertex 1 back to vertex 1 is the union of a walk of length $n - 1$ from vertex 1 to vertex $i$ ($i \in \{1, 2, 3\}$) and the edge from vertex $i$ to vertex 1. Therefore $a_{1,1}^{(n)} = a_{1,1}^{(n-1)} + a_{1,2}^{(n-1)} + a_{1,3}^{(n-1)}$, which gives the lower bound

$$a_{1,1}^{(n)} \geq \lambda^{n-1}.$$

The bounds are good since their ratio is $\lambda = 1 + \sqrt{2}$ and it stays constant as $n$ goes large.