Final Exam - Solution

Date: 27.06.2014

Problem 1. 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. 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. 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. 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. 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]* 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.