IE 512 Graphs, Networks, and Algorithms

Homework 3 Solution

Problem 3.3

(a) $(b_{|A|}, a_{|A|})$ is the last edge to be removed from the tree. It is immediate that $b_{|A|}$ is in A and $a_{|A|} = n$.

(b) At first step, we can recover b_1 to be the smallest label that is not in the non-leaf set $(c_1, \ldots, c_{|B|-1})$. If there is no such label, that is if $\{c_1, \ldots, c_{|B|-1}\} = A$, then we recover d_1 first by setting it to be the smallest label in B which is not in the non-leaf set of $(a_1, \ldots, a_{|A|-1})$. Once we have recovered at least one of b_1 or d_1, then we move to the next on in b and d and repeat the following process. If current step is for b_s or d_t, then set b_s to be the one with the smallest label that is not in $(c_t, \ldots, c_{|B|-1}) \cup (b_1, \ldots, b_{s-1})$. Similarly we can recover d_t.Continuing in this fashion we recover all b and d.

(c) The above procedure always add edges between A and B, hence bipartite assumption is always satisfied. Also, resulting graph is a tree and we can prove it the same way we did it in class for non-bipartite case.

(d) The total number of bipartite Prüfer codes is $|A|^{|B|-1}|B|^{|A|-1}$. We just showed that there is one-to-one correspondence between set the of bipartite trees and the bipartite Prüfer codes. It follows that the total number of bipartite trees is also $|A|^{|B|-1}|B|^{|A|-1}$.

Problem 3.4 A distance between two nodes is defined as the number of edges in the shortest path between those two vertices. A vertex in G is central if its greatest distance from any other vertex is as small as possible. This distance is called **radius**. The **diameter** of a graph is defined as the greatest distance between two nodes in the graph.

- We want to prove that for every graph G

$$radius(G) \leq diameter(G) \leq 2 \cdot radius(G)$$

Let $d(i, j)$ be the distance between two nodes i and j. Notice that

$$radius(G) = \min_{c \in V} \max_{k \in V} d(c, k) \ ,$$

and

$$diameter(G) = \max_{i \in V, j \in V} d(i, j) \ .$$

First, we show $radius(G) \leq diameter(G)$. From above formulation,

$$\begin{aligned} radius(G) &= \min_{c \in V} \max_{k \in V} d(c, k) \\ &\leq \max_{k \in V} d(c, k) \\ &\leq \max_{c \in V} \max_{k \in V} d(c, k) \\ &= diameter(G) \end{aligned}$$

The first inequality follows from the fact that taking any other c than the minimizer will only increase (or do not change) the distance, and the second inequality follows from the fact that taking the maximum c will again only increase (or do not change) the distance. This proves the first inequality.

Next, we show $diameter(G) \leq 2 \cdot radius(G)$. To do this, we need to use the triangular inequality:

$$d(i, j) \leq d(i, k) + d(k, j) \ ,$$

for any nodes $i, j, k \in V$. This follows from the definition of distance. The minimm number of hops between i and j will be at least $d(i, k) + d(k, j)$ since this value of distance can be achieved by taking the minimum distance path from node i and to k and then taking the minimum distance path from k to j. We have found one candidate path with distance $d(i, k) + d(k, j)$, so the minimum number of hops between i and j will only be equal or smaller. By this triangular inequality, we know that

$$\begin{aligned} diameter(G) &= \max_{i \in V} \max_{j \in V} d(i, j) \\ &\leq \max_{i \in V} \max_{j \in V} d(i, c) + d(c, j) \\ &\leq \max_{i \in V} d(i, c) + \max_{j \in V} d(c, j) \\ &= 2 \cdot radius(G) \end{aligned}$$

- We want to prove that a graph G of radius at most k and maximum degree at most d where d is an integer grater than two, has fewer than $\frac{d}{d-2}(d-1)^k$ vertices. We will construct a graph (which is a tree) with all nodes having degree d and radius exactly k that has $\frac{d(d-1)^k-2}{d-2}$ vertices. When $k = 0$, start from a root node. For $k = 1$, add d children to the root. Recursively grow your tree, and the number of nodes added at depth k is $d(d-1)^{k-1}$, except for when $k = 0$ there is only one node. summing up all the number of added nodes, we get that the total number of nodes at depth k is

$$\begin{aligned} 1 + \sum_{n=1}^{k} d(d-1)^{n-1} &= 1 + d\frac{(d-1)^k - 1}{d-2} \\ &= \frac{d(d-1)^k - 2}{d-2} \end{aligned}$$

which is always smaller than $\frac{d(d-1)^k}{d-2}$.

To complete the proof, we need to show that we cannot add any more nodes to this tree, without violating the assumptions. If we add any node to a non-leaf, then this will make that node degree $d+1$, thus violating the assumption that the maximum degree id d. If we add any node to only leaves, then this will increase the radius of the tree by at least one, thus violating the assumption that the radius id k. Hence, this is the graph with maximum nomber of nodes.

Problem 3.5 An amateur graph theorist, in his scribblings, might invent the following two definitions of k-edge connectivity. a directed graph G is k-edge connected if

(a) G remains connected after removing any $(k-1)$ edges.

OR

(b) There are at least k edge-disjoint paths between every pair of nodes in G.

Recall that a directed graph G is connected if and only if for each pair of nodes i and j there exists a directed path from node i to node j. And two paths are edge-disjoint if they do not share any edges.

It is clear that if G satisfy definition (b) then it also satisfies definition (a). consider a particular pair i and j. From (b) we know there are at least k edge-disjoint paths between i and j. When removing $k-1$ edges, the most harm will be done if we remove 1 edge per path for $k-1$ paths. Since paths are edge-disjoint, we cannot make two paths disconnected by removing a single edge. Hence, there will always be at least one path left that is not disconnected. This proves that (b) implies (a).

Now, we want to prove that if G remains connected after removing any $(k-1)$ edges then there are at least k edge-disjoint paths between every pair of nodes in G. Again consider a pair of nodes i and j. Construct a network N by making i the source and j the sink, and change every edge to be bi-directional edges that goes both ways. By (a), we know that this network has minimum cut value of at least k, because any cut $c(A, B)$ of the network N will have to have at least k edges in it. From max-flow min-cut theorem, we know that the value of the min-cut is equal to the value of the max-flow. Hence, $v(f^*) = \min_{A,B} c(A, B) \geq k$, and this implies that there are at least k flows from i to j. Since the network is integer values, each flow will have value at least one. Also, since the capacity on all the edges are one, no flow can have value larger than one. This means that there are k flows with values one each. And again since the capacity is at most one, no flows can share the same edge, and all the flows are edge-disjoint. This proves that there are at least k edge-disjoint paths between i and j.

Since the analysis did not depend on any properties of i and j, this holds for all pairs of nodes i and j.