

21-301 Assignment 04

Karan Sikka
ksikka@cmu.edu
October 21, 2013

1

Consider a d -regular bipartite graph. We will show that it has a perfect matching, and that if you remove it, you get a $d - 1$ -regular graph. Therefore, inductively, you can find d disjoint perfect matchings in a d -regular bipartite graph.

Perfect Matching

First, we seek to show a d -regular bipartite graph has a perfect matching. We claim it satisfies Hall's condition, that is that $|A| = |B|$ and $\forall S \subseteq A, |S| < |N(S)|$.

Since the graph is bipartite, all the edges connect vertices in A to vertices in B . Then $|E| = d|A|$. However, we can say the same about B . Then $|E| = d|B| = d|A|$ which implies that $|A| = |B|$.

Now we consider any subset S of A , and its neighborset, $N(S)$. There are $d|S|$ edges going out of S , connected to $N(S)$ vertices in B . We know every vertex in B has d incoming edges, so the size of $N(S)$ is $d|S|/d = |S|$. Therefore $|N(S)| = |S| \geq |S|$ and Hall's condition holds. Therefore a d -regular bipartite graph has a perfect matching.

Induction

Remove the perfect matching from the d -regular bipartite graph. The effect is that the number of outgoing edges on every vertex decreases by exactly one. The resulting graph is $d - 1$ regular. You can do this d times until you've removed all the edges from the graph resulting in a 0-regular graph. Therefore there are d perfect matchings, and since we can remove them serially from the d regular graph, they are disjoint.

2

The statement "there are disjoint subsets B_i of A_i with sizes of d_i " is also saying "in A_i , there exist d_i elements which can be labeled as B_i ". This is the same statement because giving an element one label implies that it only exists in the set for that label, and does not exist in the sets of the other labels, and therefore the labels' sets are disjoint given this.

Furthermore we know an element in A can be assigned the label B_i only if the element is a member of A_i .

In general, say the sum of all d_i is z . More formally, let $z = \sum_{i=1}^k d_i$. Then z elements will be assigned B_i labels. Assume these elements exist and are arbitrarily ordered. Then we denote Z_j as the j^{th} element $1 \leq j \leq z$. We can denote $B_{i,j}$ as the j^{th} element of B_i again assuming an arbitrary ordering ($1 \leq j \leq d_i$).

Then the statement "in A_i , there exist d_i elements which can be labeled as B_i " is the same as "there exists a perfect matching in the following bipartite graph".

We construct a bipartite graph so that the first set of vertices is Z_j and the second set is $B_{i,j}$, and there is an edge from Z_j to $B_{i,j}$ if $Z_j \in A_i$. To clarify, the edge signifies that it is possible to label Z_j as B_i .

Notice that the size of the first bipartition is z by definition of Z and the size of the second bipartition is $\sum |B_i| = \sum |d_i| = z$. The sizes of the bipartitions are equal and the first condition of Hall's condition is true.

Hawldap.

We denote Z_j as the j^{th} element of A assuming some arbitrary ordering. We denote $B_{i,j}$ as the j^{th} element of B_i again assuming an arbitrary ordering ($1 \leq j \leq d_i$).

We construct a bipartite graph so that the first set of vertices is $B_{i,j}$ and the second set is Z_j and there is an edge from $B_{i,j}$ to Z_j if $Z_j \in A_i$. As a result, the presence of an edge signifies that it is possible to put Z_j in B_i .

Since A and B are used in the problem statement, we will refer to these sets of vertices in the graph as X and Y . We notice that if we show that an A-perfect matching exists in this graph, we've shown that we can form disjoint B_i where $|B_i| = d_i$.

An A-perfect matching exists iff Hall's condition holds. That is, $\forall S \subseteq X, |N(S)| \geq |S|$.

Given any subset S of X , group the $B_{i,j}$ by same i , and call the set of i for nonempty B_i to be I . There are two cases. In the case that we get all the B_i s for a given i for all the i s in the subset, we can see that the size of the subset is the sum of the d_i s, which corresponds to given inequality in the problem statement.

Consider the $N(S)$. We can see that this consists of all the elements that could be in B_i for all i s being considered, which is precisely the size of the union of the A_i s. This corresponds to given inequality in the problem statement.

We've mapped the problem statement to the bipartite graph and proved the inequality via Hall's condition.

3

First note that a graph with $|E| = m$ and $|V| = n$ where $m = \binom{n}{2} = n * (n - 1) / 2$ is maximally connected, and therefore a connected graph with n vertices has at most m edges.

If you then add a vertex to a maximally connected graph, you get a disconnected graph with the largest possible number of edges: $m = \binom{n-1}{2} = (n - 1)(n - 2) / 2$. This is because if you add a single edge to the graph, it's connected again, since it must connect to the new vertex (all other possible edges already exist).

Therefore, any graph with $m > (n - 1)(n - 2) / 2$ must be connected.

4

Assume for sake of contradiction that there exists a tree T where two paths of max-length did not have any vertices in common.

We know that the tree is connected. Therefore there must exist a path from a vertex in one of the max-length paths to a vertex in the other. However, then we've found a longer path than either of the two that first assumed were max-length.

This is a contradiction and therefore there does not exist a tree satisfying the assumed condition and in any tree, two max-length paths must share at least one vertex.

5

Call a tree with exactly one vertex of degree i for any $2 \leq i \leq k$ for some $k > 0$ a k -weird tree. We denote the number of edges in a k -weird tree as S_k .

$S_1 = 1$. This is because if $k = 1$, then the tree has no vertices of degree 2 or greater, and the only such tree is the tree with two vertices and one edge.

$S_2 = 3$. You can see this from the tree with three vertices, where one is the root and two are leaves.

For $k > 1$, we can form a k -weird tree using the following inductive approach. Create a vertex with degree k by connecting it to $k - 1$ leaves and a root vertex of a $(k - 1)$ -weird subtree. This tree is indeed k -weird since it has exactly one vertex of degree k , and it has a $k - 1$ subtree which shows the weird property holds for all values $k - 1$ down till 2.

In this case $S_k = 1 + (k - 1) + S_{k-1}$. This is because there is one k -degree vertex, $k - 1$ of its leaves, and a $k - 1$ -weird subtree.

$$\begin{aligned} S_k &= 1 + (k - 1) + S_{k-1} \\ &= k + S_{k-1} \\ &= k + (k - 1) + \dots + S_3 \\ &= k + (k - 1) + \dots + 3 + S_2 \\ &= k + (k - 1) + \dots + 3 + 3 \\ &= \frac{(k-2)(k+3)}{2} + 3 \end{aligned}$$

We've determined the values of S_k for $k = 1$, $k = 2$, and arbitrary k .

6

For any vertex v of degree d in a tree where $d \geq 3$, there are d neighbors, by definition of degree. At most one of these neighbors is a parent vertex (not a leaf). The other $d - 1$ neighbors must either be leaves or the roots of disjoint subtrees. In the subtree case, you can apply this argument inductively, to show the number of leaves will always be greater than the number of vertexes of degree at least 3. (And as a formality, you see the base case holds with 0 vertexes and 1 vertex satisfying this condition.)

7

(this was shown in class)

The proof is by induction on the number of vertices.

In the base case $|V| = 1$ the claim obviously holds.

Consider a graph G with some $k \geq 1$ vertices which has no cycle and satisfies $|V| = |E| + 1$. We assume that this is a tree.

If you add a vertex to the graph, the graph is disconnected and is no longer a tree. We seek to add edges in order to make it a tree.

You must add an edge to make the graph connected. You can add an edge in such a way that it does not create a cycle, by simply extending off of a leaf. However, you cannot add more than that because it would have to connect to another node in G which was already maximal (given that it

was a tree), and it would create a cycle.

Therefore, if you add 1 vertex, you must add exactly 1 edge to create a tree from a tree.

$$|V'| = |V| + 1 = |E| + 1 + 1 = |E'| + 1$$

and more succinctly

$$|V'| = |E'| + 1$$