1 Spanning trees in connected graphs with per-vertex degree constraints

Assume that we wish to create a spanning tree for a graph. Correct algorithms for this problem proven by Kruskal and other mathematicians, however in this case we wish to consider an additional constraint. Assume that in the graph every vertex has some degree constraint d, where for a spanning tree to be correct every vertex in the spanning tree must have a degree exactly equal to that degree constraint d

Theorem 1. *Given a strongly connected graph $G(E, V)$ with total vertex degree of $2(v-1)$ algorithm 2 will return a correct spanning tree*

Proof. Base Case: Imagine an initial tree T, at the completion of algorithm step 3

Now, assume T has grown to size k vertices correctly, where T has a total degree constraint large enough to complete the spanning of the graph

T will now select the next untouched vertex of maximum degree. There are 3 cases for this vertex:

Case 1) The next vertex selected has a degree $= 1$
In this case, if the largest vertex remaining has a degree of 1 then all vertices not in T, U, must have a degree of 1. Thus T must have a total remaining vertex constraint of $|V \setminus U|$, which is equal to $|U|$. Therefore as there are $|U|$ vertices not in T and T has a total degree constraint of $|U|$ we know the degree constraint will not be violated, as each connection from T to U lowers the total degree constraint by one.

Case 2) The next vertex selected has a degree ≥ 1
Adding this vertex to T will increase T's total degree constraint by some value greater than or equal to 2, and then decrease T's total degree constraint by 2. Therefore the change in T cannot be less than zero, and thus T will not be decreased and therefore T will be able to complete the spanning tree.

Case 3) No new vertex can be selected
If no new vertex can be selected this means every vertex in G in in T

Therefore it is absurd that T will ever have a total degree of zero when there are vertices in G not in T

□

Theorem 2. *Given a strongly connected graph $G(E, V)$ with total vertex degree of $2(n-1)$ if algorithm 2 returns a spanning tree it will be correct*

Proof. Assume our algorithm returns an incorrect spanning tree for G. Our algorithm must necessarily cover every edge, as all the edges will have connections equal to their degree, and no degree can be zero. Additionally our spanning tree will feature $n-1$ edges, where n is the number of vertices, due to the total vertex degree constraint of $2(n-1)$. Therefore, as every vertex is touched, and the total number of edges is equal to $n-1$, the returned tree must be a spanning tree. $\square$