MST Construction Theorem

The following theorem shows how one can extend any spanning forest of a graph G that is contained in some MST of G, into a *larger* forest that is also contained in some MST of G. Many algorithms use this theorem to construct an MST efficiently: Roughly speaking, they start from the trivial forest of G with no edges (i.e., this initial forest of G consists of n trivial trees, each tree with just one node of G), and then they "apply" this theorem $n-1$ times to grow this initial forest of G into a MST of G. Each application of the theorem extends the spanning forest by one edge, and the resulting MST has $n-1$ edges.

Theorem:

Let $G = (V, E)$ be a connected undirected graph.

Let $T_1 = (V_1, E_1), T_2 = (V_2, E_2), \ldots, T_k = (V_k, E_k)$ be a spanning forest of G.

Suppose some MST T of G contains this spanning forest, i.e., T contains all the edges in $E_1 \cup E_2 \cup \ldots \cup E_k$. (u, v) is the light edge that crosses cut (V_i, V - V_i)

Let (u, v) be an edge of minimum-weight among all edges such that $u \in V_i$ and $v \in V - V_i$ (i.e., (u, v) is an edge with minimum weight among all the edges "out of T_i").

Then there is a MST T^* of G that contains all the edges in $E_1 \cup E_2 \cup \ldots \cup E_k \cup \{(u, v)\}$.

Proof (sketch):

By assumption, there is an $\boxed{\text{MST } T = (V, E')}$ that contains all the edges in $E_1 \cup E_2 \cup \ldots \cup E_k$. There are 2 possible cases:

(a) T also contains edge (u, v). In this case, $T^* = T$ and we are done.

(b) T does not contain (u, v). In this case, add edge (u, v) to T. We get graph $T' = (V, E' \cup \{(u, v)\})$. By Fact 2 (seen in class), T' has a unique cycle, and this cycle includes edge (u, v). Since $u \in V_i$ and $v \in V - V_i$, this cycle must also contain another edge (u', v') such that $u' \in V_i$ and $v' \in V - V_i$ (intuitively, this is because a cycle that has an edge "out of T_i" must also have an edge "into T_i"). Note that by the definition of (u, v), we have $w(u, v) \leq w(u', v')$.

Now remove (u', v') from T'. By Fact 2, this cuts the unique cycle of T', and we get back a spanning tree of G, denoted T^*. By construction, $T^* = (V, E' \cup \{(u, v)\} - \{(u', v')\})$.

Note that:

1. T^* contains all the edges in $E_1 \cup E_2 \cup \ldots \cup E_k$. This is because T, and therefore T', contain all these edges, and the only edge that we removed from T' to obtain T^*, namely, (u', v'), is not in $E_1 \cup E_2 \cup \ldots \cup E_k$.

2. T^* contains (u, v).

3. The weight of spanning tree T^* is $w(T^*) = w(T) - w(u', v') + w(u, v)$. Since $w(u, v) \leq w(u', v')$, we have $w(T^*) \leq w(T)$. In other words, the weight of T^* is less or equal to the weight of T. Since T is a *minimum* spanning tree of G, we conclude that T^* is also a *minimum* spanning tree of G.

By (3) above, T^* is an MST of G. Furthermore, by (1) and (2), T^* contains all the edges in $E_1 \cup E_2 \cup \ldots \cup E_k \cup \{(u, v)\}$.

Q.E.D.