Sketch of the solutions

December 16, 2017

Question 1

Recall that in an unweighted bipartite graph, after finding a maximum cardinality matching, we can decompose the graph into three parts: (1) $G_1 = (A_1 \cup B_1, E_1)$, consisting of vertices reachable via an alternative path in $E_1$ from a unmatched node in $A_1$; (2) $G_2 = (A_2 \cup B_2, E_2)$, consisting of vertices reachable via an alternative path in $E_2$ from a unmatched node in $B_2$; (3) $G_3 = (A_3 \cup B_3, E_3)$, consisting of a perfect matching among $A_3$ and $B_3$. Note that there is no edge in $A_1 \times (B_2 \cup B_3)$. These characterisations will be used for the following weighted case.

Let us initialise $\pi(a) = \max_{e \in E} w(e), \forall a \in A$ and $\pi(b) = 0, \forall b \in B$. Define $G_\pi$ as the subgraph of tight edges, i.e. $e = (a,b)$ is part of $G_\pi$ if only if $\pi(a) + \pi(b) = w(e)$. Let $M = \emptyset$.

In each phase, we augment $M$ in $G_\pi$ is possible; otherwise, we update $\pi$ to “reveal” more edges. Precisely, suppose that $M$ is a maximum cardinality matching in $G_\pi$, define $G_1$ and $G_2$ and $G_3$ as above. Now let

$$\Delta = \min\{\min\{\pi(a) + \pi(b) - w(a,b)|a \in A_1, b \in B_2 \cup B_3\},\min\{\pi(a)|a \in A_1\}\}$$

Then for each $a \in A_1$, let $\pi(a) = \pi(a) - \Delta$ and for each $b \in B_1$, let $\pi(b) = \pi(b) + \Delta$.

If, after the update, some node $a \in A_1$ has $\pi(a) = 0$ we stop the algorithm.

It can be seen that the following invariants are maintained throughout the algorithm.

1. $\pi$ is feasible, i.e., $\pi(v) \geq 0$ for all $v \in A \cup B$; furthermore, $\pi(a) + \pi(b) \geq w(a,b)$, $\forall (a,b) \in E$.

2. $M$ uses only tight edges.

3. If a node $u \in A \cup B$ has $\pi(u) > 0$, $u$ is matched by $M$.

Thus, when the algorithm terminates, the above three invariants prove Egerváry’s theorem and the complexity is easily seen to be $O(n_1^2 m)$.
Question 2

Define \( V^+ = \{ v | 0.5 < x_v^* < 1 \} \) and \( V^- = \{ v | 0 < x_v^* < 0.5 \} \). Define

\[
y_v^\epsilon = \begin{cases} 
x_v^* + \epsilon & v \in V^+ \\
x_v^* - \epsilon & v \in V^- \\
x_v^* & v \notin V^+ \cup V^-
\end{cases}
\]

\[
z_v^\epsilon = \begin{cases} 
x_v^* - \epsilon & v \in V^+ \\
x_v^* + \epsilon & v \in V^- \\
x_v^* & v \notin V^+ \cup V^-
\end{cases}
\]

By choosing \( \epsilon \) sufficiently small, we can guarantee that \( y_v^\epsilon \) and \( z_v^\epsilon \) still satisfy the linear constraints (by case analysis). Now as \( x^* = (y_v^\epsilon + z_v^\epsilon)/2 \), we reach a contradiction to the assumption that \( x^* \) is an extreme point.

For the second part, consider the following hypergraph. \( V = \{ v_1, \ldots, v_6 \} \); there are three hyperedges: \( (v_1, v_2, v_3) \), \( (v_3, v_4, v_5) \), \( (v_5, v_6, v_1) \). Now put the value of 0.5 in \( v_1, v_3 \) and \( v_5 \) and the value of 0 on the other three vertices. It can be seen that this is an extremely point.

Question 3

Recall that when Blossom algorithm terminates, we can separate the vertices into three groups, \( A(T) \), \( B(T) \) and \( U(T) \): \( A(T) \) are those who have an odd-length alternative path to an unmatched vertex/blossom; \( B(T) \) are those vertices who belong to a blossom or who have an even-length path to a unmatched vertex/blossom; \( U(T) \) induces a perfect matching among themselves. Note that there is no edge in \( B(T) \times (B(T) \cup U(T)) \), except those within the same blossom.

The critical observation is that there must be at least \( k \) vertices in \( A(T) \) (implying that the matching is of size at least \( k \)). To see this, notice that if we remove all vertices from \( A(T) \), the graph becomes disconnected—this causes a contradiction to the assumption of \( k \)-connectivity if there are less than \( k \) vertices in \( A(T) \).

For vertex cover, let us open all blossoms in \( B(T) \) and call the resulting max-cardinality matching \( M \). We choose all nodes in \( U(T) \), all nodes in \( A(T) \), all nodes in \( B(T) \), except that when a node in \( B(T) \) is matched to a node in \( A(T) \) in \( M \). A simple counting shows that this vertex cover has size at most \( 2|M| - k \).

Question 4

Look at the Gomory-Hu tree \( T \). Consider edge \((u, v) \in T \) where \( u \) is a leaf in \( T \). Thus, \( \{u\} \) is a min-cut separating \( u \) and \( v \), whose size is exactly \( \delta(\{u\}) \), which is at least \( k \). Now by max-flow-min-cut theorem, there would be at least \( k \) edge-disjoint paths connecting \( u \) and \( v \) in the original graph.

Finally, notice that since \( |V| \geq 3 \), there are at least two leaves in \( T \).
Question 5

The symmetric differences of \( V_{f_1}, V_{f_2}, \cdots, V_{f_t} \) is equivalent to either \( U \) or \( V \setminus U \) can be easily shown by induction.

Observe that \( V_1 \bigoplus V_2 \) is of even size if both \( V_1 \) and \( V_2 \) are of even size. As \( \bigoplus_{i=1}^t V_{f_i} \) is of odd size, at least one of them, say \( V_{f_k} \), is of odd size. Assume that \( f_k = (u,v) \) and \( u \in U \) (the odd set with the min-cut size). Then the capacity of edge \( f_k \) in the Gomory-Hu tree \( H \) is upper-bounded by the cut size of \( U \), since the latter also induces a \((u,v)\)-cut.

So the algorithm is simple. Check all edges \((a,b) \in H\), whose capacity is strictly less than 1, and verify whether removing \((a,b)\) in \( H \) separates the graph into two odd sets.