Notes by Samir Khuller.

1 The Primal-Dual Method applied to Max weight perfect matching in bipartite graphs

We first formulate the Max Weight Bipartite matching problem as an Integer Program (IP). Since Integer Programming is NP-hard, we cannot use this approach to actually solve the problem. Instead what we will show is that we can still consider the relaxation of the IP to the natural Linear Program (LP). While it is true that an extreme point solution to the LP, is actually an inegral solution, we will not make use of this property. (So for example, the simplex method can be used to derive a solution for max weight perfect matching. However, this is not a worst case polynomial time algorithm, however efficient it may be in practice.)

Instead, we will apply the recently learnt primal dual method for solving linear programs to the LP formulation for max weight bipartite matching. This approach actually yields a polynomial time algorithm. In fact, we will derive an algorithm that is essentially the same algorithm that we studied earlier for max weight perfect matchings. (There is a small difference between the two algorithms, but this is minor.) We will show the development of the algorithm, explaining why it is natural to think of the *label* function on vertices, the notion of the equality subgraph, as well as the idea of trying to find a max matching in the equality subgraph.

Let $x(u,v)$ be a $\{0,1\}$ variable that denotes whether or not an edge is in the matching. Let the graph be formed by two sets of vertices X and Y.

$$\max \sum_{e=(u,v)} w(u,v)x(u,v)$$

$$\sum_{u \in X} x(u,v) = 1 \text{ for } v \in Y$$

$$\sum_{v \in Y} x(u,v) = 1 \text{ for } u \in X$$

$$x(u,v) \in \{0,1\}$$

We relax the integrality requirements to $x(u,v) \geq 0$. The Dual of this LP can be written as follows.

$$\min_{v \in X \cup Y} \pi_v$$

$$\pi_u + \pi_v \geq w(u,v) \text{ for each edge } (u,v)$$

Notice that π_u is exactly the label of node u. The dual constraint is exactly the requirement that a legal label function had to satisfy. In fact, we explicitly proved that any label function was an upper bound on the weight of the max matching. This is a trivial consequence of the fact that the label is the dual of the LP formulation of matching! (Notice that any feasible solution to the dual is an upper bound on the LP, and hence the IP for matching. In fact, the LP has a larger feasible solution space and contains all the integer solutions.)

Given a label function (or π values) for all the nodes that satisfy the dual constraints on each edge, we now define J_π as the set of edges for which the dual constraints are met with equality. This forms the equality subgraph. By complementary slackness, we force all $x(u,v) = 0$ if the edge (u,v) is not in the equality subgraph. We write the simplified problem P' as follows:

$$\max \sum_{e=(u,v)} w(u,v)x(u,v)$$

$$\sum_{u \in X \,|\, (u,v) \in J_\pi} x(u,v) = 1 \text{ for } v \in Y$$

$$\sum_{v \in Y \,|\, (u,v) \in J_\pi} x(u,v) = 1 \text{ for } u \in X$$

$$x(u,v) = 0 \text{ if } j \notin J_\pi$$

$$x(u,v) \geq 0 \text{ if } j \in J_\pi$$

This is the same problem as trying to find a max weight perfect matching in the equality subgraph. In fact, any feasible solution (a perfect matching in the equality subgraph) is actually optimal.

We now write the restricted primal (RP) as follows.

$$\min_{v \in X \cup Y} x_v^a$$

$$\sum_{u \in X \,|\, (u,v) \in J_\pi} x(u,v) + x_v^a = 1 \text{ for } v \in Y$$

$$\sum_{v \in Y \,|\, (u,v) \in J_\pi} x(u,v) + x_u^a = 1 \text{ for } u \in X$$

$$x(u,v) = 0 \text{ if } j \notin J_\pi$$

$$x(u,v) \geq 0 \text{ if } j \in J_\pi$$

$$x_v^a \geq 0$$

The claim (easy to prove) is that RP has a 0 cost optimal solution if and only if P' has a feasible solution. It is easy to find a feasible solution for RP. For example set all edge variables to 0, and set all $x_v^a = 1$. An optimal solution for RP has value exactly the number of free nodes in a maximum matching in the equality subgraph (it is easy to see that this is feasible, since we can set $x_v^a = 1$ if v is free, later we will prove optimality of this solution). If the maximum matching is perfect, we are done since this is an optimal solution. If the maximum matching is not perfect then we have to revise the dual solution π to obtain a better dual solution and re-define the equality subgraph. We will now (as before) use the Dual of RP to update the π values (or label function).

The Dual of RP can be described as follows.

$$\max_{v \in X \cup Y} \ell_v$$

$$\ell_v \leq 1 \text{ for all } v \in X \cup Y$$

$$\ell_u + \ell_v \leq 0 \text{ for } (u,v) \in J_\pi$$

An optimal solution for DRP can be obtained as follows: Let $S =$ the set of free nodes in X. Grow hungarian trees from each node in S in the equality subgraph. Let $T =$ all nodes in Y encountered in the search for an augmenting path from nodes in S. Add all nodes from X that are encountered in the search to S.

$$\overline{S} = X - S.$$

$$\overline{T} = Y - T.$$

$$|S| > |T|.$$

There are no edges from S to $\overline{T}$, since this would imply that we did not grow the hungarian trees completely. The optimal dual solution is as follows. If $v \in S \cup \overline{T}$ we have $\ell_v = 1$, else $\ell_v = -1$. Notice that this is a dual feasible solution, and in fact optimal as the value of the dual is exactly the number of free nodes in the graph. (If we have matching costs solution for RP and DRP we must have optimal solutions for both.) We now revise π_v to be $\pi_v - \theta l_v$. Hence the labels in $S \cup \overline{T}$ all decrease by θ and remaining labels all increase by θ. The amount by which we change the labels is determined by the edges not in J_π. We choose

$$\theta = \min_{(u,v) \notin J_\pi and \ell_u + \ell_v > 0} \frac{\pi_u + \pi_v - w(u,v)}{2}.$$

Notice that at least one new edge will enter the equality subgraph in this way. If we are able to increase the size of the matching, the phase ends. Otherwise we repeat this. The proof that we did earlier proves that after at most n iterations, the cost of the RP decreases by 1. Its worth noting that this is not exactly the modification to the label function that we did in the earlier algorithm. There the labels of S and T were modified, but not the labels of the remaining nodes. This corresponds to a choice of picking a non-optimal solution for DRP and using that to modify the dual solution. Notice that we never really used the fact that we modified the dual solution by using an *optimal* solution to DRP, hence the scheme still works with this choice of a solution for DRP. The solution for the DRP is as follows: for $v \in S$ we have $\ell_v = 1$, for $v \in T$ we have $\ell_v = -1$, else $\ell_v = 0$.