Advanced Algorithms September 30, 2024

Lecture 6: Hungarian Method and Weighted Vertex Cover

Notes by Ola Svensson[1]

In this lecture we do the following:

- We quickly recall LP duality and complementarity slackness.
- We write down the LP of the min-cost perfect matching problem, take its dual, and consider the complementarity slackness conditions.
- We use these insights to devise the Hungarian algorithm for solving the min-cost perfect matching problem.
- We then discuss how to devise algorithms when relaxing integrality constraints is not without loss of generality. We do so by considering a simple example: the weighted vertex cover problem.

1 Recall: Linear Programming Duality

In the last lecture, we saw that if we have a linear progam with n variables $x_1, x_2, ..., x_n$ and m constraints of the following form:

$$\begin{aligned} \textbf{Minimize } & \sum_{i=1}^{n} c_i x_i \\ \textbf{Subject to: } & \sum_{i=1}^{n} A_{ji} x_i \geq b_j \quad \forall j = 1, \ldots, m, \\ & x \geq 0. \end{aligned}$$

Then, the dual program with m variables $y_1, y_2, ..., y_m$ and n constraints is as follows:

$$\begin{aligned} \textbf{Maximize } & \sum_{j=1}^{m} b_j y_j \\ \textbf{Subject to: } & \sum_{j=1}^{m} A_{ji} y_j \leq c_i \quad \forall i = 1, \ldots, n, \\ & y \geq 0. \end{aligned}$$

In other words, if the primal program is written as $\min\{c^\top x : Ax \geq b, x \geq 0\}$, then the dual can be written as $\max\{b^\top y : A^\top y \leq c, y \geq 0\}$. Here $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$.
Remark Any linear program can be written in the above form and thus we can use the above recipe to take the dual of *any* linear program.

Basically by definition, we have weak duality:

Theorem 1 (Weak Duality) *If x is* primal-feasible *(meaning that x is a feasible solution to the primal problem) and y is* dual-feasible*, then*

$$\sum_{i=1}^{n} c_i x_i \geq \sum_{j=1}^{m} b_j y_j.$$

[1]**Disclaimer:** These notes were written as notes for the lecturer. They have not been peer-reviewed and may contain inconsistent notation, typos, and omit citations of relevant works.

Perhaps more surprisingly, it turns out that the optimal solutions to the primal and dual have the same value:

Theorem 2 (Strong Duality) *If x is an optimal primal solution and y is an optimal dual solution, then*

$$\sum_{i=1}^{n} c_i x_i = \sum_{j=1}^{m} b_j y_j.$$

Furthermore, if the primal problem is unbounded, then the dual problem is infeasible and analogously if the dual is unbounded, then the primal is infeasible.

Strong duality gives an important relationship between primal and dual optimal solutions.

Theorem 3 (complementarity slackness) *Let $x \in \mathbb{R}^n$ be a feasible solution to the primal and let $y \in \mathbb{R}^m$ be a feasible solution to the dual. Then*

$$x, y \text{ are both optimal solutions } \iff \begin{cases} x_i > 0 \Rightarrow c_i = \sum_{j=1}^{m} A_{ji} y_j & \forall i = 1, \ldots, n, \\ y_j > 0 \Rightarrow b_j = \sum_{i=1}^{n} A_{ji} x_i & \forall j = 1, \ldots, m. \end{cases}$$

Proof We will apply the strong duality theorem to the weak duality theorem proof.

- $\Rightarrow$ Let x be the optimal primal solution. From the weak duality theorem proof, we have that

$$\sum_{j=1}^{m} b_j y_j \leq \sum_{j=1}^{m} \sum_{i=1}^{n} A_{ji} x_i y_j = \sum_{i=1}^{n} \left(\sum_{j=1}^{m} A_{ji} y_j \right) x_i \leq \sum_{i=1}^{n} c_i x_i. \tag{1}$$

Here we used the fact that $x, y \geq 0$. On the other hand by the strong duality theorem

$$\sum_{j=1}^{m} b_j y_j = \sum_{i=1}^{n} c_i x_i.$$

So in (1) there are equalities everywhere. Thus

$$\sum_{i=1}^{n} c_i x_i = \sum_{i=1}^{n} \left(\sum_{j=1}^{m} A_{ji} y_j \right) x_i \Rightarrow c_i x_i = \left(\sum_{j=1}^{m} A_{ji} y_j \right) x_i \text{ for } i = 1, \ldots n.$$

And finally for every $x_i,\ i = 1, \ldots n$:

$$x_i \neq 0 \quad c_i x_i = \left(\sum_{j=1}^{m} A_{ji} y_j \right) x_i \Rightarrow c_i = \left(\sum_{j=1}^{m} A_{ji} y_j \right).$$

- $\Leftarrow$ Similarly to the previous part we know that:

$$x_i c_i = \left(\sum_{j=1}^{m} A_{ji} y_j \right) x_i \qquad \forall i = 1, \ldots, n,$$
$$y_j b_j = \left(\sum_{i=1}^{n} A_{ji} x_i \right) y_j \qquad \forall j = 1, \ldots, m.$$

Thus

$$\sum_{j=1}^{m} b_j y_j = \sum_{j=1}^{m}\sum_{i=1}^{n} A_{ji} x_i y_j = \sum_{i=1}^{n}\left(\sum_{j=1}^{m} A_{ji} y_j\right) x_i = \sum_{i=1}^{n} c_i x_i.$$

The above equality is equivalent to x, y being optimal solutions to the primal and the dual linear programs, respectively. Indeed for feasible solution $x^\star$ to the primal we have by weak duality

$$\sum_{i=1}^{n} c_i x_i^\star \geq \sum_{j=1}^{m} b_j y_j = \sum_{i=1}^{n} c_i x_i.$$

Thus x is an optimal solution to the primal program and similarly y is an optimal solution to the dual.

■

2 Duality and Complementarity Slackness of Min-Cost Perfect Matching

Let $G = (A \cup B, E)$ be a bipartite weighted graph with edge-costs $c : E \to \mathbb{R}$. We wish to find a perfect matching M of minimum cost $\sum_{e \in M} c(e)$.

To write down the linear program for this problem, we have a variable x_e for every edge e, with the intended meaning that x_e takes value 1 if e is in the matching and 0 otherwise. In a *perfect* matching, every vertex is adjacent to exactly one edge of the matching. This leads to the following linear program:

$$\begin{aligned}
\textbf{Minimize} \quad & \sum_{e \in E} c(e) x_e \\
\textbf{Subject to:} \quad & \sum_{b \in B:(a,b)\in E} x_{ab} = 1 && \forall a \in A, \\
& \sum_{a \in A:(a,b)\in E} x_{ab} = 1 && \forall b \in B, \\
& x_e \geq 0 && \forall e \in E.
\end{aligned}$$

In a previous lecture, we saw that any extreme point of the above linear program is integral. Hence, we could solve the min-cost perfect matching problem by simply solving the above linear program, finding an extreme-point solution. However, this may be unnecessarily inefficient. Instead, we will see how to use LP duality to, basically, reduce this (weighted) problem to that of finding a perfect matching in an *unweighted* graph. (A problem that we already saw how to solve using augmenting paths.)

Obtaining the dual. To take the dual of the above program, let us first write it in the same form as the linear program in Section 1. We can do this by replacing each equality by two inequalities. This gives us the following equivalent linear program:

$$\begin{aligned}
\textbf{Minimize } & \sum_{e\in E} c(e)x_e \\
\textbf{Subject to: } & \sum_{b\in B:(a,b)\in E} x_{ab} \geq 1 && \forall a\in A, \\
& -\sum_{b\in B:(a,b)\in E} x_{ab} \geq -1 && \forall a\in A, \\
& \sum_{a\in A:(a,b)\in E} x_{ab} \geq 1 && \forall b\in B, \\
& -\sum_{a\in A:(a,b)\in E} x_{ab} \geq -1 && \forall b\in B, \\
& x_e \geq 0 && \forall e\in E.
\end{aligned}$$

For each $a \in A$, we then associate a variable u_a^+ to the first constraint for a ($\sum_{b\in B:(a,b)\in E} x_{ab} \geq 1$) and u_a^- to the second constraint for a ($-\sum_{b\in B:(a,b)\in E} x_{ab} \geq -1$). Similarly, we have variables v_b^+ and v_b^- for each $b \in B$. These variables take the same role as the y-variables in Section 1: the dual has a variable for each constraint in the primal. We now have that the dual equals

$$\begin{aligned}
\textbf{Maximize } & \sum_{a\in A}\left(u_a^+ - u_a^-\right) + \sum_{b\in B}\left(v_b^+ - v_b^-\right) \\
\textbf{Subject to: } & \left(u_a^+ - u_a^-\right) + \left(v_b^+ - v_b^-\right) \leq c(e) && \forall e=(a,b)\in E \\
& u_a^+, u_a^-, v_b^+, v_b^- \geq 0 && \forall a\in A, b\in B\,.
\end{aligned}$$

Complementarity Slackness. Complementarity slackness tells us that if $x, (u^+, u^-, v^+, v^-)$ are feasible, then they are both optimal if and only if the following holds:

$$\begin{aligned}
x_e > 0 &\Rightarrow \left(u_a^+ - u_a^-\right) + \left(v_b^+ - v_b^-\right) = c(e) && \forall e=(a,b)\in E, \\
u_a^+ > 0 &\Rightarrow \sum_{b\in B:(a,b)\in E} x_{ab} = 1 && \forall a\in A, \\
u_a^- > 0 &\Rightarrow -\sum_{b\in B:(a,b)\in E} x_{ab} = -1 && \forall a\in A, \\
v_b^+ > 0 &\Rightarrow \sum_{a\in A:(a,b)\in E} x_{ab} = 1 && \forall b\in B, \\
v_b^- > 0 &\Rightarrow -\sum_{a\in A:(a,b)\in E} x_{ab} = 1 && \forall b\in B\,.
\end{aligned}$$

We have thus obtained the dual and the complementarity slackness conditions in an "automatic" way.

Simplifying the notation. Before continuing, we would like to make an observation that will simplify our description and notation. Namely, in the dual that we obtained, the variables u_a^+ and u_a^- always appear only as part of the expression $(u_a^+ - u_a^-)$. Although we have $u_a^+, u_a^- \geq 0$, this expression can take any value (positive and negative). We can thus for every $a \in A$ replace $(u_a^+ - u_a^-)$ by a new variable $u_a \in \mathbb{R}$ whose value is not constrained to be nonnegative (it can take both positive and negative values). We do the same for each $b \in B$ and replace $\left(v_b^+ - v_b^-\right)$ by v_b.[2] We can thus write the primal and the dual as follows:

[2] Note that our new LP is in fact equivalent to the old one. Given a feasible solution to the old LP, we can obtain a feasible solution to the new LP by setting the values for the new variables as $u_a := u_a^+ - u_a^-$ (and similarly for v). Since we

$$\begin{aligned} \textbf{Minimize} \quad & \sum_{e \in E} c(e) x_e \\ \textbf{Subject to:} \quad & \sum_{b \in B : (a,b) \in E} x_{ab} = 1 \quad \forall a \in A, \\ & \sum_{a \in A : (a,b) \in E} x_{ab} = 1 \quad \forall b \in B, \\ & x_e \geq 0 \qquad\qquad \forall e \in E. \end{aligned}$$

and

$$\begin{aligned} \textbf{Maximize} \quad & \sum_{a \in A} u_a + \sum_{b \in B} v_b \\ \textbf{Subject to:} \quad & u_a + v_b \leq c(e) \quad \forall e = (a,b) \in E\,. \end{aligned}$$

With this simplified notation, complementarity slackness gives us that if $x, (u,v)$ are feasible, then they are both optimal if and only if the following holds:

$$\begin{aligned} x_e > 0 \Rightarrow u_a + v_b = c(e) & \qquad \forall e = (a,b) \in E, \\ u_a \neq 0 \Rightarrow \sum_{b \in B : (a,b) \in E} x_{ab} = 1 & \qquad \forall a \in A, \\ v_b \neq 0 \Rightarrow \sum_{a \in A : (a,b) \in E} x_{ab} = 1 & \qquad \forall b \in B. \end{aligned}$$

As a final simplification, observe that the last two conditions always hold since they follow immediately from the fact that x is primal-feasible (this is because in this LP we have equalities instead of inequalities).

We can thus summarize the above with the following:

Lemma 4 *A perfect matching M is of minimum cost iff there is a feasible dual solution u, v such that*

$$u_a + v_b = c(e) \qquad \textit{for every } e = (a,b) \in M.$$

We will now use this fact to develop an algorithm for finding a minimum-cost perfect matching.

3 The Hungarian Algorithm

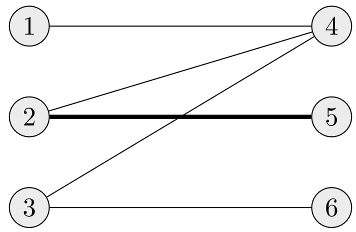
We start with some intuition before we give the formal description of the algorithm.

do the same in the objective function, the objective value remains unchanged. We can also go back and obtain a solution of the old LP from a solution of the new LP as follows: if $u_a \geq 0$, set $u_a^+ = u_a$ and $u_a^- = 0$; otherwise set $u_a^+ = 0$ and $u_a^- = -u_a$ (and similarly for v).

Also note that the new LP is what we would have obtained if we had multiplied each equality constraint $\sum_{b \in B:(u,v) \in E} x_{ab} = 1$ by an unconstrained dual variable $u_a \in \mathbb{R}$, rather than rewriting this equality constraint as two inequality constraints and then multiplying each by a nonnegative dual variable (u_a^+ or u_a^-). (And similarly for v.)

3.1 Intuition

Consider the following instance of the min-cost perfect matching problem:

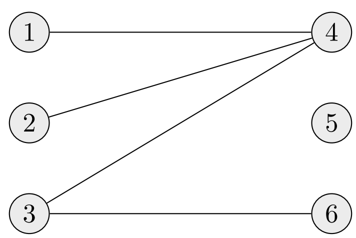


The thin edges have cost 1, whereas the thick edge has cost 2. The Hungarian algorithm will use Lemma 4 in the following way: we will maintain a dual solution u, v that is *feasible at all times*. Then, for a fixed dual solution, the lemma tells us that our perfect matching is only allowed to contain edges that are *tight*, i.e., edges $e = (a, b)$ for which $u_a + v_b = c(e)$. This reduces our problem to finding any perfect matching in the subgraph consisting only of tight edges, i.e., in the graph (V, E') where $E' = \{e = (a, b) \in E : u_a + v_b = c(e)\}$. *Intuitively, we have thus reduced our weighted problem to an unweighted one!*

Let us return to our example. We initialize our procedure with the trivial dual solution

$$v_b = 0, \quad u_a = \min_{b \in B} c_{ab}.$$

(We could have also started from $u = v = 0$.) So in the example, $v_4 = v_5 = v_6 = 0$ and $u_1 = u_2 = u_3 = 1$. The set E' of tight edges is thus:



We then try to find a perfect matching in this graph using e.g. the augmenting path algorithm we saw in Lecture 3. However, the considered graph has *no* perfect matching! This is because we started with a poor lower bound (dual solution). So we will use the fact that there is no perfect matching to improve our dual solution. We make use of *Hall's condition*, which you will prove in the exercise session:

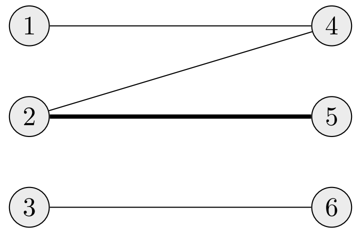
Theorem 5 (Hall's Theorem) *An n-by-n bipartite graph $G = (A \cup B, E')$ has a perfect matching if and only if $|S| \leq |N(S)|$ for all $S \subseteq A$.*

Here $N(S) = \{b \in B :$ there is an $a \in S$ such that $(a, b) \in E'\}$ denotes the neighborhood of the vertices in S.

In the above example, we have $S = \{1, 2\}$ and $N(S) = \{4\}$, which violates Hall's condition (and thus there is no perfect matching). We now use the set S to improve our dual lower bound.

We gradually increase u_a for every $a \in S$ and at the same time decrease v_b for $b \in N(S)$ at the same rate. Let us see what happens to all edges in E'. Notice that the tight edges between S and $N(S)$ will remain tight. Similarly, the tight edges between $A \setminus S$ and $B \setminus N(S)$ will remain tight. Any tight edges between $A \setminus S$ and $N(S)$ will stop being tight. Finally, by definition, there are no edges from S to $B \setminus N(S)$ initially. We continue to gradually change the dual solution until some such edge becomes tight.

In the above example, this will result in updating $u_1 = u_2 = 2$ and $v_4 = -1$. We have thus increased two variables by one unit and decreased one variable by one unit. In total, the dual lower bound was thus increased by one. The set E' of tight edges with respect to the new dual solution is now



Our (augmenting-path) algorithm will now find a perfect matching in this graph, which is optimal by Lemma 4 (since it only uses tight edges). In summary, our algorithm always maintains a dual feasible solution. We then solve the *unweighted* perfect matching problem on the edges allowed by Lemma 4. In the case of failure, we update the dual lower bound to a strictly better lower bound and repeat.

References

[1] Mateusz Golebiewski, Maciej Duleba: *Scribes of Lecture 5 in Topics in TCS 2015. http://theory.epfl.ch/courses/topicstcs/Lecture52015.pdf*