Topics in Theoretical Computer Science	March 16, 2015

Lecture 5

Lecturer: Ola Svensson — *Scribes: Mateusz Golebiewski, Maciej Duleba*

These lecture notes are partly based on *http://theory.epfl.ch/courses/topicstcs/Lecture10.pdf* and *http://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15859-f11/www/notes/lecture07.pdf.*

1 Introduction

Last lecture:

- Combinatorial Algorithm for maximum cardinality matching (König's Theorem and Hall's Theorem)
- Linear programming (Extreme point structure)

Theorem 1 (König 1931) *Let $M^\star$ be a maximum cardinality matching and $C^\star$ a minimum vertex cover of a bipartite graph. Then*

$$|M^\star| = |C^\star|$$

Theorem 2 (Hall 1935) *Given a bipartite graph $G = (A \cup B, E)$, there exist matching M s.t. all vertices in A are matched iff $\forall S \subseteq A \quad |S| \leq |N(S)|$, where $N(S) = \{b \in B \mid \exists a \in S \text{ with } (a,b) \in E\}$*

Remark Combinatorial algorithm either finds a matching M that matches all vertices in A or finds $S \subseteq A$ s.t. $|S| \leq |N(S)|$.

2 Linear Programming Duality

2.1 Intuition

Consider the following linear program:

$$\begin{aligned}
\textbf{Minimize: } & 7x_1 + 3x_2 \\
\textbf{Subject to: } & x_1 + x_2 \geq 2 \qquad \cdot y_1 \\
& 3x_1 + x_2 \geq 4 \qquad \cdot y_2 \\
& x_1, x_2 \geq 0
\end{aligned}$$

We are looking for the optimal solution OPT to this LP. To find the solution we may ask two types of questions (to find the upper and lower bound on OPT). Is there a solution of cost ≤ 10? (Is OPT ≤ 10)? Answer to this type of questions is quite simple; we just find a feasible solution to the LP with objective function ≤ 10 e.g. $x_1 = x_2 = 1$. Is there no better solution? (Is OPT ≥ 10)? Let's observe that we can bound the objective function using constraints that every feasible solution satisfies. From the first constraint we get:

$$7x_1 + 3x_2 \geq x_1 + x_2 \geq 2$$

Thus OPT ≥ 2. Similarly from the second constraint we get OPT ≥ 4. To make a better lower bound we will need to be more clever. Let's try to take some convex linear combination of the constraints with coefficients y_1, y_2 correspondingly. y_1, y_2 should be non-negative because multiplying a constraint by negative number would flip the inequality. Taking $y_1 = 1, y_2 = 2$ we obtain

$$7x_1 + 3x_2 \geq (x_1 + x_2) + 2(2x_1 + x_2) \geq 2 + 2 \cdot 4 = 10$$

For each constraint of the given LP we associate a dual variable y_i denoting the weight of the i-th constraint. What kind of variables can we take to get a valid lower bound? How should we pick coefficients to maximize lower bound for OPT? First of all we are interested in lower bounding objective function. Thus linear combination of primal constraints cannot exceed primal objective function. As mentioned earlier $y_i \geq 0$. In this way we get the following (dual) linear program for y_1, y_2:

$$\begin{array}{rll} \textbf{Maximize:} & 2y_1 + 4y_2 & \text{lower bound as tight as possible} \\ \textbf{Subject to:} & y_1 + 3y_2 \leq 7 & x_1 \text{ coefficient} \\ & y_1 + y_2 \leq 3 & x_2 \text{ coefficient} \\ & y_1, y_2 \geq 0 & \end{array}$$

Let's try now to formalize this approach.

2.2 General case

Consider the following linear program with n variables x_i for $i \in [1, n]$ and m constraints.

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

Then, the dual program has m variables y_j for $j \in [1, m]$ and n constraints:

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

Each variable y_j in the dual program corresponds to weight of one of constraints from the primal LP. We have n constraints in dual, one for every primal variable x_i.
Remark We showed how to produce dual program for minimization problem. Similar approach works also for maximization problems. Also every maximization problem can be reduced to a minimization one. We replace x_i with $-x_i$ in constraints and objective function obtaining minimization problem.

Remark One can verify that if we take the dual of the dual problem, we get back to the primal problem, as we should expect. Finding the dual linear program is an automatic procedure.

2.3 Duality Theorems

Let's focus now on the solutions of both LPs. In our example optimal solutions to primal and dual problems coincided. We now present two theorems that connect primal and dual solutions.

Theorem 3 (Weak Duality) *If x is* primal-feasible *(meaning that x is a feasible solution to the primal problem) and y is* dual-feasible*, then*

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

Proof Let's rewrite right hand side:

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

Here we used the fact that $x, y \geq 0$, otherwise inequality would be in the opposite direction.
■

This theorem tells us that every dual-feasible solution is a lower bound to any primal solution. This is intuitive: every primal feasible solution satisfies primal constraints, dual feasible solution gives us a way of lower bounding primal solution using primal constraints. Moreover from Weak Duality we can conclude that optimal solution to primal program is lower bounded by optimal solution to dual program. In fact optimal solutions to primal and duals linear programs coincide, what states the following theorem.

Theorem 4 (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 is unbounded (respectively infeasible), then the dual is infeasible (respectively unbounded).

2.4 Complementarity Slackness

There is a correspondence between primal and dual optimal solutions.

Theorem 5 *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*

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

Proof [An exercise.] We will apply the strong duality theorem to the weak duality theorem proof.

⇒ 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

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

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

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

$\Leftarrow$ Similarly to 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 to be optimal solutions to primal and dual linear program. Indeed for feasible solution $x^\star$ to primal we have by weak duality theorem:

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

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

■

3 Examples

3.1 Maximum cardinality matching

Let $G = (A \cup B, E)$ be a bipartite graph and let M be a matching. Let x_e be a variable corresponding to taking edge $e \in M$. We want to maximize cardinality of M assuring every vertex has at most one neighbouring edge in M. Writing those conditions in a form of LP gives us:

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

Thus the dual program looks like this:

$$\begin{aligned} \textbf{Minimize: } & \sum_{v \in A \cup B} y_v \\ \textbf{Subject to: } & y_a + y_b \geq 1 \quad \text{for } (a,b) \in E \\ & y_v \geq 0 \end{aligned}$$

One can easily notice that this LP is vertex cover relaxation.

3.2 Minimum cost perfect matching

Let $G = (A \cup B, E)$ be a bipartite weighted graph with weight function $c : E \to \mathbb{R}$. Let's denote c_{ab} for the cost of edge (a, b) with $a \in A \;\; b \in B$, if there is no such edge in E we set $c_{ab} = \infty$. Let M be a perfect matching x_{ab} be a variable for edge $(a, b) \in M$. Every vertex in G is adjacent to exactly one edge in M. We want to minimize the total cost of edges in M. This leads to the following linear program:

$$\begin{aligned}
\textbf{Minimize:} \quad & \sum_{a \in A, b \in B} c_{ab} x_{ab} \\
\textbf{Subject to:} \quad & \sum_{b \in B} x_{ab} = 1 \qquad \forall a \in A \\
& \sum_{a \in A} x_{ab} = 1 \qquad \forall b \in B \\
& x_{ab} \geq 0
\end{aligned}$$

And its dual

$$\begin{aligned}
\textbf{Maximize:} \quad & \sum_{a \in A} u_a + \sum_{b \in B} v_b \\
\textbf{Subject to:} \quad & u_a + v_b \leq c_{ab} \qquad \forall a \in A, b \in B
\end{aligned}$$

Here we don't have non-negativity constraints as in primal LP the constraints are given as equalities.
Remark Alternatively we can transform every equality in primal LP to two inequalities ($a = b \iff a \leq b \wedge -a \leq -b$). In dual LP we would obtain a difference of two new non-negative variables for each u.

Let's take a look at complementarity slackness for minimum cost perfect matching. Complementarity slackness tells us that if $x, (u, v)$ are feasible then they are both optimal if the following holds:

$$\begin{aligned}
x_{ab} > 0 \Rightarrow u_a + v_b = c_{ab} \quad & \forall a \in A, b \in B \\
u_a \neq 0 \Rightarrow \sum_{b \in B} x_{ab} = 1 \quad & \forall a \in A \\
v_b \neq 0 \Rightarrow \sum_{a \in A} x_{ab} = 1 \quad & \forall b \in B
\end{aligned}$$

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 whereas usually we would only have inequalities).
Remark The last two conditions have $\neq$ instead of $>$ as we don't have nonnegativity constraints on u, v. If we consider LP with every equality changed to two inequalities then in dual we would have two nonnegative variables $u_{a+}, u_{a-} \geq 0$ corresponding to each constraint and they can be collapsed: $u_{a+} - u_{a-} = u_a$ and similarly for v. Then the precondition would be that any of $u_{a+} > 0$ or $u_{a-} > 0$ holds which is stronger than $u_a \neq 0$ and similarly for v (also we don't need those preconditions to be equivalent since we are only applying complementarity slackness theorem in one direction).

In other words primal-feasible $x^\star$ is optimal if $\exists$ dual solution (u, v) such that

$$x^\star_{ab} \neq 0 \Rightarrow u_a + v_b = c_{ab} \quad \forall a \in A, b \in B$$

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

4 Minimum cost perfect matching algorithm (Hungarian)

Idea: Maintain a feasible dual solution (u, v). Try to construct a feasible primal solution that satisfies complementarity slackness and is thus optimal.

Algorithm:

- **Initialization:**

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

- **Iterative step:**
  - consider $G = (A \cup B, E')$ where $E' = \{(a,b) : u_a + v_b = c_{ab}\}$ (E is a set of all tight edges)
  - find a maximum cardinality matching in G
    - if it is perfect matching then we are done (this is a primal feasible solution and it satisfies complementarity slackness, because we chose only from edges in E' and all of them satisfied it)
    - otherwise the algorithm finds $S \subset A$ s.t. $|S| > |N(S)|$
  - we can choose small $\varepsilon > 0$ and improve the dual solution:

$$u'_a = \begin{cases} u_a + \epsilon & \text{if } a \in S \\ u_a & \text{if } a \notin S \end{cases}$$

$$v'_b = \begin{cases} v_b - \epsilon & \text{if } b \in N(S) \\ v_b & \text{otherwise if } b \notin N(S). \end{cases}$$

    edges in $S \times N(S)$ were tight and are unchanged $(+\varepsilon - \varepsilon)$
    edges in $(A \setminus S) \times (B \setminus N(S))$ are unchanged
    edges in $(A \setminus S) \times N(S)$ are decreased by ε
    edges in $S \times (B \setminus N(S))$ are increased by ε but they were not tight (they were not in E')
  - dual increases by $(|S| - |N(S)|)\varepsilon$
  - we should choose ε as large as possible for which the dual remains feasible, that is

$$\varepsilon = \min_{a \in S, b \notin N(S)} c_{ab} - u_a - v_b > 0.$$

Theorem 6 *The algorithm runs in time $O(n^3)$.*

4.1 An alternative proof that the bipartite perfect matching polytope is integer

We have shown that for any cost function c we can obtain minimum cost perfect matching and it will be integral. By carefully choosing the cost function, one can make any extreme point of the polytope to be the unique optimum solution to minimum cost perfect matching problem. This shows that all extreme points are integral i.e. it is an integer polytope. See Figure 1 for an example.

5 Maximum spanning tree

Given a connected graph $G = (V, E)$ with non-negative edge weights $w : E \to \mathbb{R}_{\geq 0}$.
Find a spanning tree T of maximum weight $w(T) = \sum_{e \in T} w(e)$.

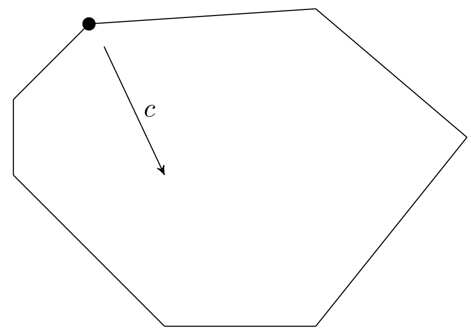

Figure 1: Example of a perfect matching polytope and a cost function vector c for the highlighted vertex.

5.1 Linear programs

Let's try to find a good LP for this problem. First of all set:

$$x_e = \begin{cases} 1 & \text{if } e \in T \\ 0 & \text{otherwise} \end{cases}$$

A good LP (that we will work with) for the maximum weight forest problem[1] is

$$\begin{aligned} \textbf{Maximize: } & \sum_{e \in E} w(e) x_e \\ \textbf{Subject to: } & \sum_{e \subseteq S} x_e \leq |S| - 1 \quad \forall S \subseteq V \text{ with } |S| \geq 2 \\ & x_e \geq 0 \end{aligned}$$

And it's dual program

$$\begin{aligned} \textbf{Minimize: } & \sum_{S \subseteq V} y_S (|S| - 1) \\ \textbf{Subject to: } & \sum_{S : e \subseteq S} y_S \geq w(e) \quad \forall e \in E \\ & y_S \geq 0 \end{aligned}$$

Remark By complementarity slackness we have that $x^\star$ is optimal if and only if there exists a feasible dual solution $y^\star$ s.t.

$$x_e^\star > 0 \Rightarrow \sum_{S : e \subseteq S} y_S^\star = w(e)$$
$$y_S^\star > 0 \Rightarrow \sum_{e : e \subseteq S} x_e^\star = |S| - 1$$

Now there arises a question: how to use this fact to find maximum weight spanning tree? Or can we use this in the proof of already known algorithm?

[1] Note that finding a maximum weight forest is equivalent to finding a maximum weight spanning tree if weights are positive.

5.2 Kruskal's Algorithm

Repeatedly pick the heaviest edge that does not create a cycle.

5.2.1 Primal and Dual solutions

Kruskal algorithm produces some feasible solution $x^\star$ to primal LP. Let's take a closer look to this solution and try to find a feasible dual solution $y^\star$ for which we will be able to prove complementarity slackness and hence get correctness of Kruskal's algorithm.

Primal solution: Let T be tree returned by Kruskal's algorithm. Define

$$x_e^\star = \begin{cases} 1 & \text{if } e \in T \\ 0 & \text{otherwise} \end{cases}$$

Note that $\sum_{e \in E} w(e) \cdot x_e = w(T)$.

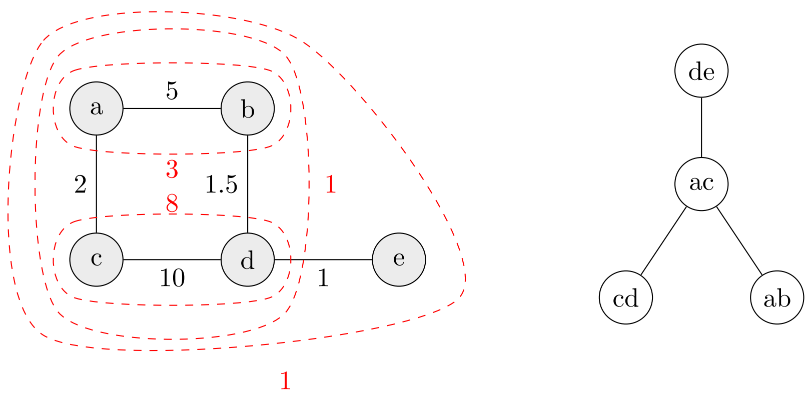

Figure 2: Example of Kruskal's Algorithm, the associated tree, and the dual solution.

Dual solution: Let $K = \{e_1, e_2 \ldots, e_{n-1}\}$ be the edges added by Kruskal's algorithm (they appear in K in the same order as in Kruskal's algorithm execution). This naturally induces a tree structure T_K where the cost of a subtree corresponds to the weight of an edge that connected the vertices in the subtree. Each edge from K corresponds to a vertex in this tree. Parent of every subtree is labelled with edge that connected this subtree into one connected component. Figure 2 shows an example of Kruskal's algorithm execution and associated with it tree T_K.
Notation. Let $V(e_j)$ be the set of vertices of G occurring in the subtree rooted at e_j. Let parent(e_j) be the parent of e_j in T_K. Define

$$y_S^\star = \begin{cases} w(e_j) - w(\text{parent}(e_j)) & \text{if } S = V(e_j) \\ 0 & \text{otherwise} \end{cases}$$

For the root of T_K we assume that weight of it's parent is equal to 0. We have set up $x^\star, y^\star$ and we want to prove that $x^\star$ is optimal. Due to remark from section 5.1 we are left with proving that $y^\star$ is a feasible dual solution, and that the complementarity slackness holds. Let's start with the second one.

5.2.2 Verifying complementarity slackness

First condition Let's assume $y^\star_S > 0$. Then $S = V(e_j)$ for some j and therefore $\sum_{e \subseteq S} x^\star_e = |S| - 1$. That's because $x^\star_e \neq 0$ for $e \subseteq S$ if and only if it's one of the edges in the associated subtree rooted at e_j. There are $|S| - 1$ such edges.

Second condition Let's assume $x^\star_e > 0$. For every such $x^\star_e$ we have that $e = e_j \in K$. So e_j corresponds to a node in T_K. Consider path $p_1, p_2 \ldots, p_k$ from $p_1 = e_j$ to the root p_k of T_K. Then

$$\sum_{S: e \subseteq S} y^\star_S = \sum_{i=1}^{k} y^\star_{V p_i} = \sum_{i=1}^{k} w(p_i) - w(\text{parent}(p_i)) = w(e_j).$$

First equality holds as only non-zero $y^\star_S$ are those occurring in T_K, then we have telescopic sum.

5.2.3 Feasibility of dual solution

Clearly $y^\star \geq 0$, as we take heavier edges earlier and they correspond to lower vertices in T_K. Consider any edge $e \in E$. Let j be the smallest index s.t. $e \in V(e_j)$. Consider path $p_1, p_2 \ldots, p_k$ from $p_1 = e_j$ to the root p_k of T_K. We have the following

$$\sum_{S: e \subseteq S} y^\star_S = \sum_{i=1}^{k} y^\star_{V p_i} = w(e_j) \geq w(e).$$

First two transformations are similar to those from last subsection. Last equality holds because $e \in V(e_j)$ and Kruskal's algorithm always adds edge of largest weight.

So in this way we showed that $x^\star$ and $y^\star$ satisfy conditions of complementarity slackness and $y^\star$ is feasible, thus $x^\star$ is optimal solution to the primal linear program proving correctness of Kruskal's algorithm.