Lecture 6: 18 September

Lecturer: J. van Leeuwen *Scribe: J.J. Franken*

6.1 Overview

We saw how *Linear Programming (LP)* is used in modelling optimization problems. In practice many problems are modelled as $0-1$ LPs or ILPs. In many cases it pays to analyze the *relaxed* LP, which may lead to a good approximation algorithm. In studying (general) LP models there is a powerful tool: *with every LP there is a* dual *LP that has the same optimum.* Sometimes the connection between the primal and the dual problem can be exploited. We show this in the design of an approximation algorithm for the Set Cover problem.

6.2 Modeling the Set Cover Problem

As leading example of how a dual LP is derived, we consider the (relaxed) Set Cover Problem (SCP). The problem consists of:

- a universe: $U = \{1, .., n\}$,
- a collection of subsets of U: $S = \{S_1, .., S_m\}$ with with costs/expenses: $c(S_i) = c_i > 0$, and
- a goal: determine a subcollection of the S_i of smallest total cost that covers U.

The *relaxed* primal LP model (SCP) is:

$$\min \quad z = \sum_{j=1}^{m} c_j x_j$$

subject to

$$\sum_{j=1}^{m} a_{ij} x_j \geq 1 \quad \text{for every } i \in U$$

$$x_j \geq 0 \quad \text{for all } j$$

where

$$a_{ij} = \begin{cases} 1 & \text{if } i \in S_j \ ; \\ 0 & \text{otherwise.} \end{cases}$$

The original problem SCP_I has the constraint '$x_i \in \{0,1\}$ instead of $x_i \geq 0$. Let z_I be the optimum solution of the SCP with $0-1$ constraint and z^* the, possibly non-integral, optimum of the relaxed SCP. Then: $z^* \leq z_I$.

Observation 6.1 *An optimal solution of the relaxed SCP has:* $x_j^* \leq 1$ *for every* j.

There is no integrality constraint in the SCP although eventually we do want to find a good, feasible $0-1$ solution.

6.2.1 The dual problem

We now design the dual of the SCP model: SCP^d. This is done by introducing a *dual variable* y_i with the ith constraint in the SCP $(1 \leq i \leq n)$, with $y_i \geq 0$. Multiply the left- and right-hand side of every constraint by the corresponding dual variable:

$$\sum_{j=1}^{m} a_{ij} x_j \cdot y_i \geq y_i$$

and add the n inequalities:

$$\sum_{i=1}^{n} \left(\sum_{j=1}^{m} a_{ij} x_j \cdot y_i\right) \geq \sum_{i=1}^{n} y_i \quad \Rightarrow \tag{1}$$

$$\Rightarrow \quad \sum_{j=1}^{m} \underbrace{\left(\sum_{i=1}^{n} a_{ij} y_i\right)}_{\leq c_j} \cdot x_j \geq \sum_{i=1}^{n} y_i \tag{2}$$

Compare equation 2 with the goal function z. Then for a fixed $x_j \geq 0$ it is interesting to 'vary' the y_i while keeping $\sum a_{ij} y_i \leq c_j$. Doing so keeps the right-hand side of equation 2 and *hence the lefthand side* below the value of $z = z(x)$, even if we maximize the left-hand side $(\sum y_i)$. But this maximization process is independent of the specific x_j's. Hence maximizing $\sum y_i$ under the given constraint amounts to approximating the minimum of the goal function z from below. This leads to the following *dual* SCP^d of SCP:

$$\max \quad z_d = \sum_{i=1}^{n} y_i$$

subject to

$$\sum_{i=1}^{n} a_{ij} y_i \leq c_j \quad \text{for every } j \in U$$

$$y_i \geq 0 \quad \text{for all } i$$

We conclude:

Theorem 6.2 (Weak duality theorem) *For every feasible solution x_j of SCP and feasible solution y_i of the dual SCP^d: $z_d \leq z$.*

Proof: This is evident from the given argument. Note that indeed:

$$\sum_i y_i = \sum_i 1 \cdot y_i \leq \sum_i (\sum_j a_{ij}\ x_j)\ y_i \leq \sum_j (\sum_i a_{ij}\ y_i)\ x_j \leq \sum_j c_j z_j.$$

■

The classical LP-theory learns that an even stronger result holds whenever an LP problem and its dual have a finite optimum, as in the case of the Set Cover problem.

Theorem 6.3 (Strong duality theorem) *SCP and SCP^d have the* same *optimum values, i.e. $z_d^* = z^*$.*

6.2.2 The complementary slackness conditions

One can characterize the optimum solutions of SCP and SCP^d precisely. Given values for the y_i, let

$$\overline{c}_j \quad = \quad c_j - \sum_1^n a_{ij} y_i.$$

Note that y_i is dual feasible if and only if $\overline{c}_j \geq 0$.

Theorem 6.4 *Let x_j and y_i be feasible solutions of SCP and SCP^d. These solutions are optimal iff*

$$\begin{aligned} x_j > 0 \quad &\Rightarrow \quad \overline{c}_j = c_j - \sum_i a_{ij} y_i = 0 \quad \text{(primal complementary slackness condition)} \\ y_i > 0 \quad &\Rightarrow \quad \sum_j a_{ij} x_j = 1 \qquad\qquad \text{(dual complementary slackness condition).} \end{aligned}$$

Proof: Let x_j and y_i be feasible. Then

$$0 \leq \sum_j \overline{c}_j x_j = \sum_j (c_j - \sum_i a_{ij} y_i) x_j = \sum_j c_j x_j - \sum_i (\underbrace{\sum_j a_{ij} x_j}_{\geq 1}) y_i \leq z - \sum_i y_i = z - z_d.$$

Suppose x and y are optimal solutions. Then $z - z^d = 0$ by strong duality and thus all inequalities become 'equalities'. Hence $\sum_j a_{ij} x_j = 1$ for all $y_i > 0$ and $\sum \overline{c}_j x_j = 0$, i.e. $\overline{c}_j = 0$ for every $x_j \geq 0$. Conversely, if the properties hold we have equality all the way and thus $z - z_d = 0$, implying that x and y are optimal. ■

The complementary slackness condition say that in the optimum: if $y_i > 0$ then there is *equality* in the corresponding constraint of the primal, and if $x_j > 0$ then there is *equality* in the corresponding constraint of the dual. This holds for every LP model and its dual with a finite optimum. It can be given the following intuitive meaning in the case of the Set Cover problem:

> *primal slackness:* if $x_j > 0$ ('set S_j is selected') then $\overline{c}_j = 0$ ('the costs c_j are fully paid by the y_i').
>
> *dual slackness:* if $y_i > 0$ ('element i pays'), then $\sum_j a_{ij}x_j = 1$ ('element i is fully covered').

Furthermore, if we have feasible solutions x and y with this property, then we know they are primal and dual optimal respectively. Of course in the optimum of SCP it can happen that the elements i are 'fractionally covered' by a number of different S_j's together. Thus we may not have a feasible solution of the original SCP yet.

This is the machinery, how can we use it?

6.3 Solving the Set Cover Problem

Consider any instance of the Set Cover problem. We will show the primal and dual LP model can be exploited in *approximating* the optimum solution. We show three approaches, the first two of which involve a call to a general LP-solver.

Definition 6.5 (Frequency) *Let $f = \max_i\{\sum_j^m a_{ij}\}$, the maximum number of times an element i of U occurs in different subsets S_j.*

Observation 6.6 *Let $\sum_j^m a_{ij}x_j \geq 1$ for a given i. Then there must be a j with $a_{ij} = 1$ and $x_j \geq \frac{1}{f}$. (Hint: there can be at most f non-zero terms.)*

6.3.1 Primal approaches

The first algorithm is a simple rounding algorithm: if any $x_j > 0$ in the optimum solution, then we pick set S_j in the cover!

> **Algorithm** SC-A
>
> **solve** the (relaxed) SCP, let the optimum solution be x_j^*.
>
> **for** every j **do**: $x_j^* > 0 \Rightarrow x_j = 1$ and pick the corresponding set S_j
>
> **return** the S_j's picked.

Theorem 6.7 (Hochbaum, 1982) *Algorithm SC-A computes a feasible solution to SCP_I and has performance ratio $\leq f$*

Proof: By observation 6.1. we have: $0 \leq x_j^* \leq 1$. As we only round *up*, it follows that the $0-1$ solution x_j is feasible, i.e. represents a set cover of U.

Now observe the following, where we use an optimum solution y_i^* of the dual to help in the estimating:

$$z^* = \sum_{j=1}^{m} c_j x_j^* \leq z_I^* \leq \sum_{j=1}^{m} c_j x_j = \sum_{\substack{j \\ \text{with } x_j^* > 0}} c_j = \sum_{\substack{j \\ \text{with } x_j^* > 0}} \left(\sum_{i=1}^{n} a_{ij} y_i^* \right) \leq$$
$$\sum_{j} \left(\sum_{i=1}^{n} a_{ij} y_i^* \right) = \sum_{i=1}^{n} \underbrace{\left(\sum_{j=1}^{m} a_{ij} \right)}_{\leq f} y_i^* \leq f \cdot \sum_{i=1}^{n} y_i^* = f \cdot z_d^* = f \cdot z^* \leq f \cdot z_I^*$$

Here primal complementary slackness condition is used in the fifth 'step'. We conclude that $\sum_j c_j x_j$, the cost of the solution, is within f from the optimal value z_I^*. ■

The effect of algorithm SC-A can be obtained equally well by solving the *dual* and checking which constraints in the dual are satisfied with equality: these constraints correspond precisely to the j with $x_j > 0$ (by complementary slackness).

> **Algorithm** SC-A'
>
> **solve** the SCP^d, let the optimum solution be y_i^*.
>
> **for** every j **do**: $\sum_i a_{ij} y_i^* = 1 \Rightarrow x_j = 1$ and pick the corresponding set S_j
>
> **return** the S_j's picked.

This algorithm computes the same result as SC-A.

The second algorithm is a simple rounding algorithm also but it is slightly more selective in determining which x_j's it will round: only if $x_j \geq \frac{1}{f}$ will we pick set S_j in the cover! Thus potentially we pick fewer subsets than in algorithm SC-A.

> **Algorithm** SC-B
>
> **solve** the (relaxed) SCP, let the optimum solution be x_j^*.
>
> **for** every j **do**: $x_j^* \geq \frac{1}{f} \Rightarrow x_j = 1$ and pick the corresponding set S_j
>
> **return** the S_j's picked.

Theorem 6.8 *Algorithm SC-B computes a feasible solution to SCP_I and has performance ratio $\leq f$*

Proof: Here we choose fewer sets, but we still round to a feasible solution as can be seen as follows. Consider any element i. As

$$\sum_j a_{ij} x_j^* \quad \geq \quad 1$$

it follows from observation 6.2 that there is a j with $a_{ij} = 1$ and $x_j \geq \frac{1}{f}$. Thus $x_j = 1$ and set S_j is picked, and i is covered by this S_j.

The performance ratio might be *lower* than for algorithm SC-A because we round fewer x_j's up. Thus the performance ratio will be $\leq f$ also. We can verify this more easily:

$$\begin{aligned} z^* &= \sum c_j x_j^* \leq z_I^* \leq \underbrace{\leq}_{\substack{bec.\\ feas.\\ sol.}} \underbrace{\sum_{j \text{ with}}}_{x_j^* \geq \frac{1}{f}} \underbrace{c_j x_j}_{\substack{\text{Cost of}\\ \text{SCP}_I \text{ sol.}}} \leq \\ &\leq f \cdot \underbrace{\sum_{j \text{ with}}}_{x_j^* \geq \frac{1}{f}} c_j x_j^* \leq f \cdot \textstyle\sum_j c_j x_j^* = f z^* \end{aligned}$$

■

A disadvantage of the primal approaches is that they still need a call to an LP-solver. In the next section we discuss an approach where this is not needed.

6.4 Primal-dual approach

The primal-dual algorithm solves the LP by trying to satisfy the complementary slackness conditions. The algorithm starts with a *dual feasible solution* and iterates while maintaining the following *invariant*:

1. for all j: $x_j \in \{0,1\}$
2. (dual feasibility) for all i: $y_i \geq 0$ and for all j: $\bar{c}_j = c_j - \sum_1^n a_{ij} y_i \geq 0$
3. (primal complementary slackness) for all j: $x_j > 0 \Rightarrow \bar{c}_j = 0$.
4. (*weak* dual complementary slackness) for all i: $y_i > 0 \Rightarrow \sum_{j=1}^n a_{ij} x_j \geq 1$.

(Recall that the *dual* complementary slackness condition was: $y_i > 0 \Rightarrow \sum_{j=1}^{n} a_{ij}x_j = 1$.)

If dual complementary slackness were satisfied for all i, then we can stop: an optimal (integral) solution is found. This is too much to hope for, and thus only a weak version is included in the invariant: it expresses that 'elements that pay' are 'certainly covered'. However, we will try to satisfy as many of the (weak) dual complementary slackness conditions as possible as the algorithm proceeds. Hopefully this will 'move' the x_j closer to a *feasible* solution of SCP_I, i.e. to a solution that covers the entire set U.

Lemma 6.9 *Suppose the algorithm maintains the invariant and succeeds in reaching a feasible solution x_j i.e. in covering U, then its performance ratio is $\leq f$.*

Proof: Suppose the algorithm reaches a feasible solution x_j $(1 \leq j \leq n)$. Let $J = \{j | x_j = 1\}$, thus $U = \bigcup_{j \in J} S_j$. Say the algorithm ends with y-values y_i. Then:

$$z^* \leq z_I^* \leq \sum_{j\in J} c_j x_j = \sum_{j \in J}\left(\sum_i a_{ij} y_i\right)\cdot 1 \leq \sum_i \left(\sum_j a_{ij}\right) y_i \leq f \cdot \sum_i y_i \leq f\cdot z_d^* = f \cdot z^* \leq f\cdot z_I^*.$$

Thus the computed solution is within a factor f from the optimum. ■

The techniques is typical for many primal-dual approaches. We now formulate the primal-dual algorithm for the Set Cover problem.

algorithm SC-BE

initialize

$y_i := 0,$
$\overline{c}_j = c_j - \sum_i a_{ij} y_i := c_j$ (prices paid)
$J := \emptyset$ (will contain the set-indices picked for the cover)
$x_j := 0$

while there are uncovered elements left **do**

pick an uncovered element $i \in U$ *(thus all x_j with $i \in S_j$ must be 0)*

pick an index k with $i \in S_k$ and $\overline{c}_k$ smallest, say $\overline{c}_k = \mu$
(thus pick a subset that is cheapest to add).

set $y_i := \mu$, add k to J and set $x_k := 1$

and repair the invariant

for all j **do** $\overline{c}_j := \overline{c}_j - a_{ij} \cdot y_i$
(this will make $c_k = 0$)

return the subsets S_j with $j \in J$.

Lemma 6.10 *Algorithm SC-BE maintains the primal-dual invariant.*

Proof: Clearly the initialization gives values to the y_i and x_j that satisfy the invariant. For example, for every j: $x_j > 0 \Rightarrow \overline{c}_j = 0)$ is vacuously true.

Suppose the invariant holds at the beginning of the **while**-loop. We argue that it holds at the end as well. We check the conditions.

1. Evident. The only change is that x_k becomes 1.

2. As i was uncovered, its y_i was 0. The y_i now becomes (possibly) non-zero, is set to $\overline{c}_k \geq 0$ and this remains ≥ 0. However *all* $\overline{c}_j$-values are updated. Consider the following for any j:

$$\overline{c}_j = c_j - \sum_t a_{tj} y_t = c_j - \sum_{t \neq i} a_{tj} y_t - a_{ij} y_i = c_j - \sum_{t \neq i} a_{tj} - 0 = c_j - \sum_{t \neq i} a_{tj}.$$

   This shows that updating $\overline{c}_j$ to $\overline{c}_j - a_{ij} \cdot y_i$ preserves the definition of $\overline{c}_j$. Furthermore, by the choice of k (and thus $c_k = \mu$) we have:

$$\overline{c}_j \geq \mu \Rightarrow (c_j - \sum_{t \neq i} a_{ij} y_t) - \mu \geq 0 \Rightarrow (c_j - \sum_{t \neq i} a_{ij} y_t) - a_{ij}\mu \geq 0 \Rightarrow \overline{c}_j - a_{ij}\mu \geq 0.$$

   The left-hand side of the latter inequality equals the updated value of $\overline{c}_j$, the righthand side shows that the update remains ≥ 0.

3. Note that x_k changes and becomes (possibly) non-zero. However, it is clear that $\overline{c}_k$ becomes 0, and thus complementary slackness is restored: $x_k > 0$ and $c_k = 0$. No other x_j values are changed.

4. At the beginning of the loop i was uncovered, i.e. we had $\sum_j a_{ij} x_j = 0$ and $y_i = 0$. By taking S_k into the cover, we get $\sum_j a_{ij} x_j = 1$ (because of the contribution of x_k). But at the same time we set y_i to $\mu \geq 0$. This satisfies the weak condition for i. For elements different from i the contribution of x_k can only increase the sum in the covering constraint. This weak complementary slackness is invariant as well.

■

Theorem 6.11 (Bar-Yehuda and Even, 1981) *Algorithm SC-BE computes a feasible solution to SCP$_I$, runs in time $\mathcal{O}(f \cdot n)$, and has a performance ratio $\leq f$.*

Proof: By construction the algorithm halts precisely when all elements are covered. Thus we get a feasible solution. As the algorithm maintains the invariant, Lemma 6.9 shows that it has a performance ratio $\leq f$. The algorithm is easily seen to be implementable in $\mathcal{O}(\sum_j |S_j|) = \mathcal{O}(f \cdot n)$ steps. ■

As a quite surprising consequence we can conclude also:

Corollary 6.12 *The weighted vertex cover problem can be solved by a linear-time approximation algorithm with a performance ratio* ≤ 2.

Proof: Vertex cover is a special case of set cover. Given a network instance, just take:

- $U =$ the set of all edges,
- subsets $S_u = \{$all edges incident to $u\}$ with $c(S_u) = 1$, for every $u \in V$,

This has $f = 2$ (every edge has two endpoints and is thus in 2 sets). Thus algorithm SC-BE runs in time $\mathcal{O}(f \cdot m)$ in this case, where m is the number of edges) ■

Interestingly, Gonzalez (1995) showed that the same result can be achieved by purely graph-theoretic means, without resorting to the LP-model.

Exercise. Apply algorithm SC-BE to solve/approximate the minimum weight dominating set problem. What is f in this case? (Hint: try max degree +1.)

6.5 Further remarks

Finally, it is known that by a different approximation algorithm the Set Cover Problem can be solved in easy polynomial time within a performance ratio of $\ln n$. Feige [2] showed that no polynomial time approximation algorithmm exists that achieves a performance ratio $\leq c \ln n$ time for a $c < 1$ unless all NP-problems can be decided in only slightly super-polynomial time.

References

[1] R. Bar-Yehuda, S. Even. A linear-time approximation algorithm for the weighted vertex cover problem. *Journal of Algorithms* 2 (1981) 198-203.

[2] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM* 45 (1998) 634-652.

[3] M.X. Goemans. *Linear programming*, course notes, October 1994.

[4] T.F. Gonzalez. A simple LP-free approximation algorithm for the minimum weight vertex cover problem, *Information Processing Letters* 54 (1995) 129-131.

[5] D.S. Hochbaum. Approximation algorithms for the set covering and vertex cover problems. *SIAM J. on Computing* 11 (1982) 555-556.