14 Rounding Applied to Set Cover

We will introduce the technique of LP-rounding by using it to design two approximation algorithms for the set cover problem, Problem 2.1. The first is a simple rounding algorithm achieving a guarantee of f, where f is the frequency of the most frequent element. The second algorithm, achieving an approximation guarantee of $O(\log n)$, illustrates the use of randomization in rounding.

Consider the polyhedron defined by feasible solutions to an LP-relaxation. For some problems, one can find special properties of extreme point solutions of this polyhedron, which can yield rounding-based algorithms. One such property is *half-integrality*, i.e., in each extreme point solution, every coordinate is $0, 1$, or $1/2$. In Section 14.3 we will show that the vertex cover problem possesses this remarkable property. This directly gives a factor 2 algorithm for weighted vertex cover; namely, find an optimal extreme point solution and round all the halves to 1. A more general property, together with an enhanced rounding algorithm, called iterated rounding, is introduced in Chapter 23.

14.1 A simple rounding algorithm

A linear programming relaxation for the set cover problem is given in LP(13.2). One way of converting a solution to this linear program into an integral solution is to round up all nonzero variables to 1. It is easy to construct examples showing that this could increase the cost by a factor of $\Omega(n)$ (see Example 14.3). However, this simple algorithm does achieve the desired approximation guarantee of f (see Exercise 14.1). Let us consider a slight modification of this algorithm that is easier to prove and picks fewer sets in general:

Algorithm 14.1 (Set cover via LP-rounding)

1. Find an optimal solution to the LP-relaxation.
2. Pick all sets S for which $x_S \geq 1/f$ in this solution.

Theorem 14.2 *Algorithm 14.1 achieves an approximation factor of f for the set cover problem.*

Proof: Let $\mathcal{C}$ be the collection of picked sets. Consider an arbitrary element e. Since e is in at most f sets, one of these sets must be picked to the extent of at least $1/f$ in the fractional cover. Thus, e is covered by $\mathcal{C}$, and hence $\mathcal{C}$ is a valid set cover. The rounding process increases x_S, for each set $S \in \mathcal{C}$, by a factor of at most f. Therefore, the cost of $\mathcal{C}$ is at most f times the cost of the fractional cover, thereby proving the desired approximation guarantee. □

The set cover instance arising from a vertex cover problem has $f = 2$. Therefore, Algorithm 14.1 gives a factor 2 approximation algorithm for the weighted vertex cover problem, thus matching the approximation guarantee established in Theorem 2.7.

Example 14.3 Let us give a tight example for Algorithm 14.1. For simplicity, we will view a set cover instance as a hypergraph: sets correspond to vertices and elements correspond to hyperedges (this is a generalization of the transformation that helped us view a set cover instance with each element having frequency 2 as a vertex cover instance).

Let $V_1, \ldots, V_k$ be k disjoint sets of cardinality n each. The hypergraph has vertex set $V = V_1 \cup \ldots \cup V_k$, and n^k hyperedges; each hyperedge picks one vertex from each V_i. In the set cover instance, elements correspond to hyperedges and sets correspond to vertices. Once again, inclusion corresponds to incidence. Each set has cost 1. Picking each set to the extent of $1/k$ gives an optimal fractional cover of cost n. Given this fractional solution, the rounding algorithm will pick all nk sets. On the other hand, picking all sets corresponding to vertices in V_1 gives a set cover of cost n. □

14.2 Randomized rounding

A natural idea for rounding an optimal fractional solution is to view the fractions as probabilities, flip coins with these biases and round accordingly. Let us show how this idea leads to an $O(\log n)$ factor randomized approximation algorithm for the set cover problem.

First, we will show that each element is covered with constant probability by the sets picked by this process. Repeating this process $O(\log n)$ times, and picking a set if it is chosen in any of the iterations, we get a set cover with high probability, by a standard coupon collector argument. The expected cost of cover picked in this manner is $O(\log n) \cdot \mathrm{OPT}_f \leq O(\log n) \cdot \mathrm{OPT}$, where OPT_f is the cost of an optimal solution to the LP-relaxation. Applying Markov's Inequality, we convert this into a high probability statement. We provide details below.

Let $\boldsymbol{x} = \boldsymbol{p}$ be an optimal solution to the linear program. For each set $S \in \mathcal{S}$, pick S with probability p_S, the entry corresponding to S in $\boldsymbol{p}$. Let $\mathcal{C}$ be the collection of sets picked. The expected cost of $\mathcal{C}$,

$$\mathbf{E}[c(\mathcal{C})] = \sum_{S \in \mathcal{S}} \mathbf{Pr}[S \text{ is picked}] \cdot c(S) = \sum_{S \in \mathcal{S}} p_S \cdot c(S) = \text{OPT}_f.$$

Next, let us compute the probability that an element $a \in U$ is covered by $\mathcal{C}$. Suppose that a occurs in k sets of $\mathcal{S}$. Let the probabilities associated with these sets be $p_1, \ldots, p_k$. Since a is fractionally covered in the optimal solution, $p_1 + p_2 + \cdots + p_k \geq 1$. Using elementary calculus, it is easy to show that under this condition, the probability that a is covered by $\mathcal{C}$ is minimized when each of the p_i's is $1/k$. Thus,

$$\mathbf{Pr}[a \text{ is covered by } \mathcal{C}] \geq 1 - \left(1 - \frac{1}{k}\right)^k \geq 1 - \frac{1}{e},$$

where e is the base of natural logarithms. Hence each element is covered with constant probability by $\mathcal{C}$.

To get a complete set cover, independently pick $d \log n$ such subcollections, and compute their union, say $\mathcal{C}'$, where d is a constant such that

$$\left(\frac{1}{e}\right)^{d \log n} \leq \frac{1}{4n}.$$

Now,

$$\mathbf{Pr}[a \text{ is not covered by } \mathcal{C}'] \leq \left(\frac{1}{e}\right)^{d \log n} \leq \frac{1}{4n}.$$

Summing over all elements $a \in U$, we get

$$\mathbf{Pr}[\mathcal{C}' \text{ is not a valid set cover}] \leq n \cdot \frac{1}{4n} \leq \frac{1}{4}.$$

Clearly, $\mathbf{E}[c(\mathcal{C}')] \leq \text{OPT}_f \cdot d \log n$. Applying Markov's Inequality (see Section B.2) with $t = \text{OPT}_f \cdot 4d \log n$, we get

$$\mathbf{Pr}[c(\mathcal{C}') \geq \text{OPT}_f \cdot 4d \log n] \leq \frac{1}{4}.$$

The probability of the union of the two undesirable events is $\leq 1/2$. Hence,

$$\mathbf{Pr}[\mathcal{C}' \text{ is a valid set cover and has cost} \leq \text{OPT}_f \cdot 4d \log n] \geq \frac{1}{2}.$$

Observe that we can verify in polynomial time whether C' satisfies both these conditions. If not, we repeat the entire algorithm. The expected number of repetitions needed is at most 2.

14.3 Half-integrality of vertex cover

Consider the vertex cover problem with arbitrary weights. Let $c : V \to \mathbf{Q}^+$ be the function assigning nonnegative weights to the vertices. The integer program for this problem is:

$$\begin{aligned} \text{minimize} \quad & \sum_{v \in V} c(v) x_v & & (14.1) \\ \text{subject to} \quad & x_u + x_v \geq 1, & (u,v) \in E \\ & x_v \in \{0,1\}, & v \in V \end{aligned}$$

The LP-relaxation of this integer program is:

$$\begin{aligned} \text{minimize} \quad & \sum_{v \in V} c(v) x_v & & (14.2) \\ \text{subject to} \quad & x_u + x_v \geq 1, & (u,v) \in E \\ & x_v \geq 0, & v \in V \end{aligned}$$

Recall that an *extreme point solution* of a set of linear inequalities is a feasible solution that cannot be expressed as convex combination of two other feasible solutions. A *half-integral solution* to LP (14.2) is a feasible solution in which each variable is 0, 1, or 1/2.

Lemma 14.4 *Let x be a feasible solution to LP (14.2) that is not half-integral. Then, x is the convex combination of two feasible solutions and is therefore not an extreme point solution for the set of inequalities in LP (14.2).*

Proof: Consider the set of vertices for which solution x does not assign half-integral values. Partition this set as follows.

$$V_+ = \left\{ v \,\middle|\, \frac{1}{2} < x_v < 1 \right\}, \quad V_- = \left\{ v \,\middle|\, 0 < x_v < \frac{1}{2} \right\}.$$

For $\varepsilon > 0$, define the following two solutions.

$$y_v = \begin{cases} x_v + \varepsilon, & x_v \in V_+ \\ x_v - \varepsilon, & x_v \in V_- \\ x_v, & \text{otherwise} \end{cases}, \quad z_v = \begin{cases} x_v - \varepsilon, & x_v \in V_+ \\ x_v + \varepsilon, & x_v \in V_- \\ x_v, & \text{otherwise.} \end{cases}$$

By assumption, $V_+ \cup V_- \neq \emptyset$, and so x is distinct from y and z. Furthermore, x is a convex combination of y and z, since $x = \frac{1}{2}(y+z)$. We will show, by choosing $\varepsilon > 0$ small enough, that y and z are both feasible solutions for LP (14.2), thereby establishing the lemma.

Ensuring that all coordinates of y and z are nonnegative is easy. Next, consider the edge constraints. Suppose $x_u + x_v > 1$. Clearly, by choosing ε small enough, we can ensure that y and z do not violate the constraint for such an edge. Finally, for an edge such that $x_u + x_v = 1$, there are only three possibilities: $x_u = x_v = \frac{1}{2}$; $x_u = 0, x_v = 1$; and $u \in V_+, v \in V_-$. In all three cases, for any choice of ε,

$$x_u + x_v = y_u + y_v = z_u + z_v = 1.$$

The lemma follows. □

This leads to:

Theorem 14.5 *Any extreme point solution for the set of inequalities in LP (14.2) is half-integral.*

Theorem 14.5 directly leads to a factor 2 approximation algorithm for weighted vertex cover: find an extreme point solution, and pick all vertices that are set to half or one in this solution.

14.4 Exercises

14.1 Modify Algorithm 14.1 so that it picks all sets that are nonzero in the fractional solution. Show that the algorithm also achieves a factor of f.
Hint: Use the primal complementary slackness conditions to prove this.

14.2 Consider the collection of sets, C, picked by the randomized rounding algorithm. Show that with some constant probability, C covers at least half the elements at a cost of at most $O(\text{OPT})$.

14.3 Give $O(\log n)$ factor randomized rounding algorithms for the set multicover and multiset multicover problems (see Section 13.2).

14.4 Give a (non-bipartite) tight example for the half-integrality-based algorithm for weighted vertex cover.

14.5 (J. Cheriyan) Give a polynomial time algorithm for the following problem. Given a graph G with nonnegative vertex weights and a valid, though not necessarily optimal, coloring of G, find a vertex cover of weight $\leq (2 - \frac{2}{k})\text{OPT}$, where k is the number of colors used.

13 Set Cover via Dual Fitting

In this chapter we will introduce the method of dual fitting, which helps analyze combinatorial algorithms using LP-duality theory. Using this method, we will present an alternative analysis of the natural greedy algorithm (Algorithm 2.2) for the set cover problem (Problem 2.1). Recall that in Section 2.1 we deferred giving the lower bounding method on which this algorithm was based. We will provide the answer below. The power of this approach will become apparent when we show the ease with which it extends to solving several generalizations of the set cover problem (see Section 13.2).

The method of dual fitting can be described as follows, assuming a minimization problem: The basic algorithm is combinatorial – in the case of set cover it is in fact the simple greedy algorithm. Using the linear programming relaxation of the problem and its dual, one shows that the primal integral solution found by the algorithm is fully paid for by the dual computed; however, the dual is infeasible. By *fully paid for* we mean that the objective function value of the primal solution found is at most the objective function value of the dual computed. The main step in the analysis consists of dividing the dual by a suitable factor and showing that the shrunk dual is feasible, i.e., it fits into the given instance. The shrunk dual is then a lower bound on OPT, and the factor is the approximation guarantee of the algorithm.

13.1 Dual-fitting-based analysis for the greedy set cover algorithm

To formulate the set cover problem as an integer program, let us assign a variable x_S for each set $S \in \mathcal{S}$, which is allowed 0/1 values. This variable will be set to 1 iff set S is picked in the set cover. Clearly, the constraint is that for each element $e \in U$ we want that at least one of the sets containing it be picked.

$$\text{minimize} \quad \sum_{S \in \mathcal{S}} c(S) x_S \tag{13.1}$$

$$\text{subject to} \quad \sum_{S:\ e \in S} x_S \geq 1, \qquad e \in U$$

$$x_S \in \{0,1\}, \qquad S \in \mathcal{S}$$

The LP-relaxation of this integer program is obtained by letting the domain of variables x_S be $1 \geq x_S \geq 0$. Since the upper bound on x_S is redundant, we get the following LP. A solution to this LP can be viewed as a fractional set cover.

$$\begin{aligned} \text{minimize} \quad & \sum_{S \in \mathcal{S}} c(S) x_S & & (13.2) \\ \text{subject to} \quad & \sum_{S:\ e \in S} x_S \geq 1, & e \in U \\ & x_S \geq 0, & S \in \mathcal{S} \end{aligned}$$

Example 13.1 Let us give a simple example to show that a fractional set cover may be cheaper than the optimal integral set cover. Let $U = \{e, f, g\}$ and the specified sets be $S_1 = \{e, f\}, S_2 = \{f, g\}, S_3 = \{e, g\}$, each of unit cost. An integral cover must pick two of the sets for a cost of 2. On the other hand, picking each set to the extent of $1/2$ gives a fractional cover of cost $3/2$. □

Introducing a variable y_e corresponding to each element $e \in U$, we obtain the dual program.

$$\begin{aligned} \text{maximize} \quad & \sum_{e \in U} y_e & & (13.3) \\ \text{subject to} \quad & \sum_{e:\ e \in S} y_e \leq c(S), & S \in \mathcal{S} \\ & y_e \geq 0, & e \in U \end{aligned}$$

Intuitively, why is LP (13.3) the dual of LP (13.2)? In our experience, this is not the right question to be asked. As stated in Section 12.1, there is a purely mechanical procedure for obtaining the dual of a linear program. Once the dual is obtained, one can devise intuitive, and possibly physically meaningful, ways of thinking about it. Using this mechanical procedure, one can obtain the dual of a complex linear program in a fairly straightforward manner. Indeed, the LP-duality-based approach derives its wide applicability from this fact.

An intuitive way of thinking about LP (13.3) is that it is packing "stuff" into elements, trying to maximize the total amount packed, subject to the constraint that no set is overpacked. A set is said to be *overpacked* if the total amount packed into its elements exceeds the cost of the set. Whenever the coefficients in the constraint matrix, objective function, and right-hand side are all nonnegative, the minimization LP is called a *covering LP* and