CS 270: Combinatorial Algorithms and Data Structures Spring 2023

Lecture 17 — March 14, 2023

Prof. Jelani Nelson *Scribe: Aadil Manazir*

1 Outline

1. Online Primal/Dual (Set Cover)
2. Approximation Algorithms (Dual Fitting for Weighted Set Cover)

2 Online Set Cover

2.1 Weighted Set Cover Problem

The weighted set cover problem is stated as follows. We are given the universe $[n]$ and a collection of subsets $S_1, S_2, \ldots, S_m \subset [n]$. Each set has a corresponding weight c_S. We want to choose a subcollection $A \subset [m]$ s.t. $\bigcup_{i\in A} S_i = [n]$ and $\sum_{i\in A} c_{S_i}$ is minimized.

2.2 Online Set Cover Problem

The following is a description of the online set cover problem. We are given that there are m subsets, but we do not know their contents. Then, we are shown each element of the universe one-by-one. When element i is shown, we are told the S_j such that $i \in S_j$. After seeing i, we must make an irrevocable decision to choose an S_j such that i is included in the cover. We still want to choose a subcollection $A \subset [m]$ s.t. $\bigcup_{i\in A} S_i = [n]$ and $\sum_{i\in A} c_{S_i}$ is minimized.

2.3 Primal and Dual Formulations of Weighted Set Cover

The primal LP for weighted set cover is as follows:

$$\begin{aligned} \min_{x\in\mathbb{R}^m} \quad & \sum_S c_S \cdot x_S \\ \text{s.t.} \quad & \forall i \in [n], \\ & \sum_{S\ni i} x_S \geq 1 \\ & x \geq 0 \end{aligned}$$

The dual LP for weighted set cover is as follows:

$$\max_{y \in \mathbb{R}^n} \quad \sum_{i=1}^{n} y_i$$
$$\text{s.t.} \quad \forall S,$$
$$\sum_{i \in S} y_i \leq c_S$$
$$y \geq 0$$

2.4 Randomized Online Algorithm for Unweighted Set Cover

We will discuss a randomized online algorithm for unweighted set cover with competitive ratio $\mathcal{O}(\log n \log m)$ from [Alon, Awerbuch, Azar, Buchbinder, Naor '09]. This paper also discusses that the online unweighted set cover problem has a competitive ratio lower bound of $\Omega(\frac{\log n \log m}{\log \log n + \log \log m})$, but we won't show this in class. We will discuss two analyses of the same algorithm.

2.5 Fractional Algorithm Description

We will first detail and analyze the fractional algorithm and then discuss how to make its solution integral.

Algorithm 1 Online Unweighted Set Cover with Fractional Solution

1: $x \leftarrow (1/m, \ldots, 1/m)$
2: Upon seeing $i \in S_{j1}, \ldots, S_{jk}$, $\forall S \in S_{j1}, \ldots, S_{jk}$, until the ith constraint in the primal is satisfied, evolve x_S simultaneously in continuous time according to $\frac{d}{dt} x_S(t) = x_S(t)$.

In other words, after i is revealed, $x_{S_{jr}} \leftarrow e^t x_{S_{jr}}$, $r \in 1, \ldots, k$, where t is the minimum t s.t. $e^t \sum_{S \ni i} x_S \geq 1$. Clearly, we choose $t = -\ln \sum_{S \ni i} x_S$.

2.6 Primal Only Analysis

We will show that the fractional algorithm has competitive ratio of $O(\log m)$. To do so, we will prove two claims. The first is the following:

Claim 2.1. $\frac{d}{dt}(\text{cost}_p(x)) \leq 1$ when we evolve.

Proof. Since the derivative is a linear operator,

$$\frac{d}{dt}(\text{cost}_p(x)) = \frac{d}{dt}(\sum_{r=1}^{k} x_{S_{jr}}(t))$$

$$= \sum_{r=1}^{k} \frac{d}{dt} x_{S_{jr}}(t)$$
$$= \sum_{r=1}^{k} x_{S_{jr}}(t) \leq 1.$$

In fact, this inequality is strict as we only evolve when $\sum_{r=1}^{k} x_{S_{jr}}(t) < 1$. $\square$

The second claim is the following:

Claim 2.2. The total time we are in evolution mode is $\leq$ OPT $\cdot \ln m$.

Proof. Notice that every time we evolve, we are evolving an element of OPT; if OPT does not contain any x_S that is evolving, then it cannot cover the universe. Furthermore, recall that evolving for t time causes $x_S \leftarrow e^t x_S$. Hence, the maximum amount of time that x_S can be evolved is $\ln(\frac{1}{1/m}) = \ln m$. Hence, the total evolution time is $\leq$ OPT $\cdot \ln m$. $\square$

Note that the initial cost $= \mathbf{1}^T[1/m, \ldots, 1/m] = 1$. Since we have that the total amount of time evolving is $\leq \text{OPT} \cdot \ln m$, and $\frac{d}{dt}(\text{cost}_p(x)) \leq 1$ during evolution, the final cost $\leq 1 + 1 \cdot \text{OPT} \cdot \ln m = 1 + \text{OPT} \cdot \ln m$.

2.7 Primal/Dual Analysis

We will now use primal/dual analysis to show that the fractional algorithm has a $\mathcal{O}(\log m)$ competitive ratio.
While covering i in the original algorithm, in the dual problem, we now evolve y_i according to $\frac{d}{dt}(y_i(t)) = 1$. We initialize the y_i to 0. Hence, we have

$$\begin{aligned} \frac{d}{dt}(\text{cost}_d(y)) &= \frac{d}{dt} \sum_{i=1}^{n} y_i \\ &= \sum_{i=1}^{n} \frac{d}{dt}(y_i) \\ &= 1 \end{aligned}$$

However, this yields a contradiction. Recall that $\frac{d}{dt}(\text{cost}_p(x)) < 1$, but $\frac{d}{dt}(\text{cost}_d(x)) = 1$. Hence, at the end, $\text{cost}_p(x) < \text{cost}_d(x)$. However, this contradicts weak duality, which states that

$$\text{cost}_d(y) \leq \text{OPT} \leq \text{cost}_p(x)$$

The issue with the prior reasoning is that we didn't check y for dual-feasibility. In fact, y was not dual-feasible. We now provide a y that is dual-feasible.

Claim 2.3. $\frac{y}{\ln m}$ is dual-feasbile

Proof. We will show that $\forall S, \sum_{i \in S} y_i \leq \ln m$. Recall that we increase $\sum_{i \in S} y_i$ when we see an uncovered $i \in S$ online. The total time spent evolving any y_i where $i \in S$ is $\leq \ln m$ since x_S cannot evolve for more than $\ln m$ time (discussed above). Since $\sum_{i \in S} y_i$ evolves at a rate of 1, we have $\sum_{i \in S} y_i \leq \ln m$. Hence,

$$\sum_{i \in S} \frac{y_i}{\ln m} = \frac{1}{\ln m} \sum_{i \in S} y_i \leq 1 = c_S$$

so $\frac{y}{\ln m}$ is dual-feasible. $\square$

Hence, we have the following inequalities. The first follows from weak-duality, the second by the definition of OPT, and the third from the analysis of the derivatives above.

$$\begin{aligned}
&\text{cost}_d(\frac{y}{\ln m}) \leq \text{OPT} \leq \text{cost}_p(x) \leq \text{cost}_d(y) \\
&\implies \text{cost}_p(x) \leq \text{cost}_d(y) \leq \text{OPT} \cdot \ln m \\
&\implies \text{cost}_p(x) \leq \text{OPT} \cdot \ln m
\end{aligned}$$

which is our desired result.

2.8 Integral Solution

We now discuss how to obtain an integral solution from our fractional solution. Our integral solution algorithm will have an expected competitive ratio of $\mathcal{O}(\log n \log m)$.
Let $r \in \mathbb{N}$ be a chosen parameter. Then, for each S, sample i.i.d $\alpha_1^{(S)}, \ldots, \alpha_r^{(S)} \sim U[0,1]$. Now, run the fractional algorithm. If at any time during the evolution, x_S is greater than $\min(\alpha_i^{(S)}), i \in 1, \ldots, r$, we take x_S in the collection (set $x_S = 1$). Now, we show the following claim regarding the cost of our integral solution:

Claim 2.4. $\mathbb{E}[\text{cost of integral solution}] \leq r \cdot \mathbb{E}[\text{cost}_p(x)]$, where $\text{cost}_p(x)$ is the cost of the fractional solution.

Proof.

$$\begin{aligned}
\text{cost of int soln} &= \sum_S \mathbb{1}(\text{took set } S) \\
\implies \mathbb{E}[\text{cost of int soln}] &= \sum_S \mathbb{P}(\text{took set } S)
\end{aligned}$$

By union bound, we have

$$\begin{aligned}
\sum_S \mathbb{P}(\text{took set } S) &\leq \sum_S r \cdot x_S \\
&= r \sum_S x_S \\
&= r \cdot \text{cost}_p(x)
\end{aligned}$$

□

We now show that the integral solution is feasible with high probability if we choose r on the order of $\log n$. With the following claim, we have the result that the expected competitive ratio of the integral solution is $\mathcal{O}(\log m \log n)$.

Claim 2.5. Choose r on the order of $\ln n$. Then the integral solution is feasible with high probability.

Proof. Choose $r = 100 + \ln n$ (100 is an arbitrary big number). Then, $\forall i \in [n]$,

$$\begin{aligned}
\mathbb{P}(i \text{ not covered}) &= \prod_{S \ni i} (1 - x_S)^r \\
&\leq \prod_{S \ni i} e^{-r \cdot x_S} \\
&= e^{-r \cdot \sum_{S \ni i} x_S} \\
&\leq e^{-r} \\
&= \frac{e^{-100}}{n}
\end{aligned}$$

By union bound, we have $\mathbb{P}(\text{Universe not covered}) \leq e^{-100}$; clearly, this probability can be made arbitrarily small. □

3 Approximation Algorithms (Weighted Set Cover)

We will now discuss an approximation algorithm for weighted set cover from [Chvátal '79].

3.1 Greedy Algorithm for Weighted Set Cover

While there exists an uncovered element, take the S that minimizes $\frac{c_S}{\text{\# newly covered elements}}$.

3.2 Primal/Dual Analysis

When we take S into our solution, we do the following

- $x_S \leftarrow 1$
- for each newly covered i, $y_i \leftarrow \frac{c_S}{\text{\# of newly covered elements}}$

In each iteration, we add c_S to the objective of the primal. We also add c_S to the objective of the dual. Hence, the above x, y must satisfy strong duality, as long as y is feasible. Unfortunately, y isn't feasible, so we must dual-fit y. We scale down by H_n, the nth harmonic number.

Claim 3.1. $\frac{y}{H_n}$ is dual feasible.

Proof. We will show that $\forall S, \sum_{i \in S} y_i \leq c_S \cdot H_n$. Enumerate the items in S in the order that they were covered: $e_1, e_2, \ldots, e_k \in [n]$. Consider any $e_i \in S$. Since we could have taken S to cover e_i at a price of $\frac{c_S}{k-i+1}$, the greedy price for e_i is $\leq \frac{c_S}{k-i+1}$. Hence,

$$\begin{aligned}
\sum_{i \in S} y_i &\leq c_S \cdot \left(\frac{1}{k} + \frac{1}{k-1} + \cdots + 1\right) \\
&= c_S \cdot H_k \\
&\leq c_S \cdot H_n
\end{aligned}$$

Hence, $\frac{y}{H_n}$ is dual feasible. □

Finally, we can apply the same chain of inequalities in section 2.7 above to get $\text{cost}_p(x) \leq \text{OPT} \cdot H_n$.

References

[1] Noga Alon, Baruch Awerbuch, Yossi Azar, Niv Buchbinder, and Joseph (Seffi) Naor. The Online Set Cover Problem. *SIAM Journal on Computing*, Vol.39, No. 2, Pg. 361-370

[2] Vasek Chv´atal. A greedy heuristic for the set-covering problem. Mathematics of OperationsResearch, 4(3):233–235, 1979.