IOE 691: Approximation & Online Algorithms

Lecture Notes: Online Primal-Dual

Instructor: Viswanath Nagarajan

Many discrete optimization problems can be cast as covering or packing integer programs. As for (offline) approximation algorithms, LP relaxations turn out to be useful in online algorithms as well. However, it is highly non-trivial to even solve LPs online, and good competitive ratios are only possible for structured LPs such as covering/packing LPs.

1 Fractional covering

We now consider the *covering* LP which has the form:

$$\min c^T x \;:\; Ax \geq b,\, x \geq 0.$$

Above, $x \in \mathbb{R}^n$ denotes the decision variables. The costs $c \in \mathbb{R}^n$, right-hand-sides $b \in \mathbb{R}^m$ and constraint matrix $A \in \mathbb{R}^{m\times n}$ are all *non-negative.* By scaling the constraint matrix, we can ensure that all costs and RHS are one. So we obtain the following. pure-covering LP and its dual:

$$\begin{aligned} (P) \quad = \quad & \min \sum_{j=1}^{n} x_j & \qquad (D) \quad = \quad & \max \sum_{i=1}^{m} y_i \\ \sum_{j=1}^{n} a_{ij} x_j \geq 1, & \quad \forall i \in [m], & \sum_{i=1}^{m} a_{ij} y_i \leq 1, & \quad \forall j \in [n] \\ & x \geq 0. & & y \geq 0. \end{aligned}$$

In the online setting, rows of A (constraints in P and variables in D) arrive over time, and we are not allowed to decrease any variable. This “monotonicity” constraint prevents us from simply resolving an new LP after each new constraint. We will see that this also corresponds to unknown/online input in many applications.

Let d be the maximum number of non-zero entries in any row of A and $\rho = \max_j \frac{\max a_{ij}}{\min a_{ij}}$ be the “aspect ratio” of A. The main result here is:

Theorem 1.1 *There is a* $(1 + o(1)) \ln(1 + \rho d)$ *competitive algorithm for covering LPs.*

This algorithm is best seen as a continuous-time algorithm. (It can also be made to run in polynomial time by discretizing.) Initially, all primal/dual variables are 0. We use t to denote a continuous time variable. Let $\delta > 0$ be a parameter to be set later (to optimize the competitive ratio).

When constraint i arrives, repeat the following continuously:

$$\frac{\partial}{\partial t} x_j = a_{ij} x_j + \delta \;\text{ for all } j : a_{ij} > 0 \qquad \text{and} \qquad \frac{\partial}{\partial t} y_i = 1,$$

until $\sum_j a_{ij} x_j \geq 1$.

Let $x_j(i^-)$ and $x_j(i^+)$ denote the variable values just before and just after the arrival of constraint i. Note that the rate of primal-objective increase is:

$$\frac{\partial}{\partial t}P = \sum_{j=1}^{n} a_{ij}x_j + d\delta \le 1 + d\delta = (1+d\delta)\cdot\frac{\partial}{\partial t}D.$$

Above, the first equality uses the fact that each constraint/row i has at most d non-zero a_{ij}. The inequality is by the update rule, which stops raising variables when the constraint is satisfied. The last equality is by definition of the dual-objective.

For any j and i with $a_{ij} > 0$, we have $\partial y_i = \frac{\partial x_j}{a_{ij}x_j+\delta}$. Integrating,

$$a_{ij}y_i = \ln\left(\frac{x_j(i^+)+\delta/a_{ij}}{x_j(i^-)+\delta/a_{ij}}\right) \le \ln\left(\frac{x_j(i^+)+\delta/a_{max,j}}{x_j(i^-)+\delta/a_{max,j}}\right) \tag{1}$$

Observe that (1) is trivially true when $a_{ij} = 0$.

Adding over all constraints i,

$$\sum_{i=1}^{m} a_{ij}y_i = \sum_{i=1}^{m} \ln\left(\frac{x_j(i^+)+\delta/a_{max,j}}{x_j(i^-)+\delta/a_{max,j}}\right) = \ln\left(\frac{x_j+\delta/a_{max,j}}{\delta/a_{max,j}}\right) = \ln\left(1+a_{max,j}x_j/\delta\right).$$

Note that $x_j \le 1/a_{min,j}$, because at this value *every* constraint involving x_j would be trivially satisfied. Combined with the above equation, this implies that $\sum_{i=1}^{m} a_{ij}y_i \le \ln(1+\rho/\delta)$ for all $i \in [m]$, where we used the definition of the aspect-ratio ρ.

It follows that x and y are monotone primal and dual solutions where x is feasible in the primal and y approximately satisfies all the dual-constraints. In particular, if we scale down all dual variables y by a factor $\ln(1+\rho/\delta)$ then we would obtain a feasible dual solution. After this scaling, the final primal and dual objectives are within a factor $(1+d\delta)\cdot\ln(1+\rho/\delta)$. Setting $\delta = 1/(d\ln(1+\rho d))$ and using weak-duality, we obtain Theorem 1.1.

Notice that we get an online algorithm for the dual (packing) LP as well: the dual variables y are also monotonically increasing. However, the dual constraints are only satisfied approximately (after relaxing by an $\ln(d\rho)$ factor). Indeed, it is easy to see that any reasonable online algorithm for a packing LP (even with a single constraint) must violate its constraints by such a factor.

1.1 Lower bound for covering LPs

It turns out that no online algorithm can do better than the above algorithm, even when all entries in the constraint matrix are 0 or 1 (so $\rho = 1$). Consider any online algorithm for covering LPs. The online instance has n variables with objective $\sum_{j=1}^{n} x_j$, and the following constraints:

1. The first constraint is $\sum_{j=1}^{n} x_j \ge 1$. After the online algorithm raises its variables to satisfy this, re-number the variables by increasing value, i.e. $x_1 \le x_2 \le \cdots x_n$. Note that $x_n \ge \frac{1}{n}$.
2. The second constraint is then $\sum_{j=1}^{n-1} x_j \ge 1$, where the largest variable x_n is excluded. Again, we re-number variables in increasing value. Note that $x_{n-1} \ge \frac{1}{n-1}$.
3. The i^{th} constraint is $\sum_{j=1}^{n-i+1} x_j \ge 1$. After the online algorithm's move, we re-number so that $x_1 \le x_2 \le \cdots x_{n-i+1}$. Note that $x_{n-i+1} \ge \frac{1}{n-i+1}$

So, it is clear that the total cost of the algorithm is at least $\frac{1}{n} + \frac{1}{n-1} + \cdots \frac{1}{1} \approx \ln n$.

On the other hand, the optimal LP solution, knowing this sequence of constraints, just sets variable $x_1 = 1$ (according to the final numbering) and all other variables to 0. So the optimal cost is one. Note that in this instance $a_{max} = a_{min} = 1$ and the row-sparsity $d = n$.

To summarize,

Theorem 1.2 *No online algorithm for covering LPs can be better than* $\ln(d)$ *competitive.*

1.2 Removing dependence on the aspect ratio

In some applications, the aspect ratio ρ is large and it is undesirable to have it appear in the competitive ratio. It turns out that we can actually get a logarithmic ratio depending only on the row-sparsity d. However, this algorithm is more complex and involves maintaining duals in a non-monotone manner.

Consider again the covering LP (P) and its dual (D). We will again present the algorithm using a continuous time variable t. All primal/dual variables are initially 0. Let $\theta \approx 2 \ln d$ be the allowed violation of the dual constraints.

When constraint i arrives, repeat the following continuously until $\sum_{j=1}^{n} a_{ij} x_j \geq 1$.

1. Update $\frac{\partial}{\partial t} x_j = a_{ij} x_j + \delta$ for all j with $a_{ij} > 0$.
2. Update $\frac{\partial}{\partial t} y_i = 1$.
3. For each j with $y^T A_j = \sum_{i=1}^{m} a_{ij} y_i = \theta$,

$$\text{let } m_j := \arg\max_{k \leq i} \{a_{kj} : y_k > 0\}, \text{ and update } \frac{\partial}{\partial t} y_{m_j} = -\frac{a_{ij}}{a_{m_j,j}}.$$

As before, $x_j(i^-)$ and $x_j(i^+)$ denote the variable values just before and just after the arrival of constraint i. Call dual constraint $j \in [n]$ *tight* if $y^T A_j = \theta$. We need to decrease some dual variables to avoid further violation of the tight dual-constraints.

Note that the rate of primal increase is $\frac{\partial}{\partial t} P \leq 1 + \delta d$, just as before.

The rate of increase in the dual objective is:

$$\frac{\partial}{\partial t} D \geq 1 - \sum_{j:tight} \frac{a_{ij}}{a_{m_j,j}}. \tag{2}$$

Note that the increase is due to variable y_i (for the current primal-constraint i) and the decrease is due to all tight dual-constraints.

Consider any dual variable y_i. Note that it remains 0 until the i^{th} primal constraint arrives. Immediately after the i^{th} primal constraint, we have, just as for (1),

$$a_{ij} y_i \leq \ln \left(\frac{x_j(i^+) + \delta / a_{max,j}}{x_j(i^-) + \delta / a_{max,j}} \right). \tag{3}$$

Subsequently, the dual variable y_i *does not* increase (because we only decrease previous duals). So the above bound holds at all points in the algorithm.

We now bound the last term in (2). Consider any tight dual-constraint j and let

$$S = \{k \le i : y_k > 0\}$$

denote all positive dual variables. Note that $a_{m_j,j} = \max_{k\in S} a_{k,j}$. So, using (3) we have:

$$\theta = y^T A_j = \sum_{k\in S} a_{kj} y_k \le \sum_{k\in S} \ln\left(\frac{x_j(k^+) + \delta/a_{m_j,j}}{x_j(k^-) + \delta/a_{m_j,j}}\right) \le \ln\left(1 + a_{m_j,j}x_j/\delta\right) \le \ln\left(1 + \frac{a_{m_j,j}}{a_{ij}\delta}\right),$$

where we used $x_j \le 1/a_{ij}$ as constraint i is still not covered. Rearranging, we get:

$$\frac{a_{m_j,j}}{a_{ij}} \ \ge\ \delta\left(e^\theta - 1\right), \qquad \forall \text{ tight } j.$$

Combined with (2),

$$\frac{\partial}{\partial t} D \quad \ge \quad 1 - \sum_{j:tight, a_{ij}>0} \frac{1}{\delta(e^\theta - 1)} \quad \ge \quad 1 - \frac{d}{\delta(e^\theta - 1)}.$$

Setting

$$\delta = \frac{1}{d\ln(1+d)} \qquad \text{and} \qquad \theta = \ln\left(1 + (d\ln(1+d))^2\right) \approx 2\ln(d),$$

we get $\frac{\partial}{\partial t} D \ge 1 - \frac{1}{\ln d}$ and $\frac{\partial}{\partial t} P \le 1 + \frac{1}{\ln d}$. Hence, we obtain:

Theorem 1.3 *There is a $(2 + o(1))\ln(1+d)$ competitive algorithm for covering LPs.*

2 Set cover

Given n sets (with costs c_j) that need to cover m elements. Elements arrive online and need to be covered immediately upon arrival. Consider the IP:

$$\min \sum_{j=1}^{n} c_j \cdot x_j$$

$$\sum_{j:i\in S_j} x_j \ge 1, \quad \forall i$$

$$x \in \{0,1\}^n.$$

We can obtain a fractional solution x online with competitive ratio $O(\log d)$. Let $\theta = O(\log \ell)$ where ℓ is the size of the maximum set (i.e. column sparsity). The online rounding first chooses a random value $\tau_j \in [0,1]$ for each set j independently. Then, set j is selected into solution T when $x_j > \tau_j/\theta$. Upon arrival of element i, if T does not cover it then we select the min-cost set covering i and add it to solution R. The final solution is $T \cup R$ which is clearly feasible. The expected cost of T is $O(\log \ell)$ times the fractional value. It remains to bound the expected cost of R.

Fix any element i and let D_i denote the cost of the min-cost set containing i. Note that $D_i \le \sum_{j:i\in S_j} c_j x_j$ as x covers i fractionally. The probability that i is *not* covered by T is at most $\frac{1}{2\ell}$ by Chernoff bound. So the expected cost of R is at most:

$$\sum_i D_i \cdot \frac{1}{2\ell} \le \frac{1}{2\ell} \sum_i \sum_{j:i\in S_j} c_j x_j = \frac{1}{2\ell} \sum_j c_j x_j \cdot |S_j| \le \frac{1}{2} c^T x.$$

Theorem 2.1 *There is an* $O(\log d \ \log \ell)$ *randomized online algorithm for set cover.*

This result can also be extended to online facility location.