Advanced topics on Algorithms

Luciano Gualà
www.mat.uniroma2.it/~guala/

Approximation algorithms
Episode III

Linear Programming: rounding

minimum Set Cover problem

Input:

- universe U of n elements
- a collection of subsets of U, $\mathcal{S}=\{S_1,\dots,S_k\}$
- each $S \in \mathcal{S}$ has a positive cost $c(S)$

Feasible solution:

a subcollection $\mathcal{C} \subseteq \mathcal{S}$ that covers U (whose union is U)

measure (min):

cost of $\mathcal{C}$: $\sum_{S \in \mathcal{C}} c(S)$

frequency of an element e: number of sets e belongs to

f: frequency of the most frequent element

an Integer Linear Programming (ILP) formulation of SC

LP-relaxation

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

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

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

relax with
$x_S \geq 0$ & $\underbrace{x_S \leq 1}_{\text{redundant}}$

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

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

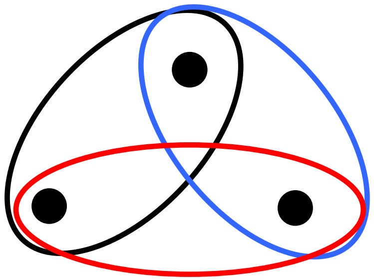
$$x_S \geq 0 \qquad S \in \mathcal{S}$$

a feasible solution is
a fractional SC

OPT_f: cost of the min fractional SC

$$\boxed{OPT_f \leq OPT}$$

an example



- 3 elements
- 3 sets
- all sets have cost 1

OPT=2

OPT_f=1.5

set all x_S to $\frac{1}{2}$

LP-relaxation

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

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

$$x_S \geq 0 \qquad S \in \mathcal{S}$$

which algorithm do you know to solve a linear program?

- simplex algorithm (exponential running time in the worst case, but almost linear in practice)
- ellipsoid method (poly-time in the worst case, but not so good in practice)

why worst-case complexity fails here to predict the behavior of these algorithms?

beyond worst case analysis

a good candidate for another advanced topic

Algorithm 14.1 (Set cover via LP-rounding)

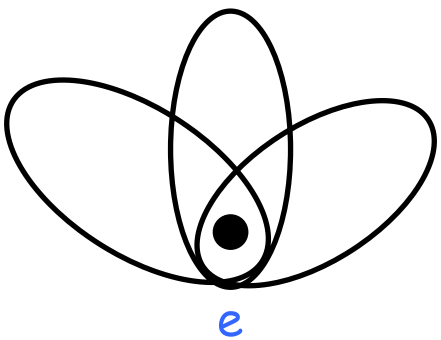
1. Find an optimal solution to the LP-relaxation.
2. Pick all sets S for which $x_S \geq 1/f$ in this solution.

Theorem

The above algorithm is an f-approximation algorithm for the SC problem.

proof

the solution computed is a feasible cover



pick an element e and consider the at most f sets containing e

since e is covered in the fractional solution there is at least a set S with $x_S \geq 1/f$

⇒ e is covered in the computed integer solution

the rounding process increases each x_S by a factor of at most f

cost of the computed cover $\leq f\, OPT_f \leq f\, OPT$

■

a special case: weighted Vertex Cover problem

Input:

- an undirected graph $G=(V,E)$
- each vertex v has a cost $c(v)$

Feasible solution:

$U \subseteq V$ such that every edge $(u,v) \in E$ is covered, i.e. $u \in U$ or $v \in U$

measure (min):

$$cost(U) : \sum_{v \in U} c(v)$$

f: frequency of the most frequent element =2

⇒ The LP-rounding algorithm is a 2-approximation algorithm for the weighted VC problem

tight example

view a set cover instance as a hypergraph:

- sets correspond to vertices
- elements correspond to hyperedges
- a set/vertex v covers an element/hyperedge $X \subseteq V$ if $v \in X$

Let $V_1, \ldots, V_k$ be k sets of cardinality n each. The hypergraph has:

- vertex set: $V = V_1 \cup \ldots \cup V_k$
- n^k hyperedges: each hyperedge picks one vertex from V_i
- all sets/vertices have cost 1

f=k

OPT_f= n (set each set/vertex variable to 1/k)

returned rounded solution has cost kn

OPT=n (pick V_1)

LP-duality

minimize $7x_1 + x_2 + 5x_3$

subject to

$$x_1 - x_2 + 3x_3 \geq 10$$

$$5x_1 + 2x_2 - x_3 \geq 6$$

$$x_1, x_2, x_3 \geq 0$$

z: value of the optimal solution

Is z at most α?

YES certificate: any feasible solution of value $\leq \alpha$

$\mathbf{x}=(2,1,3)$ feasible solution of value $7\cdot 2+1\cdot 1+5\cdot 3=30$

minimize $\quad 7x_1 + x_2 + 5x_3$

subject to

$$x_1 - x_2 + 3x_3 \geq 10$$

$$5x_1 + 2x_2 - x_3 \geq 6$$

$$x_1, x_2, x_3 \geq 0$$

z: value of the optimal solution

Is z at least α?

$$7x_1 + x_2 + 5x_3 \geq$$

$$x_1 - x_2 + 3x_3 \geq 10$$

$\Rightarrow$ $z \geq 10$

minimize $\quad 7x_1 + x_2 + 5x_3$

subject to $\quad x_1 - x_2 + 3x_3 \geq 10$

$$5x_1 + 2x_2 - x_3 \geq 6$$

$$x_1, x_2, x_3 \geq 0$$

z: value of the optimal solution

Is z at least α?

$$7x_1 + x_2 + 5x_3 \geq$$

$y_1 \quad x_1 - x_2 + 3x_3 \geq 10$

$+$

$y_2 \quad 5x_1 + 2x_2 - x_3 \geq 6$

$\Rightarrow \quad z \geq 16$

primal program

minimize $7x_1 + x_2 + 5x_3$

subject to

$$x_1 - x_2 + 3x_3 \geq 10$$
$$5x_1 + 2x_2 - x_3 \geq 6$$
$$x_1, x_2, x_3 \geq 0$$

dual program

maximize $10y_1 + 6y_2$

subject to

$$y_1 + 5y_2 \leq 7$$
$$-y_1 + 2y_2 \leq 1$$
$$3y_1 - y_2 \leq 5$$
$$y_1, y_2 \geq 0$$

every feasible solution of the dual gives a lower bound to the optimal solution of the primal

every feasible solution of the primal gives an upper bound to the optimal solution of the dual

two solutions with the same value must be both optimal!

optimal solutions:
$\mathbf{x}=(7/4,0,11/4)$ $\mathbf{y}=(2,1)$ both of value 26

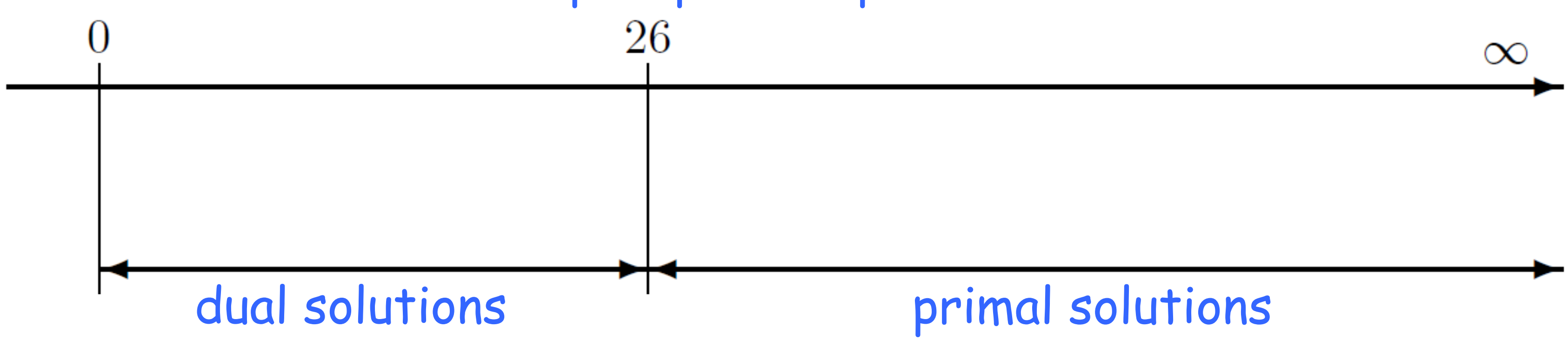
primal program

$$\begin{array}{ll} \text{minimize} & 7x_1 + x_2 + 5x_3 \\ \text{subject to} & x_1 - x_2 + 3x_3 \geq 10 \\ & 5x_1 + 2x_2 - x_3 \geq 6 \\ & x_1, x_2, x_3 \geq 0 \end{array}$$

dual program

$$\begin{array}{ll} \text{maximize} & 10y_1 + 6y_2 \\ \text{subject to} & y_1 + 5y_2 \leq 7 \\ & -y_1 + 2y_2 \leq 1 \\ & 3y_1 - y_2 \leq 5 \\ & y_1, y_2 \geq 0 \end{array}$$

dual opt = primal opt



primal program

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

subject to $\sum_{j=1}^{n} a_{ij} x_j \geq b_j \qquad i=1,\ldots,m$

$x_j \geq 0 \qquad j=1,\ldots,n$

dual program

maximize $\sum_{i=1}^{m} b_i y_i$

subject to $\sum_{i=1}^{m} a_{ij} y_i \leq c_j \qquad j=1,\ldots,n$

$y_i \geq 0 \qquad i=1,\ldots,m$

Theorem (LP-duality theorem)

The primal program has a finite optimum iff its dual has finite optimum. Moreover, if $\mathbf{x}=(x_1,\ldots,x_n)$ and $\mathbf{y}=(y_1,\ldots,y_m)$ are optimal solutions for the primal and dual programs, respectively, then

$$\sum_{j=1}^{n} c_j x_j = \sum_{i=1}^{m} b_i y_i$$

Theorem (weak duality theorem)

If $x=(x_1,\dots,x_n)$ and $y=(y_1,\dots,y_m)$ are feasible solutions for the primal and dual programs, respectively, then

$$\sum_{j=1}^{n} c_j x_j \ge \sum_{i=1}^{m} b_i y_i$$

proof

since for y is feasible and x_j's are nonnegative

$$\sum_{j=1}^{n} c_j x_j \ge \sum_{j=1}^{n} \left(\sum_{i=1}^{m} a_{ij} y_i \right) x_j =$$

$$= \sum_{i=1}^{m} \left(\sum_{j=1}^{n} a_{ij} x_j \right) y_i \ge \sum_{i=1}^{m} b_i y_i$$

since for x is feasible and y_i's are nonnegative

$\blacksquare$

Set Cover via dual-fitting

ILP:

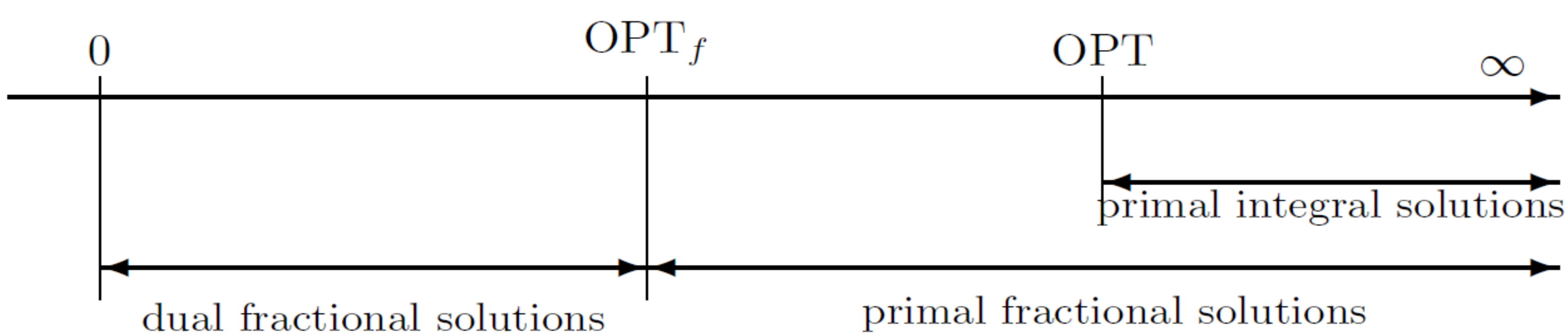
$$\begin{aligned} \text{minimize} \quad & \sum_{S \in \mathcal{S}} c(S) x_S & \\ \text{subject to} \quad & \sum_{S : e \in S} x_S \geq 1 & e \in U \\ & x_S \in \{0,1\} & S \in \mathcal{S} \end{aligned}$$

LP-relaxation

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

dual program

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



greedy strategy: pick the most cost-effective set and remove the covered elements, until all elements are covered.

Let C be the set of elements already covered.

cost-effectiveness of S: c(S)/|S-C|

average cost at which
S covers new elements

Algorithm 2.2 (Greedy set cover algorithm)

1. $C \leftarrow \emptyset$
2. While $C \neq U$ do

   Find the most cost-effective set in the current iteration, say S.

   Let $\alpha = \frac{\text{cost}(S)}{|S-C|}$, i.e., the cost-effectiveness of S.

   Pick S, and for each $e \in S - C$, set $\text{price}(e) = \alpha$.

   $C \leftarrow C \cup S$.
3. Output the picked sets.

average cost at which
e is covered

$$\text{for each } e \in U, \qquad y_e = \frac{\text{price}(e)}{H_n}$$

Lemma

The vector $\mathbf{y}$ defined above is a feasible solution for the dual program.

proof

We show that no set is *overpacked* in $\mathbf{y}$

consider a set S of k element, and list them in the order they are covered by the algorithm (break ties arbitrarily), say $e_1, \dots, e_k$.

When e_i is covered S contains at least $k-i+1$ uncovered elements

S can cover e_i at an average cost of at most $c(S)/(k-i+1)$

By the greedy choice, $\text{price}(e_i) \le c(S)/(k-i+1)$

Thus: $y_{e_i} \le \dfrac{1}{H_n} \dfrac{c(S)}{k-i+1}$

$$\sum_{i=1}^{k} y_{e_i} \le \frac{c(S)}{H_n}\left(\frac{1}{k} + \frac{1}{k-1} + \cdots + \frac{1}{1}\right) \le \frac{H_k}{H_n} c(S) \le c(S)$$

■

Theorem

The greedy algorithm is a H_n-approximation algorithm for the SC problem.

proof

$$\text{cost of the cover} = \sum_{e \in U} \text{price}(e) = H_n \sum_{e \in U} y_e \leq H_n \, OPT_f \leq H_n OPT$$

$\blacksquare$