Lec 20: Iterative Rounding

Using rank-based arguments to reason about LP solutions.

Sps. x^* is a vertex solution of $\min\{c^Tx \mid Ax \geq b\}$ where $x \in \mathbb{R}^n$

then $\exists$ n linearly independent constraints that are tight at x^*.

i.e. $\exists$ $a_{i_1}^T\ a_{i_2}^T \ldots a_{i_n}^T$ which are n rows of A

st these are L.I. and moreover $a_{i_j}^T x^* = b_i$

Use this info to argue what x^* looks like.

E.g. the Beck Fiala theorem.

Suppose we have n elements U and m sets over these. $(S_i \subseteq U)$

~~Each set~~ each element belongs to at most k ~~elements~~ sets.

Color the elements red/blue s.t. the discrepancy $(|\#\text{reds} - \#\text{blues}|)$ of each set is small.

Thm [Beck/Fiala]
$\exists$ coloring s.t. discrepancy $\leq 2k-1$

$$\left\{ \begin{array}{l} \chi: U \to \{-1,1\} \\ \text{st. } \mathrm{disc}(\chi) = \max_i |\chi(S_i)|. \end{array} \right\}$$

Pf: write LP relaxation.

Define a_i = incidence vector of set S_i $\quad a_i \in \{0,1\}^n \quad A = \begin{pmatrix} - a_1^T - \\ - a_2^T - \\ - a_m^T - \end{pmatrix}$

Would love to have $Ax = 0$. (i.e. zero discrepancy!) $\quad A \in \{0,1\}^{m \times n}$
$\cdot\ x \in \{-1,1\}$

LP relaxation: $Ax = 0$
$-1 \leq x_i \leq 1.\ \forall i$

n vars, $m+n$ constraints.

So consider basic solution x, ~~the~~ what are the n tight L.I constraints?

if any of them are -1 or 1, "fix" them.

F = fixed ~~constraints~~ variables. so $F \leftarrow \phi$ at start

M = constraints still in play $M \leftarrow [m]$.

Then $a_i^T x = 0 \quad \forall i \in M$

$-1 \le x_j \le 1 \quad \forall j \in [n] \setminus F$.

• Now if n tight LI constraints, if any $x_i = 1$ or -1, fix its value and add to F.

set $\hat{x}_j \leftarrow$ this integer value

Now get new system

$$\sum_{j \notin F} a_{ij} \cdot x_j = -\sum_{j \in F} a_{ij} \hat{x}_j \quad \forall i \in M.$$

$$-1 \le x_j \le 1 \quad \forall j \in [n] \setminus F.$$

Fewer variables so make progress.

• Sps none of these n tight LI constraints are $x_i = -1$ or $x_i = 1$.

b is this RHS, whatever it is.

then n tight constraints
$$\begin{aligned} a_1^T x &= b_1 \\ a_2^T x &= b_2 \quad \Leftrightarrow \quad A_{1:n} x = b. \\ &\vdots \\ a_n^T x &= b_n \end{aligned}$$

$A_{1:n}$ has at most k 1's per column (by set-system pty).

and is square matrix

$\Rightarrow \exists$ row with $\le k$ 1's. $\quad a_i^T x = b_i$

– For each such row, drop the constraint. remove from M.

Now fewer constraints. So make progress.

Note: if these x_i values don't change, have discrepancy 0.

But these values may later change by 2 (from, say, $-1+\varepsilon$ to $+1$)

so the discrepancy of these constraints $i \notin M$ may later become $2k-1$.

Open problem: get better than $2k - o(k)$

current best is $2k - \lg^* k$ by Boris Bukh.

Conjecture: $O(\sqrt{k})$?

—— x ——

This idea of using the fact that n-dim LPs must have n tight LI constraints, so that the rank of the tight submatrix s.t. $A_I x = b_I$ $|I| = n$. A_I is n.

These are called "rank" arguments

—— x ——

Rank arguments often give algorithms of the same type as above

— also called iterative rounding.

Bin packing with K types of items. $k = O(1)$?

items of size $s_1, s_2 \ldots s_k$; $s_i \in [0,1]$ (n_i copies of s_i)

Find the smallest # of bins of size 1 s.t. all items can be packed into them.

Pattern: configuration of items that fits into a bin.

$\mathcal{P}$ = set of patterns. say N patterns.

write pattern as $X_p = (a_1, a_2 \ldots a_k)$ $a_i \in \mathbb{Z}$.

copies of item $1, 2 \ldots k$ in p.

so $\sum a_i s_i \le 1$.

then want

$$\min \sum x_p$$
$$\sum_{p \in \mathcal{P}} x_p X_p \geq \begin{pmatrix} n_1 \\ n_2 \\ n_k \end{pmatrix}.$$
$$x_p \geq 0.$$

x_p = # ~~copies~~ bins of pattern p. in the solution.

- # constraints $= K + N$
- # variables $= N$.

$\Rightarrow \exists$ LI N tight constraints which are tight at optimal basic soln.

Which are these constraints? at most K are covering constraints, rest N-k must be $x_p \geq 0$

$\Rightarrow$ actually $\geq N-k$ of the variables are 0.

$\Rightarrow$ at most k fractional vars.

Round each x_i to $\lceil x_i \rceil$.

opens at most $LP + k$ bins in fact $\leq \lceil LP + (k-1) \rceil$ bins.

Q: how to solve this LP? Some other day... (use column generation).

but can indeed solve it, and get a basic solution

$\Rightarrow$ get an $OPT + (k-1)$ solution

—— x ——

Third Example: Unrelated machines Scheduling (again).

But now we'll add a cost. (so called Generalized Assignment Problem).

Say given a target T of load. And sizes p_{ij} for job j & m/c i

cost c_{ij}

$$\min \sum_{ij} c_{ij} x_{ij}$$

$$\text{st} \quad \sum_i x_{ij} = 1 \quad \forall j \in J$$

$$\sum_j x_{ij} p_{ij} \leq T \quad \text{for } i \in M$$

$$x_{ij} = 0 \text{ for } p_{ij} > T, \quad 1 \geq x_{ij} \geq 0. \ \forall \text{ all others.}$$

Say we have a set E of permissible edges. $ij \notin E$ if $p_{ij} > T$.

so. $\min c^T x$

$$x(\partial_j) = 1 \ \forall j \in J.$$

$$p_i^T x_i \leq T_i \ \forall i \in M. \qquad T_i = T \text{ initially } \forall i$$

$$x \in [0,1]^E.$$

$F \leftarrow \emptyset$ fixed assignments

→ if $\exists$ edge with $x_e = 0$ drop from E. $E \leftarrow E \setminus \{ij\}$.

else if $\exists$ edge with $x_{ij} = 1$ then "assign" j to m/c i, fix its value

so. $\begin{cases} F \leftarrow \{j \to i\} \cup F & \hat{x}_{ij} = 1. \\ \text{set } T_i \leftarrow T_i - p_{ij} & J \leftarrow J \setminus \{j\}. \end{cases}$

$E \leftarrow E \setminus \{ij\}$.

else if $\exists$ m/c with ≤ 2 edges in E adjacent to it,

remove the constraint from M, so $M \leftarrow M \setminus \{i\}$.

Resolve the LP, and go to → to find a basic sol^n.

Note: each round if $\exists\ x_e \in \{0,1\}$ or $\exists$ m/c of degree ≤ 2, we make progress.

Thm: sps we assign all jobs to machines then

load of machine $\leq$ ~~LP value~~ T + 2 extra jobs

until we drop the constraint, maintain $\leq T$.

b/c we dropped the constraint when it had ≤ 2 jobs incident to it.

$\Rightarrow$ load $\leq 3T$, 3 approx

[slightly worse than 2apx last time.
Can do better, but leave details to book Lau-Singh-Ravi]

Thm: Will always make progress. (so ~~always~~ basic soln always has either $x_e \in \{0,1\}$ or a m/c with degree ≤ 2.

Pf: Sps not. then have ~~\#~~ tight LI constraints given by $J' \subseteq J$ and $M' \subseteq M$. (none of $x_e \in [0,1]$ tight).

good.

Now :

$$|J'|+|M'| = |E| \geq \frac{\sum'_{j\in J} d_E(j) + \sum'_{i\in M} d_E(i)}{2} \geq |J|+|M| \geq |J'|+|M'|.$$

degree of j = # edges in E incident to j

b/c degrees ≥ 2

So all constraints must be tight. Means.

(a) All edges incident to m/c in M, and not on machines dropped from constraints

(b) all machines in M have degree $= 2$, ~~and~~ all jobs in J have degree 2,
~~(and that means there are no m/c in M, b/c we already said~~
~~all m/c in M have 2 degree~~

(c) $J' = J$, $M' = M$.

So the assignment is degree 2 on both sides. $d_E(j) = d_E(i) = 2 \;\; \forall\, i\in M,\, j\in J$

But we already said that $d_E(i) > 2$ for all $i \in M$

and we just deduced that jobs in j cannot go to machines outside M (by (a)).

$\Rightarrow$ contradiction

$\Rightarrow$ Always make progress, eventually assign job to m/cs with load $\leq 3T$

(And can do better by slightly more careful arguments).

— x —

Iterative rounding very powerful idea.

E.g. for Survivable Network Design, a result of Jain is extremely elegant 2apx. Shows $\exists$ edge that has value $\geq 1/2$, take it and recurse.

See Lau-Singh-Ravi book for many more elegant ideas.