EECS 495: Randomized Algorithms **Lecture 3**
Ordering Tournaments

Reading: Paper: *Aggregating Inconsistent Information: Ranking and Clustering*, Ailon, Charikar, Newman 2005.

Outline

Techniques:

- pivot algorithm
- charging arguments
- linear programing and duality

Application:

- feedback arc set

LP Duality

Recall linear programming:

$$\begin{array}{ll} \min & 7x_1 + x_2 + 5x_3 \\ \text{s.t.} & x_1 - x_2 + 3x_3 \geq 10 \\ & 5x_1 + 2x_2 - x_3 \geq 6 \\ & x_1, x_2, x_3 \geq 0 \end{array}$$

[[*We use LPs to lower-bound OPT. Hence sometimes useful to lower-bound the LP.*]]

Question: Can objective be negative?

Question: Can objective be ≤ 10?

$$7x_1 + x_2 + 5x_3 \geq x_1 - x_2 + 3x_3 \geq 10$$

Question: Better bound?

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

$$\geq (x_1 - x_2 + 3x_3) + (5x_1 + 2x_2 - x_3) \geq 16$$

Question: Better bound?

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

$$\geq 2(x_1 - x_2 + 3x_3) + (5x_1 + 2x_2 - x_3) \geq 26$$

Idea: Find non-neg multipliers for constraints so that

- When take sum, coeef of each x_i in sum is at most coeff of x_i in objective
- Sum of rhs is as large as possible

Formulate as the *dual LP*:

$$\begin{array}{ll} \max & 10y_1 + 6y_2 \\ \text{s.t.} & y_1 + 5y_2 \leq 7 \\ & -y_1 + 2y_2 \leq 1 \\ & 3y_1 - y_2 \leq 5 \\ & y_1, y_2 \geq 0 \end{array}$$

Note: Duality:

- Weak duality: Feasible solutions to dual give lower bound on primal and vice versa.
- Strong duality: Futhermore, opts are equal.

Example: For above primal/dual:

- feasible primal: $x = (7/4, 0, 11/4)$, value 26
- feasible dual: $y = (2, 1)$, value 26

So opt is 26.

Tournaments

Given:

- n teams V
- win/lose outcomes of all possible $\binom{n}{2}$ games

Output:

- global ranking of teams

Question: How to rank teams when outcomes are inconsistent?

Example: $A = \{(i, j), (j, k), (k, i)\}$

Goal: Find ranking that minimizes number of "mistakes".

Def: A *tournament* is a directed graph $G = (V, A)$ such that for every pair of vertices $i, j \in V$ either $(i, j) \in A$ or $(j, i) \in A$.

[[*Graph is directed so order matters in edge notation.*]]

Goal: Find a linear ordering of vertices that minimizes the number of back-edges.

Def: The above problem in general graphs is called *feedback arc set*; for tournaments we write *FAS-tournament.*

Note: FAS-tournament is NP-hard (see assigned paper).

Approximation

Algorithm: PageRank

Do a random walk with restarts on graph and rank according to time spent at each node.

Example: $\Omega(n)$-approx:

Draw: Vertices $\{1, \ldots, n\}$ perfectly ranked. Vertex n beats 1 and loses to everyone else. PageRank ranks n high but can get just one back-edge if we rank n low.

[[*What's going on? If you feel PageRank is the "right" algorithm, then our objective must be wrong. Important to define the right objective.*]]

Algorithm: FAS-PIVOT(G)

- Set $V_L \to \emptyset$, $V_R \to \emptyset$
- Pick random pivot $i \in V$
- For all vertices $j \in V \setminus \{i\}$:
  - If $(j, i) \in A$, then add j to V_L
  - Else add j to V_R
- Let G_L be tournament induced by V_L
- Let G_R be tournament induced by V_R
- Return order FAS-PIVOT(G_L), i, FAS-PIVOT(G_R)

Claim: Algorithm FAS-PIVOT is a randomized expected 3-approximation algorithm for FAS-TOURNAMENT.

Proof: Let C^{OPT} be cost of opt, C^{PIV} be cost of FAS-PIVOT. Want to show:

$$E[C^{PIV}] \leq 3C^{OPT}.$$

Idea: Count directed triangles (i, j, k).

STEP ONE: Bounding $E[C^{PIV}]$.

Edge (i, j) becomes backward iff

- $\exists\, k$ s.t. (i,j,k) form directed triangle and
- k was chosen as pivot when all three input to same recursive call
- Charge cost of edge (i,j) to triangle (i,j,k)

Draw picture.

- Let T denote set of directed triangles
- For $t \in T$, let A_t denote event one vertex of t is chosen as pivot when all three are part of same recursive call
- Let p_t be probability of A_t

Triangle charged exactly when A_t occurs, and can be charged at most once, so

$$E[C^{PIV}] = \sum_{t \in T} p_t.$$

STEP 2a: Bounding C^{OPT}, write as LP/dual.

Fact: For each *edge disjoint* triangle, OPT makes a mistake $\rightarrow$ cardinality of largest set of edge-disjoint triangles lower-bounds C^{OPT}.

Fact: Also true for a *fractional packing* of triangles, i.e., set of weights on triangles so that no edge carries more than one unit.

Formulate OPT as LP:

- variables x_e for each arc indicating if it is backward
- objective min $\sum_{e \in A} x_e$
- constraints $x_{e1} + x_{e2} + x_{e3} \geq 1$ for each triangle

Write dual:

- variables β_t for each triangle
- objective max $\sum_{t \in T} \beta_t$
- constraints $\sum_{t: e \in t} \beta_t \leq 1$

Dual is fractional packing of triangles.

STEP 2b: Bounding C^{OPT}, find feasible dual soln.

Idea: Use p_t to define feasible dual soln.

Let $t = (i,j,k)$ be some triangle.

- Conditioned on A_t, each one of 3 vertices of t was pivot with probability 1/3
- Thus any edge $e = (i,j)$ of t becomes a backward edge with probability 1/3 (conditioned on A_t)
- Let B_e be event e becomes backward edge. Then

$$\Pr[B_e \wedge A_t] = \Pr[B_e | A_t] \Pr[A_t] = \frac{1}{3} p_t$$

Note: For 2 diff triangles t and t' sharing edge e, events $B_e \wedge A_t$ and $B_e \wedge A_{t'}$ are disjoint!

$$\sum_{t: e \in t} \frac{1}{3} p_t \leq 1$$

(sum of prob. of disjoint events is at most one)

Thus, $C^{OPT} \geq \sum_t p_t/3 = E[C^{PIV}]/3$.