# 15-251: Great Theoretical Ideas In Computer Science

## Recitation 10 : Approximation Algorithms and Voting Rules

## Lecture Review

- The goal of an optimization problem is to find the minimum (or maximum) value under some constraints
- OPT($I$) is the value of the optimal solution to an instance $I$ of an optimization problem
- We say an algorithm $\mathcal{A}$ for an optimization problem is a factor-$\alpha$ approximation if for all instances $I$ of the problem $\mathcal{A}$ outputs a solution that is at least as good as $\alpha \cdot \text{OPT}(I)$.
- In an election we have $n$ *voters* who have each have a ranking list for the $m$ alternatives. A *preference profile* is the collective rankings of the alternatives by all the voters and a *voting rule* maps preference profiles to an alternative.
- The alternative outputted by a voting rule given a preference profile is the winner of the election
- A couple voting rules
  - **Plurality:** the alternative that is ranked first by the most voters wins
  - **Borda Count:** each voter awards $m - k$ points to their $k^{th}$ ranked alternative and the alternative with the most total points wins
- The $r$-*manipulation* problem asks if there is a way a voter can manipulate their preference list given the preference profile of the all the other voters (the non-manipulators) to force a preferred alternative $p$ win, under the voting rule $r$.

## Pokémon Coverage

Consider a set of Pokémon and a set of trainers each having a subset of these Pokémon. Given $k$ (assuming $k$ is less than the number of trainers), the problem is to maximize the number of distinct Pokémon covered. Prove that there exists a polynomial-time $(1 - 1/e)$-approximation algorithm for this problem by considering the following greedy algorithm and by using the following steps:

On input $S_1, \ldots S_m$ (each set correponds to the Pokémon that a given trainer has) and $k$ (the number of trainers chosen):

- Let $T = \emptyset$ (keeping track of trainers chosen)
- Let $U = \emptyset$ (keeping track of Pokémon covered)
- Repeat $k$ times:
  - Pick $j$ such that $j \notin T$ and $|S_j - U|$ is maximized.
  - Add $j$ to T.
  - Update $U$ to $U \cup S_j$.
- Output $T$.

(a) Show that the algorithm runs in polynomial time.

(b) Let $T^*$ denote the optimum solution, and let $U^* = \cup_{j \in T^*} S_j$. Note that the value of the optimum solution is $|U^*|$. Define $U_i$ to be set $U$ in the above algorithm after $i$ iterations of the loop. Let $r_i = |U^*| - |U_i|$. Prove that $r_i \leq (1 - \frac{1}{k})^i |U^*|$.

(c) Using the inequality $1 - \frac{1}{k} \leq e^{-\frac{1}{k}}$, conclude that the algorithm is a $(1 - \frac{1}{e})$-approximation algorithm for the problem.

## An Honest Algorithm for the Dishonest

Prove that the following greedy algorithm solves the Borda count-*manipulation* problem in polynomial time:

Given as input an election, a manipulator $x$, and a preferred candidate $p$:

- Rank $p$ in the first place for $x$.
- While there are unranked alternatives:
  - If there is an alternative that can be placed in the next spot without preventing $p$ from winning, place this alternative.
  - Otherwise, output False.
- Output True.