Lecture 22

Lecturer: Sitan Chen   *Scribe: Alex Lombardi*

1   Outline

In this lecture, we will cover the notion of *extension complexity* and its relationship to problems in nondeterministic communication complexity and hardness of approximation.

2   Context: Linear Programming

The general setup of a linear programming problem is to maximize some linear objective function $f(\underline{x}) = \underline{c}^\top \underline{x}$ over all $\underline{x} \in \mathbf{R}^n$ subject to a collection of linear constraints. The constraints can be expressed in the form

$$A\underline{x} \leq \underline{b},$$

where A is a $m \times n$ matrix, $\underline{x}$ is a vector of size n, and $\underline{b}$ is a vector of size m. Each of the m linear constraints that A imposes on $\underline{x}$ is called a *facet*, and requires that $\underline{x}$ lie in a corresponding closed halfspace in $\mathbf{R}^n$.

Our understanding of linear programs boils down to the following result:

Theorem (Fundamental Theorem of Linear Programming). *The objective function $f(\underline{x})$ of any linear program is maximized at one of the vertices of the convex polytope cut out by the constraints $A\underline{x} \leq \underline{b}$.*

This theorem tells us that LPs can be reformulated as the following problem: maximize a linear objective function $f(x) = \underline{c}^\top \underline{x}$ over a *discrete set* $X \subset \mathbf{R}^n$. In this framework, the associated polytope is the convex hull of the set X, denoted $\mathrm{Conv}(X)$.

Linear programming problems can be solved efficiently (polynomial time) in the "size of the program", i.e. the size of the constraint matrix A. In other words, we can efficiently maximize $\underline{c}^\top \underline{x}$ over $\underline{x}$ in any polytope with only $\mathrm{poly}(n)$ facets (as long as n is polynomial in the relevant input parameter).

Example (Maximum-weighted bipartite matching).

Suppose that we are given a weighted bipartite graph $G = (V_1, V_2, E, W)$, where E denotes the collection of edges between V_1 and V_2 and W denotes the corresponding weights of the edges. The problem is to find a matching with maximal total weight.

This can be reformulated as the following linear program: we have a variable x_e for each $e \in E$, where we think of $x_e = 0$ (the edge is not in the matching) or $x_e = 1$ (the edge is in the matching), although the linear program does not explicitly require that the x_e be integers. We then have the following constraints:

$$x_e \geq 0 \text{ for all } e \in E,$$

$$\sum_{e=(v,w)} x_e \leq 1 \text{ for all fixed } v \in V_1 \cup V_2.$$

Note that if all of the x_e are integers and satisfy the above constraints, then the assignment of $x_e \in \{0, 1\}$ corresponds to a matching. Finally, the objective function for this LP is given by

$$f(\underline{x}) = \sum_{e \in E} w_e x_e.$$

A priori, it is unclear that solutions to this LP correspond to matchings of maximum weight. However, the Birkoff-von Neumann theorem states that the polytope defined by the above constraints is the convex hull of points with integer coordinates (in particular, it says that any doubly stochastic matrix can be written as a convex combination of permutation matrices). Therefore, maximizing the objective function over the vertices of the polytope will in fact give us a matching with maximal weight. Since the number of variables and the number of constraints defining this LP are $O(n^2) = \text{poly}(n)$, we conclude that this problem can be solved efficiently by linear programming.

On the other hand, consider an NP-hard problem, such as max clique. It is possible to write down linear programs which solve the max clique problem; however, naive attempts to do this produce linear programs with exponentially many facets. In particular, the *clique polytope* in $\mathbf{R}^{n^2}$, defined to be the convex hull of all points $x \in \{0, 1\}^{n^2}$ that describe a set of edges on the vertex set $[n]$ producing a clique, can be shown to have $\Omega(2^{\binom{n}{2}})$ facets. Since max clique is NP-hard, we should expect any linear program solving this problem to have exponential size.

3   Extended Formulations

Before giving a formal definition of an extended formulation, we give an example.

Example (The Permutahedron).

Inner description: the permutahedron $P \subset \mathbf{R}^n$ is defined to be the convex hull of $\{(\pi(1), ..., \pi(n)), \pi \in S_n\}$, i.e. all permutations of $[n]$ thought of as vectors.

Outer description: define a variable x_i for each $i \in [n]$ (corresponding to the value of the ith coordinate of a vector in $\mathbf{R}^n$). Then, for every nonempty, proper subset $S \subset [n]$, we have the constraint

$$\sum_{i \in S} x_i \leq \sum_{k=n-|S|+1}^{n} k.$$

This gives us $2^n - 2$ facets defining P; at first glance, this seems to imply that linear programming on the permutahedron requires exponential time. However, it turns out that we can introduce extra variables to realize the permutahedron as the projection of a higher-dimensional polytope which has only polynomially many facets!

To this end, introduce variables $y_{i,j}$ for all pairs (i, j); intuitively, $y_{i,j}$ corresponds to the indicator function $\chi(\pi(i) == j)$. To match this intuition, we have the constraints

$$\sum_{j} y_{i,j} \leq 1 \text{ for all } i \in [n], \quad \sum_{i} y_{i,j} \leq 1 \text{ for all } j \in [n],$$

$$y_{i,j} \geq 0 \text{ for all } (i, j), \text{ and } x_i = \sum_{i=1}^{n} j y_{i,j} \text{ for all } i \in [n].$$

In particular, if the matrix $(y_{i,j})$ corresponds to a permutation matrix, then the vector (x_i) corresponds to a vertex in the permutahedron. Now, the Birkhoff-von Neumann theorem applies again: the constraints on the $y_{i,j}$ tell us that the matrix $Y = (y_{i,j})$ is a doubly stochastic matrix, hence a convex combination of permutation matrices. Thus, if we define the polytope $\tilde{P} \subset \mathbf{R}^{n^2+n}$ to be the polytope satisfying the constraints defined above, the projection $\text{proj}_x(\tilde{P})$ of $\tilde{P}$ onto its first n coordinates is exactly P. Furthermore, $\tilde{P}$ has only $n^2 + 3n$ facets defining it, so we can efficiently solve linear programming problems on $\tilde{P}$, which gives us a way to solve linear programming problems on P.

With this in mind, we now formally define extended formulation.

Definition (Extended Formulation). *An extended formulation of a polytope $P \subset \mathbf{R}^n$ defined by $A\underline{x} \leq \underline{b}$ is a polytope $\tilde{P} \subset \mathbf{R}^{n+r}$ defined by a system*

$$C\underline{x} + D\underline{y} \leq \underline{d}, \underline{x} \in \mathbf{R}^n, \underline{y} \in \mathbf{R}^r,$$

such that the projection $\mathrm{proj}_x(\tilde{P}) = P$.

Remark. *The geometric intuition for why this may be useful is the following: polyhedra often have fewer high-dimensional facets than lower dimensional facets. For example, the cube in $\mathbf{R}^3$ has only 6 faces, but 12 edges.*

Given an extended formulation $\tilde{P}$ of P, we define the size of $\tilde{P}$ to be the number of constraints defining $\tilde{P}$.

Definition. *The* extension complexity *$xc(P)$ of a polytope P is the minimum size of an extended formulation $\tilde{P}$ of P.*

Example. *If P denotes the permutahedron, we have that $xc(P) \leq n^2 + 3n$.*

As hinted at earlier, if $\mathrm{xc}(P)$ is polynomial in the relevant input parameter n, then maximizing linear functions over P can be done in polynomial time: run a polynomial time linear programming algorithm on $\tilde{P}$ and project the output back down to P. This fact provides a large source of bogus proofs that P = NP, by taking some NP-hard problem (expressed as an intractable linear programming problem) and claiming to have a polynomial size extended formulation of it. More interestingly, it is possible to rule out proofs of P = NP of this form, using communication complexity!

4 Yannakakis' Factorization Theorem

To relate extension complexity to communication complexity, we pass through the concept of *nonnegative rank*, which we define here.

Definition. *An $m \times n$ matrix M has nonnegative rank (denoted $\mathrm{rank}_+(M)$) at most r if it can be factored in the form $M = AB$ where A is a nonnegative $m \times r$ matrix and B is a nonnegative $r \times n$ matrix.*

Equivalently, we say that $\mathrm{rank}_+(M) \leq r$ if and only if $M = \sum_{i=1}^r M_i$ where each M_i is a nonnegative matrix with $\mathrm{rank}(M_i) = 1$.

We relate extension complexity to nonnegative rank in the following way: if P is a polytope with v vertices (denoted $\underline{x}_1, ..., \underline{x}_v$) and f facets, we define the *slack matrix* (a $v \times f$ matrix) associated to P to be

$$S_P = (\underline{b}_i - A_i \cdot \underline{v}_j)_{i,j}.$$

In other words, the (i,j) entry of S_P measures the slack of the ith inequality for the jth vertex. We then have the following result.

Theorem (Yannakakis). *$xc(P) = rank_+(S_P)$.*

Proof. (sketch) We only prove the inequality $\mathrm{xc}(p) \geq \mathrm{rank}_+(S_P)$, which is what we need to prove lower bounds on $\mathrm{xc}(P)$. Let $\tilde{P}$ denote an extended formulation of P with constraints $C\underline{x} + D\underline{y} \leq \underline{d}$. Since the inequalities $C\underline{x} + D\underline{y} \leq \underline{d}$ imply that $A\underline{x} \leq \underline{b}$ for all $x \in \mathbf{R}^n$, Farkas' lemma in convex geometry tells us that each of the m constraint vectors defining $A\underline{x} \leq \underline{b}$ can be written as a nonnegative linear combination of the constraint vectors defining $C\underline{x} + D\underline{y} \leq \underline{d}$. Therefore, if there are r constraints defining $\tilde{P}$, the slack matrix S_P can be written as a nonnegative linear combination of r (nonnegative) rank one matrices (corresponding to the constraints of $\tilde{P}$), proving that $\mathrm{rank}_+(S_P) \leq \mathrm{xc}(P)$. $\square$

5 Connection to Communication Complexity

Given our lower bound $\text{xc}(P) \geq \text{rank}_+(S_P)$, the connection to communication complexity becomes clearer. Recall the following result from earlier this semester: if the $|X| \times |Y|$ matrix M_f associated to a function $f : X \times Y \to \{0, 1\}$ can be covered by t monochromatic rectangles, then $\text{cc}(f) \leq \log_2(t)$. The upper bound $\log_2(t)$ also holds for the *nondeterministic communication complexity* $\text{ncc}(f)$, which is defined as follows. Alice and Bob have inputs x and y respectively, while a prover Carlos attempts to convince Alice and Bob that $f(x, y) = 1$. A successful communication protocol is one where if $f(x, y) = 1$, there exists some z such that Alice and Bob accept when Carlos sends them both z, while if $f(x, y) = 0$, no choice of z will convince Alice and Bob to accept. The nondeterministic communication complexity $\text{ncc}(f)$ is the minimum number of bits that Carlos must send in such a communication protocol.

Lemma. *If M_f can be covered by t monochromatic rectangles, then $\text{ncc}(f) \leq \log_2(t)$.*

Proof. If $M(f)$ can be covered by t monochromatic rectangles, then to convince Alice and Bob that $f(x, y) = 1$ (when Alice and Bob have inputs x and y), Carlos sends the rectangle $i \in [t]$ that (x, y) belongs to. Clearly such a communication requires $\log_2(t)$ bits. Alice and Bob can verify that $f(x, y) = 1$ given i by computing $f(x, y')$ (in Alice's case) or $f(x', y)$ (in Bob's case) for some (x, y') or (x', y) in the ith rectangle. If $f(x, y) = 0$ then the computation described in the previous sentence will lead Alice and Bob to both reject. Therefore, this is a valid communication protocol requiring $\log_2(t)$ bits of communication, proving that $\text{ncc}(f) \leq t$. $\square$

Using the above result as a prior that nondeterministic communication complexity can be related to some notion of "rank" of a matrix, we proceed to relate $\text{rank}_+(S_P)$ (for any polytope P) to the nondeterministic communication complexity of an associated function $f = \text{Face-Vertex}(P)$. The function $f : V(P) \times F(P) \to \{0, 1\}$ takes in a vertex of P and a facet of P, and outputs 0 on input v_i, f_j if and only if the inequality defined by f_j is an equality for the vertex v_i.

Claim. $\text{ncc}(\text{Face-Vertex(P)}) \leq \log_2(\text{rank}_+(S_P))$.

Proof. Let $\text{rank}_+(S_P) = r$, so that $S_p = \sum_{i=1}^{r} M_i$ with each M_i nonnegative and rank 1; for the purposes of a communication protocol, Alice and Bob can have access to this decomposition as well as factorizations $M_i = \underline{\alpha_i}(\underline{\beta_i})^\top$ for each i (this is the rank 1 property). As described above, the function $f(v_i, f_j) = 1$ if and only if the (j, i)-entry of S_P is nonzero, which is true if and only if the (j, i)-entry of M_k is nonzero for some $1 \leq k \leq r$. Moreover, the (j, i)-entry of $M_k = \underline{\alpha_k}(\underline{\beta_k})^\top$ is nonzero if and only if $(\alpha_k)_j (\beta_k)_i \neq 0$. Thus, to prove to Alice and Bob that $f(v_i, f_j) = 1$, Carlos sends k to Alice and Bob; Alice verifies that $(\beta_k)_i \neq 0$ and Bob verifies that $(\alpha_k)_j \neq 0$. If $f(v_i, f_j) = 0$, then one of those two verifications will fail, so at least one of Alice and Bob will reject. Thus, this is a valid communication protocol which requires that Carlos send $\log_2(r)$ bits, proving that $\text{ncc}(\text{Face-Vertex}(P)) \leq \log_2(\text{rank}_+(S_P))$. $\square$

To summarize, we have proved the inequality $\text{xc}(P) \geq \text{ncc}(\text{Face-Vertex}(P))$, so proving lower bounds on the extension complexity of a polytope has been reduced to lower bounds in communication complexity. This has led to (unconditional) lower bounds on the extension complexity of polytopes related to various NP-hard problems.

6 Generalization to Approximations, and Braverman-Moitra

Let $P \subset \mathbf{R}^n$ be a polytope, and consider the following generalization of extended formulation.

Definition. *A ρ-approximate extended formulation for P is an extended formulation $\tilde{P}$ of P such that for all linear objective functions w,*

$$\max[w^\top \underline{x}, \underline{x} \in P] \leq \max[w^\top \underline{x}, (\underline{x}, \underline{y}) \in \tilde{P}] \leq \rho \max[w^\top \underline{x}, \underline{x} \in P],$$

i.e. such that a ρ-approximate solution to any linear optimization problem on P can be obtained by solving the corresponding linear optimization problem on $\tilde{P}$.

The extension complexity $\mathrm{xc}_\rho(P)$ is then the minimal size of a ρ-approximate extended formulation of P.

Approximate extended formulations can be usefully rephrased in terms of extending a nested pair of polytopes $P \subset Q$. An extended formulation for (P, Q) is some polytope $\tilde{K}$ defined by the constraints $C\underline{x} + D\underline{y} \leq \underline{d}$, such that $\mathrm{proj}_x(\tilde{K}) = K$ for some $P \subset K \subset Q$. Denote by $\mathrm{xc}(P, Q)$ the minimum size of such a $\tilde{K}$. Then, it turns out that a ρ-approximate extended formulation of P is equivalent to an extended formulation of the pair $(P, \rho Q)$ where Q is a suitably defined auxiliary polytope related to P.

In the setting of a pair $P \subset Q$, one can define an analogous slack matrix $S_{P,Q}$ to be

$$S_{P,Q} = (\underline{b}_i - A_i \underline{v}_j)_{i,j},$$

where $(A, \underline{b})$ form the constraints defining Q and $\underline{v}_1, ..., \underline{v}_n$ are the collection of vertices that span P. We then have the following result generalizing Yannakakis' factorization theorem.

Theorem (Braun et al.). $\mathrm{xc}_\rho(P) = \mathrm{rank}_+(S_{P,\rho Q})$.

Finally, in some recent work, Braverman and Moitra proved optimal lower bounds for the extension complexity of approximating max clique. They accomplished this by phrasing the problem in terms of the extension complexity of (P, Q) for a suitably chosen Q, and obtaining lower bounds for $\mathrm{rank}_+(S_{P,Q})$ by a reduction to bounding the nondeterministic communication complexity of unique disjointness. In the end, they obtain the following result.

Theorem (Braverman-Moitra). *Obtaining a $n^{1-\epsilon}$-approximation of max clique has extension complexity $2^{\Omega(n^\epsilon)}$.*