CS369E: Communication Complexity (for Algorithm Designers) Lecture #5: Lower Bounds for the Extension Complexity of Polytopes*

Tim Roughgarden†

February 5 & 12, 2015

1 Linear Programs, Polytopes, and Extended Formulations

1.1 Linear Programs for Combinatorial Optimization Problems

You've probably seen some polynomial-time algorithms for the problem of computing a maximum-weight matching of a bipartite graph.[1] Many of these, like the Kuhn-Tucker algorithm [9], are "combinatorial algorithms" that operate directly on the graph.

Linear programming is also an effective tool for solving many discrete optimization problems. For example, consider the following linear programming relaxation of the maximum-weight bipartite matching problem (for a weighted bipartite graph $G = (U, V, E, w)$):

$$\max \sum_{e \in E} w_e x_e \tag{1}$$

subject to

$$\sum_{e \in \delta(v)} x_e \leq 1 \tag{2}$$

for every vertex $v \in U \cup V$ (where $\delta(v)$ denotes the edges incident to v) and

$$x_e \geq 0 \tag{3}$$

†Department of Computer Science, Stanford University, 474 Gates Building, 353 Serra Mall, Stanford, CA 94305. Email: tim@cs.stanford.edu.

[1]Recall that a graph is *bipartite* if its vertex set can be partitioned into two sets U and V such that every edge has one endpoint in each of U, V. Recall that a *matching* of a graph is a subset of edges that are pairwise disjoint.

for every edge $e \in E$.

In this formulation, each decision variable x_e is intended to encode whether an edge e is in the matching ($x_e = 1$) or not ($x_e = 0$). It is easy to verify that the vectors of $\{0,1\}^E$ that satisfy the constraints (2) and (3) are precisely the characteristic vectors of the matchings of G, with the objective function value of the solution to the linear program equal to the total weight of the matching.

Since every characteristic vector of a matching satisfies (2) and (3), and the set of feasible solutions to the linear system defined by (2) and (3) is convex, the convex hull of the characteristic vectors of matchings is contained in this feasible region.[2] Also note that every characteristic vector $\mathbf{x}$ of a matching is a vertex[3] of this feasible region — since all feasible solutions have all coordinates bounded by 0 and 1, the 0-1 vector $\mathbf{x}$ cannot be written as a non-trivial convex combination of other feasible solutions. The worry is does this feasible region contain anything other than the convex hull of characteristic vectors of matchings? Equivalently, does it have any vertices that are fractional, and hence do not correspond to matchings? (Note that integrality is not explicitly enforced by (2) or (3).)

A nice fact is that the vertices of the feasible region defined by (2) and (3) are precisely the characteristic vectors of matchings of G. This is equivalent to the Birkhoff-von Neumann theorem (see Exercises). There are algorithms that solve linear programs in polynomial time (and output a vertex of the feasible region, see e.g. [6]), so this implies that the maximum-weight bipartite matching problem can be solved efficiently using linear programming.

How about the more general problem of maximum-weight matching in general (non-bipartite) graphs? While the same linear system (2) and (3) still contains the convex hull of all characteristic vectors of matchings, and these characteristic vectors are vertices of the feasible region, there are also other, fractional, vertices. To see this, consider the simplest non-bipartite graph, a triangle. Every matching contains at most 1 edge. But assigning $x_e = \frac{1}{2}$ for each of the edges e yields a fractional solution that satisfies (2) and (3). This solution clearly cannot be written as a convex combination of characteristic vectors of matchings.

It is possible to add to (2)–(3) additional inequalities — "odd cycle inequalities" stating that, for every odd cycle C of G, $\sum_{e \in C} x_e \leq (|C| - 1)/2$ — so that the resulting smaller set of feasible solutions is precisely the convex hull of the characteristic vectors of matchings. Unfortunately, many graphs have an exponential number of odd cycles. Is it possible to add only a polynomial number of inequalities instead? Unfortunately not — the convex hull of the characteristic vectors of matchings can have $2^{\Omega(n)}$ "facets" [13].[4] We define facets

[2] Recall that a set $S \subseteq \mathbb{R}^n$ is *convex* if it is "filled in," with $\lambda \mathbf{x} + (1-\lambda)\mathbf{y} \in S$ whenever $\mathbf{x}, \mathbf{y} \in S$ and $\lambda \in [0,1]$. Recall that the *convex hull* of a point set $P \subseteq \mathbb{R}^n$ is the smallest (i.e., intersection of all) convex set that contains it. Equivalently, it is the set of all finite convex combinations of points of P, where a convex combination has the form $\sum_{i=1}^{p} \lambda_i \mathbf{x}_i$ for non-negative λ_i's summing to 1 and $\mathbf{x}_1, \ldots, \mathbf{x}_p \in P$.

[3] There is an unfortunate clash of terminology when talking about linear programming relaxations of combinatorial optimization problems: a "vertex" might refer to a node of a graph or to a "corner" of a geometric set.

[4] This linear programming formulation still leads to a polynomial-time algorithm, but using fairly heavy machinery — the "ellipsoid method" [8] and a "separation oracle" for the odd cycle inequalities [11]. There are also polynomial-time combinatorial algorithms for (weighted) non-bipartite matching, beginning with Edmonds [3].

more formally in Section 3.1, but intuitively they are the "sides" of a polytope,[5] like the $2n$ sides of an n-dimensional cube. It is intuitively clear that a polytope with ℓ facets needs ℓ inequalities to describe — it's like cleaving a shape out of marble, with each inequality contributing a single cut. We conclude that there is no linear program with variables $\{x_e\}_{e \in E}$ of polynomial size that captures the maximum-weight (non-bipartite) matching problem.

1.2 Auxiliary Variables and Extended Formulations

The exponential lower bound above on the number of linear inequalities needed to describe the convex hull of characteristic vectors of matchings of a non-bipartite graph applies to linear systems in $\mathbb{R}^E$, with one dimension per edge. The idea of an *extended formulation* is to add a polynomial number of auxiliary decision variables, with the hope that radically fewer inequalities are needed to describe the region of interest in the higher-dimensional space.

This idea might sound like grasping at straws, but sometimes it actually works. For example, fix a positive integer n, and represent a permutation $\pi \in S_n$ by the n-vector $\mathbf{x}_\pi = (\pi(1), \pi(2), \ldots, \pi(n))$, with all coordinates in $\{1, 2, \ldots, n\}$. The *permutahedron* is the convex hull of all $n!$ such vectors. The permutahedron is known to have $2^{n/2} - 2$ facets (see e.g. [5]), so a polynomial-sized linear description would seem out of reach.

Suppose we add n^2 auxiliary variables, y_{ij} for all $i, j \in \{1, 2, \ldots, n\}$. The intent is for y_{ij} to be a 0-1 variable that indicates whether or not $\pi(i) = j$ — in this case, the y_{ij}'s are the entries of the $n \times n$ permutation matrix that corresponds to π.

We next add a set of constraints to enforce the desired semantics of the y_{ij}'s (cf., (2) and (3)):

$$\sum_{j=1}^{n} y_{ij} \leq 1 \tag{4}$$

for $i = 1, 2, \ldots, n$;

$$\sum_{i=1}^{n} y_{ij} \leq 1 \tag{5}$$

for $j = 1, 2, \ldots, n$; and

$$y_{ij} \geq 0 \tag{6}$$

for all $i, j \in \{1, 2, \ldots, n\}$. We also add constraints that enforce consistency between the permutation encoded by the x_i's and by the y_{ij}'s:

$$x_i = \sum_{j=1}^{n} j y_{ij} \tag{7}$$

for all $i = 1, 2, \ldots, n$.

[5] A *polytope* is just a high-dimensional polygon — an intersection of halfspaces that is bounded or, equivalently, the convex hull of a finite set of points.

It is straightforward to check that the vectors $\mathbf{y} \in \{0,1\}^{n^2}$ that satisfy (4)–(6) are precisely the permutation matrices. For such a $\mathbf{y}$ corresponding to a permutation π, the constraints (7) force the x_i's to encode the same permutation π. Using again the Birkhoff-von Neumann Theorem, every vector $\mathbf{y} \in \mathbb{R}^{n^2}$ that satisfies (4)–(6) is a convex combination of permutation matrices (see Exercises). Constraint (7) implies that the x_i's encode the same convex combination of permutations. Thus, if we take the set of solutions in $\mathbb{R}^{n+n^2}$ that satisfy (4)–(7) and project onto the x-coordinates, we get exactly the permutahedron. This is what we mean by an *extended formulation* of a polytope.

To recap the remarkable trick we just pulled off: blowing up the number of variables from n to $n + n^2$ reduced the number of inequalities needed from $2^{n/2}$ to $n^2 + 3n$. This allows us to optimize a linear function over the permutahedron in polynomial time. Given a linear function (in the x_i's), we optimize it over the (polynomial-size) extended formulation, and retain only the x-variables of the optimal solution.

Given the utility of polynomial-size extended formulations, we'd obviously like to understand which problems have them. For example, does the non-bipartite matching problem admit such a formulation? The goal of this lecture is to develop communication complexity-based techniques for ruling out such polynomial-size extended formulations. We'll prove an impossibility result for the "correlation polytope" [4]; similar (but much more involved) arguments imply that every extended formulation of the non-bipartite matching problem requires an exponential number of inequalities [14].

Remark 1.1 (Geometric Intuition) It may seem surprising that adding a relatively small number of auxiliary variables can radically reduce the number of inequalities needed to describe a set — described in reverse, that projecting onto a subset of variables can massively blow up the number of sides. It's hard to draw (low-dimensional) pictures that illustrate this point. If you play around with projections of some three-dimensional polytopes onto the plane, you'll observe that non-facets of the high-dimensional polytope (edges) often become facets (again, edges) in the low-dimensional projection. Since the number of lower-dimensional faces of a polytope can be much bigger than the number of facets — already in the 3-D cube, there are 12 edges and only 6 sides — it should be plausible that a projection could significantly increase the number of facets.

2 Nondeterministic Communication Complexity

The connection between extended formulations of polytopes and communication complexity involves *nondeterministic* communication complexity. We studied this model implicitly in parts of Lecture #4; this section makes the model explicit.

Consider a function $f : X \times Y \to \{0,1\}$ and the corresponding 0-1 matrix $M(f)$, with rows indexed by Alice's possible inputs and columns indexed by Bob's possible inputs. In Lecture #4 we proved that if every covering of $M(f)$ by monochromatic rectangles[6] requires

[6]Recall that a *rectangle* is a subset $S \subseteq X \times Y$ that has a product structure, meaning $S = A \times B$ for some $A \subseteq X$ and $B \subseteq Y$. Equivalently, S is closed under "mix and match:" whenever $(\mathbf{x}_1, \mathbf{y}_1)$ and $(\mathbf{x}_2, \mathbf{y}_2)$

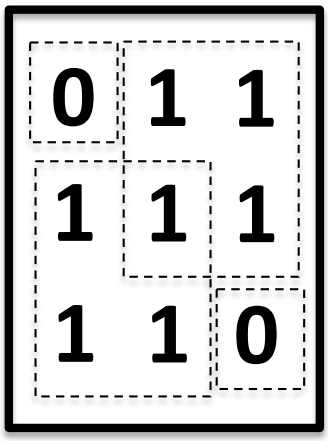

Figure 1: A covering by four monochromatic rectangles that is not a partition.

at least t rectangles, then the deterministic communication complexity of f is at least $\log_2 t$. The reason is that every communication protocol computing f with communication cost c induces a partition of $M(f)$ into at most 2^c monochromatic rectangles, and partitions are a special case of coverings. See also Figure 1.

Communication complexity lower bounds that are proved through coverings are actually much stronger than we've let on thus far — they apply also to *nondeterministic* protocols, which we define next.

You presumably already have a feel for nondeterminism from your study of the complexity class NP. Recall that one way to define NP is as the problems for which membership can be verified in polynomial time. To see how an analog might work with communication protocols, consider the complement of the EQUALITY problem, ¬EQUALITY. If a third party wanted to convince Alice and Bob that their inputs $\mathbf{x}$ and $\mathbf{y}$ are different, it would not be difficult: just specify an index $i \in \{1, 2, \ldots, n\}$ for which $x_i \neq y_i$. Specifying an index requires $\log_2 n$ bits, and specifying whether or not $x_i = 0$ and $y_i = 1$ or $x_i = 1$ and $y_i = 0$ requires one additional bit. Given such a specification, Alice and Bob can check the correctness of this "proof of non-equality" without any communication. If $\mathbf{x} \neq \mathbf{y}$, there is always a $(\log_2 +1)$-bit proof that will convince Alice and Bob of this fact; if $\mathbf{x} = \mathbf{y}$, then no such proof will convince Alice and Bob otherwise. This means that ¬EQUALITY has *nondeterministic communication complexity* at most $\log_2 n + 1$.

Coverings of $M(f)$ by monochromatic rectangles are closely related to the nondeterministic communication complexity of f. We first show how coverings lead to nondeterministic protocols. It's easiest to formally define such protocols after the proof.

Proposition 2.1 *Let $f : X \times Y \to \{0, 1\}$ be a Boolean function and $M(f)$ the correspond-*

are in S, so are $(\mathbf{x}_1, \mathbf{y}_2)$ and $(\mathbf{x}_2, \mathbf{y}_1)$. A rectangle is *monochromatic* (w.r.t. f) if it contains only 1-entries of $M(f)$ or only 0-entries of $M(f)$. In these cases, we call it a *1-rectangle* or a *0-rectangle*, respectively.

ing matrix. If there is a cover of the 1-entries of $M(f)$ by t 1-rectangles, then there is a nondeterministic protocol that verifies $f(\mathbf{x}, \mathbf{y}) = 1$ with cost $\log_2 t$.

Proof: Let $R_1, \ldots, R_t$ denote a covering of the 1s of $M(f)$ by 1-rectangles. Alice and Bob can agree to this covering in advance of receiving their inputs. Now consider the following scenario:

1. A *prover* — a third party — sees both inputs $\mathbf{x}$ *and* $\mathbf{y}$. (This is the formal model used for nondeterministic protocols.)

2. The prover writes an index $i \in \{1, 2, \ldots, t\}$ — the name of a rectangle R_i — on a blackboard, in public view. Since R_i is a rectangle, it can be written as $R_i = A_i \times B_i$ with $A_i \subseteq X$, $B_i \subseteq Y$.

3. Alice accepts if and only if $\mathbf{x} \in A_i$.

4. Bob accepts if and only if $\mathbf{y} \in B_i$.

This protocol has the following properties:

1. If $f(\mathbf{x}, \mathbf{y}) = 1$, then there exists a proof such that Alice and Bob both accept. (Since $f(\mathbf{x}, \mathbf{y}) = 1$, $(\mathbf{x}, \mathbf{y}) \in R_i$ for some i, and Alice and Bob both accept if "i" is written on the blackboard.)

2. If $f(\mathbf{x}, \mathbf{y}) = 0$, there is no proof that both Alice and Bob accept. (Whatever index $i \in \{1, 2, \ldots, t\}$ is written on the blackboard, since $f(\mathbf{x}, \mathbf{y}) = 0$, either $\mathbf{x} \notin R_i$ or $y \notin R_i$, causing a rejection.)

3. The maximum length of a proof is $\log_2 t$. (A proof is just an index $i \in \{1, 2, \ldots, t\}$.)

These three properties imply, by definition, that the nondeterministic communication complexity of the function f and the output 1 is at most $\log_2 t$. ■

The proof of Proposition 2.1 introduces our formal model of nondeterministic communication complexity: Alice and Bob are given a "proof" or "advice string" by a prover, which can depend on both of their inputs; the communication cost is the worst-case length of the proof; and a protocol is said to compute an output $z \in \{0, 1\}$ of a function f if $f(\mathbf{x}, \mathbf{y}) = z$ if and only if there exists proof such that both Alice and Bob accept.

With nondeterministic communication complexity, we speak about both a function f and an output $z \in \{0, 1\}$. For example, if f is EQUALITY, then we saw that the nondeterministic communication complexity of f and the output 0 is at most $\log_2 n + 1$. Since it's not clear how to convince Alice and Bob that their inputs *are* equal without specifying at least one bit for each of the n coordinates, one might expect the nondeterministic communication complexity of f and the output 1 to be roughly n. (And it is, as we'll see.)

We've defined nondeterministic protocols so that Alice and Bob never speak, and only verify. This is without loss of generality, since given a protocol in which they do speak,

one could modify it so that the prover writes on the blackboard everything that they would have said. We encourage the reader to formalize an alternative definition of nondeterministic protocols without a prover and in which Alice and Bob speak nondeterministically, and to prove that this definition is equivalent to the one we've given above (see Exercises).

Next we prove the converse of Proposition 2.2.

Proposition 2.2 *If the nondeterministic communication complexity of the function f and the output 1 is c, then there is a covering of the 1s of $M(f)$ by 2^c 1-rectangles.*

Proof: Let $\mathcal{P}$ denote a nondeterministic communication protocol for f and the output 1 with communication cost (i.e., maximum proof length) at most c. For a proof ℓ, let $Z(\ell)$ denote the inputs $(\mathbf{x}, \mathbf{y})$ where both Alice and Bob accept the proof. We can write $Z(\ell) = A \times B$, where A is the set of inputs $\mathbf{x} \in X$ of Alice where she accepts the proof ℓ, and B is the set of inputs $\mathbf{y} \in Y$ of Bob where he accepts the proof. By the assumed correctness of $\mathcal{P}$, $f(\mathbf{x}, \mathbf{y}) = 1$ for every $(\mathbf{x}, \mathbf{y}) \in Z(\ell)$. That is, $Z(\ell)$ is a 1-rectangle.

By the first property of nondeterministic protocols, for every 1-input $(\mathbf{x}, \mathbf{y})$ there is a proof such that both Alice and Bob accept. That is, $\cup_\ell Z(\ell)$ is precisely the set of 1-inputs of f — a covering of the 1s of $M(f)$ by 1-rectangles. Since the communication cost of $\mathcal{P}$ is at most c, there are at most 2^c different proofs ℓ. ■

Proposition 2.2 implies that communication complexity lower bounds derived from covering lower bounds apply to nondeterministic protocols.

Corollary 2.3 *If every covering of the 1s of $M(f)$ by 1-rectangles uses at least t rectangles, then the nondeterministic communication complexity of f is at least $\log_2 t$.*

Thus our arguments in Lecture #4, while simple, were even more powerful than we realized — they prove that the nondeterministic communication complexity of Equality, Disjointness, and Greater-Than (all with output 1) is at least n. It's kind of amazing that these lower bounds can be proved with so little work.

3 Extended Formulations and Nondeterministic Communication Complexity

What does communication complexity have to do with extended formulations? To forge a connection, we need to show that an extended formulation with few inequalities is somehow useful for solving hard communication problems. While this course includes a number of clever connections between communication complexity and various computational models, this connection to extended formulations is perhaps the most surprising and ingenious one of them all. Superficially, extended formulations with few inequalities can be thought of as "compressed descriptions" of a polytope, and communication complexity is generally useful for ruling out compressed descriptions of various types. It is not at all obvious that this vague intuition can be turned into a formal connection, let alone one that is useful for proving non-trivial impossibility results.

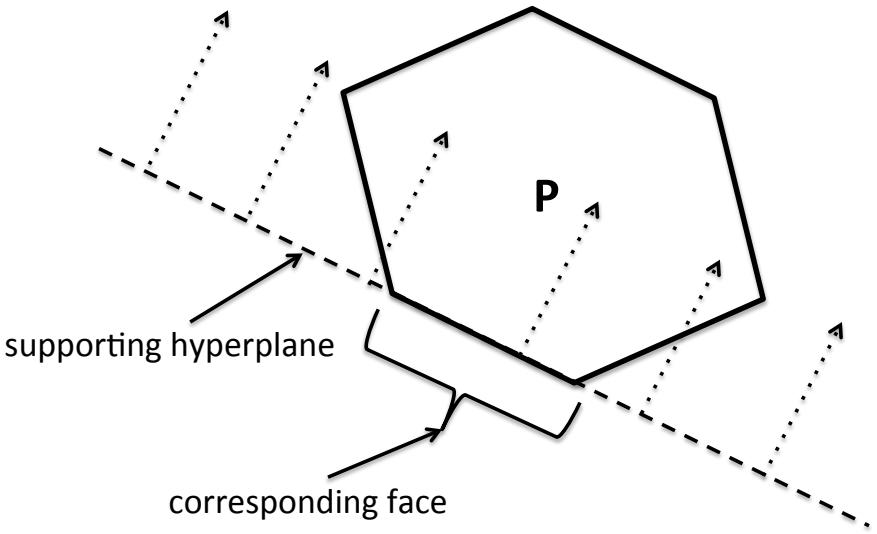

Figure 2: A supporting hyperplane of a polytope P and the corresponding face of the polytope.

3.1 Faces and Facets

We discuss briefly some preliminaries about polytopes. Let P be a polytope in variables $\mathbf{x} \in \mathbb{R}^n$. By definition, an *extended formulation* of P is a set of the form

$$Q = \{(\mathbf{x}, \mathbf{y}) : \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{y} \leq \mathbf{d}\},$$

where $\mathbf{x}$ and $\mathbf{y}$ are the original and auxiliary variables, respectively, such that

$$\{\mathbf{x} : \exists \mathbf{y} \text{ s.t. } (\mathbf{x}, \mathbf{y}) \in Q\} = P.$$

This is, projecting Q onto the original variables $\mathbf{x}$ yields the original polytope P. The extended formulation of the permutahedron described in Section 1.2 is a canonical example. The *size* of an extended formulation is the number of inequalities.[7]

Recall that $\mathbf{x} \in P$ is a *vertex* if it cannot be written as a non-trivial convex combination of other points in P. A *supporting hyperplane of* P is a vector $\mathbf{a} \in \mathbb{R}^n$ and scalar $b \in \mathbb{R}$ such that $\mathbf{a}\mathbf{x} = b$ for all $\mathbf{x} \in P$. Every supporting hyperplane $\mathbf{a}, b$ induces a *face* of P, defined as $\{\mathbf{x} \in P : \mathbf{a}\mathbf{x} = b\}$ — the intersection of the boundaries of P and of the the halfspace defined by the supporting hyperplane. (See Figure 2.) Note that a face is generally induced by many different supporting hyperplanes. The empty set is considered a face. Note also that faces are nested — in three dimensions, there are vertices, edges, and sides. In general, if f is a face of P, then the vertices of f are precisely the vertices of P that are contained in f.

[7]There is no need to keep track of the number of auxiliary variables — there is no point in having an extended formulation of this type with more variables than inequalities (see Exercises).

A *facet* of P is a maximal face — a face that is not strictly contained in any other face. Provided P has a non-empty interior, its facets are $(n-1)$-dimensional.

There are two different types of finite descriptions of a polytope, and it is useful to go back and forth between them. First, a polytope P equals the convex hull of its vertices. Second, P is the intersection of the halfspaces that define its facets.[8]

3.2 Yannakakis's Lemma

What good is a small extended formulation? We next make up a contrived communication problem for which small extended formulations are useful. For a polytope P, in the corresponding Face-Vertex(P) problem, Alice gets a face f of P (in the form of a supporting hyperplane $\mathbf{a}, b$) and Bob gets a vertex v of P. The function $FV(f, v)$ is defined as 1 if v does *not* belong to f, and 0 if $v \in f$. Equivalently, $FV(f, v) = 1$ if and only if $\mathbf{a^T v} < b$, where $\mathbf{a}, b$ is a supporting hyperplane that induces f. Polytopes in n dimensions generally have an exponential number of faces and vertices. Thus, trivial protocols for Face-Vertex(P), where one party reports their input to the other, can have communication cost $\Omega(n)$.

A key result is the following.

Lemma 3.1 (Yannakakis's Lemma [15]) *If the polytope P admits an extended formulation Q with r inequalities, then the nondeterministic communication complexity of* Face-Vertex(P) *is at most* $\log_2 r$.

That is, if we can prove a linear lower bound on the nondeterministic communication complexity of the Face-Vertex(P) problem, then we have ruled out subexponential-size extended formulations of P.

Sections 3.3 and 3.4 give two different proof sketches of Lemma 3.1. These are roughly equivalent, with the first emphasizing the geometric aspects (following [10]) and the second the algebraic aspects (following [15]). In Section 4 we put Lemma 3.1 to use and prove strong lower bounds for a concrete polytope.

Remarkably, Yannakakis [15] did not give any applications of his lemma — the lower bounds for extended formulations in [15] are for "symmetric" formulations and proved via direct arguments. Lemma 3.1 was suggested in [15] as a potentially useful tool for more general impossibility results, and finally in the past five years (beginning with [4]) this prophecy has come to pass.

3.3 Proof Sketch of Lemma 3.1: A Geometric Argument

Suppose P admits an extended formulation $Q = \{(\mathbf{x}, \mathbf{y}) : \mathbf{Cx} + \mathbf{Dy} \leq \mathbf{d}\}$ with only r inequalities. Both P and Q are known to Alice and Bob before the protocol begins. A first idea is for Alice, who is given a face f of the original polytope P, to tell Bob the name of the

[8] Proofs of all of these statements are elementary but outside the scope of this lecture; see e.g. [16] for details.

"corresponding face" of Q. Bob can then check whether or not his "corresponding vertex" belongs to the named face or not, thereby computing the function.

Unfortunately, knowing that Q is defined by r inequalities only implies that it has at most r *facets* — it can have a very large number of faces. Thus Alice can no more afford to write down an arbitrary face of Q than a face of P.

We use a third-party prover to name a suitable facet of Q than enables Alice and Bob to compute the FACE-VERTEX(P) function; since Q has at most r facets, the protocol's communication cost is only $\log_2 r$, as desired.

Suppose the prover wants to convince Alice and Bob that Bob's vertex v of P does not belong to Alice's face f of P. If the prover can name a facet f^* of Q such that:

(i) there exists $\mathbf{y}_v$ such that $(v, \mathbf{y}_v) \notin f^*$; and

(ii) for every $(\mathbf{x}, \mathbf{y}) \in Q$ with $\mathbf{x} \in f$, $(\mathbf{x}, \mathbf{y}) \in f^*$;

then this facet f^* proves that $v \notin f$. Moreover, given f^*, Alice and Bob can verify (ii) and (i), respectively, without any communication.

All that remains to prove is that, when $v \notin f$, there exists a facet f^* of Q such that (i) and (ii) hold. First consider the inverse image of f in Q, $\tilde{f} = \{(\mathbf{x}, \mathbf{y}) \in Q : \mathbf{x} \in f\}$. Similarly, define $\tilde{v} = \{(v, \mathbf{y}) \in Q\}$. Since $v \notin f$, $\tilde{f}$ and $\tilde{v}$ are disjoint subsets of Q. It is not difficult to prove that $\tilde{f}$ and $\tilde{v}$, as inverse images of faces under a linear map, are faces of Q (exercise). An intuitive but non-trivial fact is that every face of a polytope is the intersection of the facets that contain it.[9] Thus, for every vertex v^* of Q that is contained in $\tilde{v}$ (and hence not in $\tilde{f}$) — and since $\tilde{v}$ is non-empty, there is at least one — we can choose a facet f^* of Q that contains $\tilde{f}$ (property (ii)) but excludes v^* (property (i)). This concludes the proof sketch of Lemma 3.1.

3.4 Proof Sketch of Lemma 3.1: An Algebraic Argument

The next proof sketch of Lemma 3.1 is a bit longer but introduces some of the most important concepts in the study of extended formulations.

The *slack matrix* of a polytope P has rows indexed by faces F and columns indexed by vertices V. We identify each face with a canonical supporting hyperplane $\mathbf{a}, b$. Entry S_{fv} of the slack matrix is defined as $b - \mathbf{a}^{\mathbf{T}}\mathbf{v}$, where $\mathbf{a}, b$ is the supporting hyperplane corresponding to the face f. Observe that all entries of S are nonnegative. Define the *support* $\text{supp}(S)$ of the slack matrix S as the $F \times V$ matrix with 1-entries wherever S has positive entries, and 0-entries wherever S has 0-entries. Observe that $\text{supp}(S)$ is a property only of the polytope P, independent of the choices of the supporting hyperplanes for the faces of P. Observe also that $\text{supp}(S)$ is precisely the answer matrix for the FACE-VERTEX(P) problem for the polytope P.

We next identify a sufficient condition for FACE-VERTEX(P) to have low nondeterministic communication complexity; later we explain why the existence of a small extended

[9] This follows from Farkas's Lemma, or equivalently the Separating Hyperplane Theorem. See [16] for details.

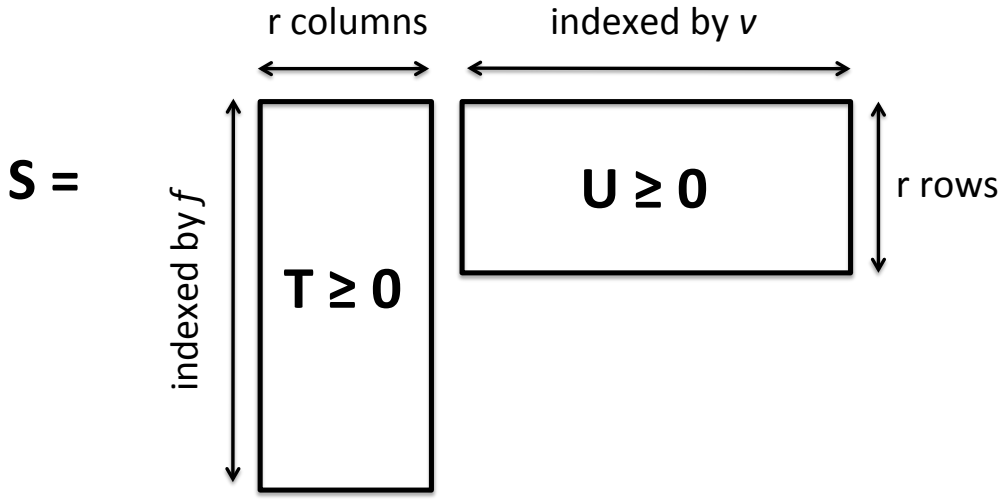

Figure 3: A rank-r factorization of the slack matrix S into nonnegative matrices T and U.

formulation implies this sufficient condition. Suppose the slack matrix S has *nonnegative rank* r, meaning it is possible to write $S = TU$ with T a $|F| \times r$ nonnegative matrix and U a $r \times |V|$ nonnegative matrix (Figure 3).[10] Equivalently, suppose we can write S as the sum of r outer products of nonnegative vectors (indexed by F and V):

$$S = \sum_{j=1}^{r} \alpha_j \cdot \beta_j^T, \tag{8}$$

where the α_j's correspond to the columns of T and the β_j's to the rows of U.

We claim that if the slack matrix S of a polytope P has nonnegative rank r, then there is a nondeterministic communication protocol for Face-Vertex(P) with cost at most $\log_2 r$. As usual, Alice and Bob can agree to the decomposition (8) in advance. A key observation is that, by inspection of (8), $S_{fv} > 0$ if and only if there exists some $j \in \{1, 2, \ldots, r\}$ with $\alpha_{fj}, \beta_{jv} > 0$. (We are using here that everything is nonnegative and so no cancellations are possible.) Equivalently, the supports of the outer products $\alpha_j \cdot \beta_j^T$ can be viewed as a covering of the 1-entries of supp(S) by r 1-rectangles. Given this observation, the protocol for Face-Vertex(P) should be clear.

1. The prover announces an index $j \in \{1, 2, \ldots, r\}$.

2. Alice accepts if and only if the fth component of α_j is strictly positive.

3. Bob accepts if and only if the vth component of β_j is strictly positive.

[10] This is called a *nonnegative matrix factorization*. It is the analog of the singular value decomposition (SVD), but with the extra constraint that the factors are nonnegative matrices. It obviously only makes sense to ask for such decompositions for nonnegative matrices (like S).

The communication cost of the protocol is clearly $\log_2 r$. The key observation above implies that there is a proof (i.e., an index $j \in \{1, 2, \ldots, r\}$) accepted by both Alice and Bob if and only if Bob's vertex v does not belong to Alice's face f.

It remains to prove that, whenever a polytope P admits an extended formulation with a small number of inequalities, its slack matrix admits a low-rank nonnegative matrix factorization.[11] We'll show this by exhibiting nonnegative r-vectors λ_f (for all faces f of P) and μ_v (for all vertices v of P) such that $S_{fv} = \lambda_f^T \mu_v$ for all f and v. In terms of Figure 3, the λ_f's and μ_v's correspond to the rows of T and columns of U, respectively.

The next task is to understand better how an extended formulation $Q = \{(\mathbf{x}, \mathbf{y}) : \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{y} \leq \mathbf{d}\}$ must be related to the original polytope P. Given that projecting Q onto the variables $\mathbf{x}$ yields P, it must be that every supporting hyperplane of P is logically implied by the inequalities that define Q. To see one way how this can happen, suppose there is a non-negative r-vector $\lambda \in \mathbb{R}^r_+$ with the following properties:

(P1) $\lambda^T \mathbf{C} = \mathbf{a}^T$;

(P2) $\lambda^T \mathbf{D} = \mathbf{0}$;

(P3) $\lambda^T \mathbf{d} = b$.

(P1)–(P3) imply that, for every $(\mathbf{x}, \mathbf{y})$ in Q (and so with $\mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{y} \leq \mathbf{d}$), we have

$$\underbrace{\lambda^T \mathbf{C}}_{=\mathbf{a}^T} \mathbf{x} + \underbrace{\lambda^T \mathbf{D}}_{=\mathbf{0}} \mathbf{y} \leq \underbrace{\lambda^T \mathbf{d}}_{=b}$$

and hence $\mathbf{a}^\mathbf{T}\mathbf{x} \leq b$ (no matter what $\mathbf{y}$ is).

Nonnegative linear combinations λ of the constraints of Q that satisfy (P1)–(P3) are one way in which the constraints of Q imply constraints on the values of $\mathbf{x}$ in the projection of Q. A straightforward application of Farkas's Lemma (see e.g. [1]) implies that such nonnegative linear combinations are the *only way* in which the constraints of Q imply constraints on the projection of Q.[12] Put differently, whenever $\mathbf{a}^\mathbf{T}\mathbf{x} \leq b$ is a supporting hyperplane of P, there exists a nonnegative linear combination λ that proves it (i.e., that satisfies (P1)–(P3)). This clarifies what the extended formulation Q really accomplishes: ranging over all $\lambda \in \mathbb{R}^r_+$ satisfying (P2) generates all of the supporting hyperplanes $\mathbf{a}, b$ of P (with $\mathbf{a}$ and b arising as $\lambda^T \mathbf{C}$ and $\lambda^T \mathbf{d}$, respectively).

To define the promised λ_f's and μ_v's, fix a face f of P with supporting hyperplane $\mathbf{a}^\mathbf{T}\mathbf{x} \leq b$. Since Q's projection does not include any points not in P, the constraints of Q imply this supporting hyperplane. By the previous paragraph, we can choose a nonnegative vector λ_f so that (P1)–(P3) hold.

[11] The converse also holds, and might well be the easier direction to anticipate. See the Exercises for details.

[12] Farkas's Lemma is sometimes phrased as the Separating Hyperplane Theorem. It can also be thought of as the feasibility version of strong linear programming duality.

Now fix a vertex v of P. Since Q's projection includes every point of P, there exists a choice of $\mathbf{y}_v$ such that $(v, \mathbf{y}_v) \in Q$. Define $\mu_v \in \mathbb{R}^r_+$ as the slack in Q's constraints at the point $(v, \mathbf{y}_v)$:

$$\mu_v = \mathbf{d} - \mathbf{C}\mathbf{v} - \mathbf{D}\mathbf{y}_v.$$

Since $(v, \mathbf{y}_v) \in Q$, μ_v is a nonnegative vector.

Finally, for every face f of P and vertex v of P, we have

$$\lambda_f^T \mu_v = \underbrace{\lambda_f^T \mathbf{d}}_{=b} - \underbrace{\lambda_f^T \mathbf{C}\mathbf{v}}_{=\mathbf{a}^T v} - \underbrace{\lambda_f^T \mathbf{D}\mathbf{y}_v}_{=0} = b - \mathbf{a}^T\mathbf{v} = S_{fv},$$

as desired. This completes the second proof of Lemma 3.1.

4 A Lower Bound for the Correlation Polytope

4.1 Overview

Lemma 3.1 reduces the task of proving lower bounds on the size of extended formulations of a polytope P to proving lower bounds on the nondeterministic communication complexity of Face-Vertex(P). The case study of the permutahedron (Section 1.2) serves as a cautionary tale here: the communication complexity of Face-Vertex(P) is surprisingly low for some complex-seeming polytopes, so proving strong lower bounds, when they exist, typically requires work and a detailed understanding of the particular polytope of interest.

Fiorini et al. [4] were the first to use Yannakakis's Lemma to prove lower bounds on the size of extended formulations of interesting polytopes.[13] We follow the proof plan of [4], which has two steps.

1. First, we exhibit a polytope that is tailor-made for proving a nondeterministic communication complexity lower bound on the corresponding Face-Vertex(P) problem, via a reduction from Disjointness. We'll prove this step in full.

2. Second, we extend the consequent lower bound on the size of extended formulations to other problems, such as the Traveling Salesman Problem (TSP), via reductions. These reductions are bread-and-butter NP-completeness-style reductions; see the Exercises for more details.

This two-step plan does not seem sufficient to resolve the motivating problem mentioned in Section 1, the non-bipartite matching problem. For an NP-hard problem like TSP, we fully expect all extended formulations of the convex hull of the characteristic vectors of solutions to be exponential; otherwise, we could use linear programming to obtain a subexponential-time algorithm for the problem (an unlikely result). The non-bipartite matching problem is polynomial-time solvable, so it's less clear what to expect. Rothvoss [14] proved that every extended formulation of the convex hull of the perfect matchings of the complete graph

[13] This paper won the Best Paper Award at STOC '12.

has exponential size.[14] The techniques in [14] are more sophisticated variations of the tools covered in this lecture — a reader of these notes is well-positioned to move on to the proof in [14].

4.2 Preliminaries

We describe a polytope P for which it's relatively easy to prove nondeterministic communication complexity lower bounds for the corresponding Face-Vertex(P) problem. The polytope was studied earlier for other reasons [12, 2].

Given a 0-1 n-bit vector $\mathbf{x}$, we consider the corresponding (symmetric and rank-1) outer product $\mathbf{x}\mathbf{x}^T$. For example, if $\mathbf{x} = 10101$, then

$$\mathbf{x}\mathbf{x}^T = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

For a positive integer n, we define COR as the convex hull of all 2^n such vectors $\mathbf{x}\mathbf{x}^T$ (ranging over $\mathbf{x} \in \{0,1\}^n$). This is a polytope in $\mathbb{R}^{n^2}$, and its vertices are precisely the points $\mathbf{x}\mathbf{x}^T$ with $\mathbf{x} \in \{0,1\}^n$.

Our goal is to prove the following result.

Theorem 4.1 ([4]) *The nondeterministic communication complexity of* Face-Vertex(COR) *is* $\Omega(n)$.

This lower bound is clearly the best possible (up to a constant factor), since Bob can communicate his vertex to Alice using only n bits (by specifying the appropriate $\mathbf{x} \in \{0,1\}^n$).

Lemma 3.1 then implies that every extended formulation of the COR polytope requires $2^{\Omega(n)}$ inequalities, no matter how many auxiliary variables are added. Note the dimension d is $\Theta(n^2)$, so this lower bound has the form $2^{\Omega(\sqrt{d})}$.

Elementary reductions (see the Exercises) translate this extension complexity lower bound for the COR polytope to a lower bound of $2^{\Omega(\sqrt{n})}$ on the size of extended formulations of the convex hull of characteristic vectors of n-point traveling salesman tours.

4.3 Some Faces of the Correlation Polytope

Next we establish a key connection between certain faces of the correlation polytope and inputs to Disjointness. Throughout, n is a fixed positive integer.

Lemma 4.2 ([4]) *For every subset $S \subseteq \{1,2,\ldots,n\}$, there is a face f_S of* COR *such that: for every $R \subseteq \{1,2,\ldots,n\}$ with characteristic vector $\mathbf{x}_R$ and corresponding vertex $\mathbf{v}_R = \mathbf{x}_R\mathbf{x}_R^T$ of* COR*,*

$$\mathbf{v}_R \in f_S \quad \text{if and only if} \quad |S \cap R| = 1.$$

[14] This paper won the Best Paper Award at STOC '14.

That is, among the faces of COR are 2^n faces that encode the "unique intersection property" for each of the 2^n subsets S of $\{1, 2, \ldots, n\}$. Note that for a given S, the sets R with $|S \cap R|$ can be generated by (i) first picking a element of S; (ii) picking a subset of $\{1, 2, \ldots, n\} \setminus S$. Thus if $|S| = k$, there are $k2^{n-k}$ sets R with which it has a unique intersection.

Lemma 4.2 is kind of amazing, but also not too hard to prove.

Proof of Lemma 4.2: For every $S \subseteq \{1, 2, \ldots, n\}$, we need to exhibit a supporting hyperplane $\mathbf{a^T x} \leq b$ such that $\mathbf{a^T v}_R = b$ if and only if $|S \cap R| = 1$, where $\mathbf{v}_R$ denotes $\mathbf{x}_R \mathbf{x}_R^T$ and $\mathbf{x}_R$ the characteristic vector of $R \subseteq \{1, 2, \ldots, n\}$.

Fix $S \subseteq \{1, 2, \ldots, n\}$. We develop the appropriate supporting hyperplane, in variables $\mathbf{y} \in \mathbb{R}^{n^2}$, over several small steps.

1. For clarity, let's start in the wrong space, with variables $\mathbf{z} \in \mathbb{R}^n$ rather than $\mathbf{y} \in \mathbb{R}^{n^2}$. Here $\mathbf{z}$ is meant to encode the characteristic vector of a set $R \subseteq \{1, 2, \ldots, n\}$. One sensible inequality to start with is

$$\sum_{i \in S} z_i - 1 \geq 0. \tag{9}$$

   For example, if $S = \{1, 3\}$, then this constraint reads $z_1 + z_3 - 1 \geq 0$.

   The good news is that for 0-1 vectors $\mathbf{x}_R$, this inequality is satisfied with equality if and only if $|S \cap R| = 1$. The bad news is that it does not correspond to a supporting hyperplane: if S and R are disjoint, then $\mathbf{x}_R$ violates the inequality. How can we change the constraint so that it holds with equality for $\mathbf{x}_R$ with $|S \cap R| = 1$ and also valid for all R?

2. One crazy idea is to square the left-hand side of (9):

$$\left(\sum_{i \in S} z_i - 1 \right)^2 \geq 0. \tag{10}$$

   For example, if $S = \{1, 3\}$, then the constraint reads (after expanding) $z_1^2 + z_3^2 + 2z_1 z_3 - 2z_1 - 2z_3 + 1 \geq 0$.

   The good news is that every 0-1 vector $\mathbf{x}_R$ satisfies this inequality, and equality holds if and only if $|S \cap R| = 1$. The bad news is that the constraint is non-linear and hence does not correspond to a supporting hyperplane.

3. The obvious next idea is to "linearize" the previous constraint. Wherever the constraint has a $\mathbf{z}_i^2$ or a $\mathbf{z}_i$, we replace it by a variable $\mathbf{y}_{ii}$ (note these partially cancel out). Wherever the constraint has a $2z_i z_j$ (and notice for $i \neq j$ these always come in pairs), we replace it by a $y_{ij} + y_{ji}$. Formally, the constraint now reads

$$-\sum_{i \in S} y_{ii} + \sum_{i \neq j \in S} y_{ij} + 1 \geq 0. \tag{11}$$

Note that the new variable set is $\mathbf{y} \in \mathbb{R}^{n^2}$. For example, if $S = \{1, 3\}$, then the new constraint reads $y_{13} + y_{31} - y_{11} - y_{33} \geq -1$.

A first observation is that, for $\mathbf{y}$'s that are 0-1, symmetric, and rank-1, with $\mathbf{y} = \mathbf{z}\mathbf{z}^T$ (hence $y_{ij} = z_i \cdot z_j$ for $i, j \in \{1, 2, \ldots, n\}$), the left-hand sides of (10) and (11) are the same by definition. Thus, for $\mathbf{y} = \mathbf{x}_R\mathbf{x}_R^T$ with $\mathbf{x} \in \{0, 1\}^n$, $\mathbf{y}$ satisfies the (linear) inequality (11), and equality holds if and only if $|S \cap R| = 1$.

We have shown that, for every $S \subseteq \{1, 2, \ldots, n\}$, the linear inequality (11) is satisfied by every vector $\mathbf{y} \in \mathbb{R}^{n^2}$ of the form $\mathbf{y} = \mathbf{x}_R\mathbf{x}_R^T$ with $\mathbf{x} \in \{0, 1\}^n$. Since COR is by definition the convex hull of such vectors, every point of COR satisfies (11). This inequality is therefore a supporting hyperplane, and the face it induces contains precisely those vertices of the form $\mathbf{x}_R\mathbf{x}_R^T$ with $|S \cap R| = 1$. This completes the proof. ■

4.4 Face-Vertex(COR) and Unique-Disjointness

In the Face-Vertex(COR) problem, Alice receives a face f of COR and Bob a vertex $\mathbf{v}$ of COR. In the 1-inputs, $\mathbf{v} \notin f$; in the 0-inputs, $\mathbf{v} \in F$. Let's make the problem only easier by restricting Alice's possible inputs to the 2^n faces (one per subset $S \subseteq \{1, 2, \ldots, n\}$) identified in Lemma 4.2. In the corresponding matrix M_U of this function, we can index the rows by subsets S. Since every vertex of COR has the form $\mathbf{y} = \mathbf{x}_R\mathbf{x}_R^T$ for $R \subseteq \{1, 2, \ldots, n\}$, we can index the columns of M_U by subsets R. By Lemma 4.2, the entry (S, R) of the matrix M_U is 1 if $|S \cap R| \neq 1$ and 0 of $|S \cap R| = 1$. That is, the 0-entries of M_U correspond to pairs (S, R) that intersect in a unique element.

There is clearly a strong connection between the matrix M_U above and the analogous matrix M_D for Disjointness. They differ on entries (S, R) with $|S \cap R| \geq 2$: these are 0-entries of M_D but 1-entries of M_U. In other words, M_U is the matrix corresponding to the communication problem ¬Unique-Intersection: do the inputs S and R fail to have a unique intersection?

The closely related Unique-Disjointness problem is a "promise" version of Disjointness. The task here is to distinguish between:

(1) inputs (S, R) of Disjointness with $|S \cap R| = 0$;

(0) inputs (S, R) of Disjointness with $|S \cap R| = 1$.

For inputs that fall into neither case (with $|S \cap R| > 1$), the protocol is off the hook — any output is considered correct. Since a protocol that solves Unique-Disjointness has to do only less than one that solves ¬Unique-Intersection, communication complexity lower bounds for former problem apply immediate to the latter.

We summarize the discussion of this section in the following proposition.

Proposition 4.3 ([4]) *The nondeterministic communication complexity of* Face-Vertex(COR) *is at least that of* Unique-Disjointness.

4.5 A Lower Bound for Unique-Disjointness

4.5.1 The Goal

One final step remains in our proof of Theorem 4.1, and hence of our lower bound on the size of extended formulations of the correlation polytope.

Theorem 4.4 ([4, 7]) *The nondeterministic communication complexity of* Unique-Disjointness *is* $\Omega(n)$.

4.5.2 Disjointness Revisited

As a warm-up, we revisit the standard Disjointness problem. Recall that, in Lecture #4, we proved that the nondeterministic communication complexity of Disjointness is at least n by a fooling set argument. Next we prove a slightly weaker lower bound, via an argument that generalizes to Unique-Disjointness.

The first claim is that, of the $2^n \times 2^n = 4^n$ possible inputs of Disjointness, exactly 3^n of them are 1-inputs. The reason is that the following procedure, which makes n 3-way choices, generates every 1-input exactly once: independently for each coordinate $i = 1, 2, \ldots, n$, choose between the options (i) $x_i = y_i = 0$; (ii) $x_i = 1$ and $y_i = 0$; and (iii) $x_i = 0$ and $y_i = 1$.

The second claim is that every 1-rectangle — every subset A of rows of M_D and B of columns of M_D such that $A \times B$ contains only 1-inputs — has size at most 2^n. To prove this, let $R = A \times B$ be a 1-rectangle. We assert that, for every coordinate $i = 1, 2, \ldots, n$, either (i) $x_i = 0$ for all $\mathbf{x} \in A$ or (ii) $y_i = 0$ for all $\mathbf{y} \in B$. That is, every coordinate has, for at least one of the two parties, a "forced zero" in R. For if neither (i) nor (ii) hold for a coordinate i, then since R is a rectangle (and hence closed under "mix and match") we can choose $(\mathbf{x}, \mathbf{y}) \in R$ with $x_i = y_i = 1$; but this is a 0-input and R is a 1-rectangle. This assertion implies that the following procedure, which makes n 2-way choices, generates every 1-input of R (and possibly other inputs as well): independently for each coordinate $i = 1, 2, \ldots, n$, set the forced zero ($x_i = 0$ in case (i) or $y_i = 0$ in case (ii)) and choose a bit for this coordinate in the other input.

These two claims imply that every covering of the 1-inputs by 1-rectangles requires at least $(3/2)^n$ rectangles. Proposition 2.2 then implies a lower bound of $\Omega(n)$ on the nondeterministic communication complexity of Disjointness.

4.5.3 Proof of Theorem 4.4

Recall that the 1-inputs $(\mathbf{x}, \mathbf{y})$ of Unique-Disjointness are the same as those of Disjointness (for each i, either $x_i = 0$, $y_i = 0$, or both). Thus, there are still exactly 3^n 1-inputs. The 0-inputs $(\mathbf{x}, \mathbf{y})$ of Unique-Disjointness are those with $x_i = y_i = 1$ in exactly one coordinate i. We call all other inputs, where the promise fails to hold, **-inputs*. By a 1-rectangle, we now mean a rectangle with no 0-inputs (*-inputs are fine). With this revised definition, it is again true that every nondeterministic communication protocol that solves

UNIQUE-DISJOINTNESS using c bits of communication induces a covering of the 1-inputs by at most 2^c 1-rectangles.

Lemma 4.5 *Every 1-rectangle of* UNIQUE-DISJOINTNESS *contains at most* 2^n *1-inputs.*

As with the argument for DISJOINTNESS, Lemma 4.5 completes the proof of Theorem 4.4: since there are 3^n 1-inputs and at most 2^n per 1-rectangle, every covering by 1-rectangles requires at least $(3/2)^n$ rectangles. This implies that the nondeterministic communication complexity of UNIQUE-DISJOINTNESS is $\Omega(n)$.

Why is the proof of Lemma 4.5 harder than in Section 4.5.2? We can no longer easily argue that, in a rectangle $R = A \times B$, for each coordinate i, either $x_i = 0$ for all $\mathbf{x} \in A$ or $y_i = 0$ for all $\mathbf{y} \in B$. Assuming the opposite no longer yields a contraction: exhibiting $\mathbf{x} \in A$ and $\mathbf{y} \in B$ with $x_i = y_i = 1$ does not necessarily contradict the fact that R is a 1-rectangle, since $(\mathbf{x}, \mathbf{y})$ might be a *-input.

Proof of Lemma 4.5: The proof is one of those slick inductions that you can't help but sit back and admire.

We claim, by induction on $k = 0, 1, 2, \ldots, n$, that if $R = A \times B$ is a 1-rectangle for which all $\mathbf{x} \in A$ and $\mathbf{y} \in B$ have 0s in their last $n - k$ coordinates, then the number of 1-inputs in R is at most 2^k. The lemma is equivalent to the case of $k = n$. The base case $k = 0$ holds, because in this case the only possible input in R is $(\mathbf{0}, \mathbf{0})$.

For the inductive step, fix a 1-rectangle $R = A \times B$ in which the last $n - k$ coordinates of all $\mathbf{x} \in A$ and all $\mathbf{y} \in B$ are 0. To simplify notation, from here on we ignore the last $n - k$ coordinates of all inputs (they play no role in the argument).

Intuitively, we need to somehow "zero out" the kth coordinate of all inputs in R so that we can apply the inductive hypothesis. This motivates focusing on the kth coordinate, and we'll often write inputs $\mathbf{x} \in A$ and $\mathbf{y} \in B$ as $\mathbf{x}'a$ and $\mathbf{y}'b$, respectively, with $\mathbf{x}', \mathbf{y}' \in \{0,1\}^{k-1}$ and $a, b \in \{0, 1\}$. (Recall we're ignoring that last $n - k$ coordinates, which are now always zero.)

First observe that, whenever $(\mathbf{x}'a, \mathbf{y}'b)$ is a 1-input, we cannot have $a = b = 1$. Also:

(*) If $(\mathbf{x}'a, \mathbf{y}'b) \in R$ is a 1-input, then R cannot contain both the inputs $(\mathbf{x}'0, \mathbf{y}'1)$ and $(\mathbf{x}'1, \mathbf{y}'0)$.

For otherwise, R would also contain the 0-input $(\mathbf{x}'1, \mathbf{y}'1)$, contradicting that R is a 1-rectangle. (Since $(\mathbf{x}'a, \mathbf{y}'b)$ is a 1-input, the unique coordinate of $(\mathbf{x}'1, \mathbf{y}'1)$ with a 1 in both inputs is the kth coordinate.)

The plan for the rest of the proof is to define two sets S_1, S_2 of 1-inputs — not necessarily rectangles — such that:

(P1) the number of 1-inputs in S_1 and S_2 combined is at least that in R;

(P2) the inductive hypothesis applies to $\text{rect}(S_1)$ and $\text{rect}(S_2)$, where $\text{rect}(S)$ denotes the smallest rectangle containing a set S of inputs.[15]

[15] Equivalently, the closure of S under the "mix and match" operation on pairs of inputs. Formally, $\text{rect}(S) = X(S) \times Y(S)$, where $X(S) = \{\mathbf{x} : (\mathbf{x}, \mathbf{y}) \in S \text{ for some } \mathbf{y}\}$ and $Y(S) = \{\mathbf{y} : (\mathbf{x}, \mathbf{y}) \in S \text{ for some } \mathbf{x}\}$.

If we can find sets S_1, S_2 with properties (P1),(P2), then we are done: by the inductive hypothesis, the rect(S_i)'s have at most 2^{k-1} 1-inputs each, the S_i's are only smaller, and hence (by (P1)) R has at most 2^k 1-inputs, as required.

We define the sets in two steps, focusing first on property (P1). Recall that every 1-input $(\mathbf{x}, \mathbf{y}) \in R$ has the form $(\mathbf{x}'1, \mathbf{y}'0)$, $(\mathbf{x}'0, \mathbf{y}'1)$, or $(\mathbf{x}'0, \mathbf{y}'0)$. We put all 1-inputs of the first type into a set S'_1, and all 1-inputs of the second type into a set S'_2. When placing inputs of the third type, we want to avoid putting two inputs of the form $(\mathbf{x}'a, \mathbf{y}'b)$ with the same $\mathbf{x}'$ and $\mathbf{y}'$ into the same set (this would create problems in the inductive step). So, for an input $(\mathbf{x}'0, \mathbf{y}'0) \in R$, we put it in S'_1 if and only if the input $(\mathbf{x}'1, \mathbf{y}'0)$ was not already put in S'_1; and we put it in S'_2 if and only if the input $(\mathbf{x}'0, \mathbf{y}'1)$ was not already put in S'_2. Crucially, observation (*) implies that R cannot contain two 1-inputs of the form $(\mathbf{x}'1, \mathbf{y}'0)$ and $(\mathbf{x}'0, \mathbf{y}'1)$, so the 1-input $(\mathbf{x}'0, \mathbf{y}'0)$ is placed in at least one of the sets S'_1, S'_2. (It is placed in both if R contains neither $(\mathbf{x}'1, \mathbf{y}'0)$ nor $(\mathbf{x}'0, \mathbf{y}'1)$.) By construction, the sets S'_1 and S'_2 satisfy property (P1).

We next make several observations about S'_1 and S'_2. By construction:

(**) for each $i = 1, 2$ and $\mathbf{x}', \mathbf{y}' \in \{0,1\}^{k-1}$, there is at most one input of S'_i of the form $(\mathbf{x}'a, \mathbf{y}'b)$.

Also, since S'_1, S'_2 are subsets of the rectangle R, rect(S'_1), rect(S'_2) are also subsets of R. Since R is a 1-rectangle, so are rect(S'_1), rect(S'_2). Also, since every input $(\mathbf{x}, \mathbf{y})$ of S'_i (and hence rect(S'_i)) has $y_k = 0$ (for $i = 1$) or $x_k = 0$ (for $i = 2$), the kth coordinate contributes nothing to the intersection of any inputs of rect(S'_1) or rect(S'_2).

Now obtain S_i from S'_i (for $i = 1, 2$) by zeroing out the kth coordinate of all inputs. Since the S'_i's only contain 1-inputs, the S_i's only contain 1-inputs. Since property (**) implies that $|S_i| = |S'_i|$ for $i = 1, 2$, we conclude that property (P1) holds also for S_1, S_2.

Moving on to property (P2), since rect(S'_1), rect(S'_2) contain no 0-inputs and contain only inputs with no intersection in the kth coordinate, rect(S_1), rect(S_2) contain no 0-inputs.[16] Finally, since all inputs of S_1, S_2 have zeroes in their final $n - k + 1$ coordinates, so do all inputs of rect(S_1), rect(S_2). The inductive hypothesis applies to rect(S_1) and rect(S_2), so each of them has at most 2^{k-1} 1-inputs. This implies the inductive step and completes the proof. ■

References

[1] V. Chvátal. *Linear Programming*. Freeman, 1983.

[2] R. de Wolf. Nondeterministic quantum query and quantum communication complexities. *SIAM Journal on Computing*, 32(3):681–699, 2003.

[16] The concern is that zeroing out an input in the kth coordinate turns some *-input (with intersection size 2) into a 0-input (with intersection size 1); but since there were no intersections in the kth coordinate, anyways, this can't happen.

[3] J. Edmonds. Maximum matching and a polyhedron with 0,1-vertices. *Journal of Research of the National Bureau of Standards, Series B*, 69B:125–130, 1965.

[4] S. Fiorini, S. Massar, S. Pokutta, H. R. Tiwary, and R. de Wolf. Exponential lower bounds for polytopes in combinatorial optimization. *Journal of the ACM*, 62(2), 2015. Article 17.

[5] M. X. Goemans. Smallest compact formulation for the permutahedron. *Mathematical Programming, Series A*, 2014. To appear.

[6] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer, 1988. Second Edition, 1993.

[7] V. Kaibel and S. Weltge. A short proof that the extension complexity of the correlation polytope grows exponentially. *Discrete & Computational Geometry*, 53(2):397–401, 2015.

[8] L. G. Khachiyan. A polynomial algorithm in linear programming. *Soviet Mathematics Doklady*, 20(1):191–194, 1979.

[9] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955.

[10] L. Lovász. Communication complexity: A survey. In B. H. Korte, editor, *Paths, Flows, and VLSI Layout*, pages 235–265. Springer-Verlag, 1990.

[11] M. Padberg and M. R. Rao. Odd minimum cut-sets and b-matchings. *Mathematics of Operations Research*, 7:67–80, 1982.

[12] I. Pitowsky. Correlation polytopes: Their geometry and complexity. *Mathematical Programming*, 50:395–414, 1991.

[13] W. Pulleyblank and J. Edmonds. Facets of 1-matching polyhedra. In *Proceedings of the Working Seminar on Hypergraphs*, pages 214–242. Springer, 1974.

[14] T. Rothvoß. The matching polytope has exponential extension complexity. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*, pages 263–272, 2014.

[15] M. Yannakakis. Expressing combinatorial optimization problems by linear programs. *Journal of Computer and System Sciences*, 43(3):441–466, 1991.

[16] G. M. Ziegler. *Lectures on Polytopes*. Springer, 1995.