On the Power of Symmetric Linear Programs

Albert Atserias
Universitat Politècnica de Catalunya

Anuj Dawar
University of Cambridge

Joanna Ochremiak
University of Cambridge and CNRS, LaBRI

***Abstract*—We consider families of symmetric linear programs (LPs) that decide a property of graphs (or other relational structures) in the sense that, for each size of graph, there is an LP defining a polyhedral lift that separates the integer points corresponding to graphs with the property from those corresponding to graphs without the property. We show that this is equivalent, with at most polynomial blow-up in size, to families of symmetric Boolean circuits with threshold gates. In particular, when we consider polynomial-size LPs, the model is equivalent to definability in a non-uniform version of fixed-point logic with counting (FPC). Known upper and lower bounds for FPC apply to the non-uniform version. In particular, this implies that the class of graphs with perfect matchings has polynomial-size symmetric LPs while we obtain an exponential lower bound for symmetric LPs for the class of Hamiltonian graphs. We compare and contrast this with previous results (Yannakakis 1991) showing that any symmetric LPs for the matching and TSP polytopes have exponential size. As an application, we establish that for random, uniformly distributed graphs, polynomial-size symmetric LPs are as powerful as general Boolean circuits. We illustrate the effect of this on the well-studied planted-clique problem.**

I. Introduction

The theory of linear programming is a powerful and widely-used tool for studying combinatorial optimization problems. By the same token, the limitations of such methods are an important object of study in complexity theory. A major step in this line of work was the seminal paper of Yannakakis [1] that initiated the study of *symmetric* linear programs for combinatorial problems.

A polytope in $\mathbb{R}^n$ is the convex hull of a finite set of points in $\mathbb{R}^n$. Dually, it is the intersection of the finite number of half-spaces that define its facets. Consider a language $S \subseteq \{0,1\}^*$ and let $S_n \subseteq \{0,1\}^n$ be the collection of strings in S of length n. We can associate with S_n the polytope $P_n \subseteq \mathbb{R}^n$ that is the convex hull of the points $\mathbf{x} \in \mathbb{R}^n$ with 0-1 coordinates that correspond to the strings in S_n. If this polytope has a succinct representation as a system of linear inequalities, we can use linear programming methods to optimize linear functions over S_n. In general, a succinct representation might mean that its size grows polynomially with n. Thus, the size of the polytope P_n, say measured by the number of its facets, is an important measure of the complexity of S.

In general, even when P_n has a large number of facets, it may admit a succinct representation as the projection onto $\mathbb{R}^n$ of a polytope $Q \subseteq \mathbb{R}^{n+m}$ of higher dimension. In this situation, we call Q a *lift* of P_n and P_n a *shadow* of Q. This is the basis for so-called *extended formulations* of combinatorial optimization problems. It allows us to optimize over S_n using linear programs with auxiliary variables. A classic example is the convex hull of all strings in $\{0,1\}^n$ of odd Hamming weight, known as the *parity polytope*, which has exponentially many facets but has an extended formulation using only polynomially many inequalities. An interesting feature of many such examples of small extended formulations is that they are strongly symmetric, i.e., any basic automorphism of the shadow polytope extends to an automorphism of its lift.

Yannakakis [1] established lower bounds on the size of symmetric lifts for the perfect matching polytope and the travelling salesman polytope. The *perfect matching polytope* on $2n$ vertices is the convex hull of points in $\{0,1\}^E$ where $E = \binom{[2n]}{2}$ which represent the edge sets of a perfect matching on $2n$ vertices. Yannakakis shows that any symmetric lift Q of this polytope necessarily has a number of facets that is exponential in n. Here "symmetric" means that any permutation of the n vertices extends to an automorphism of Q. This lower bound is then used to show a similar lower bound for symmetric lifts of the Hamilton cycle polytope (also known as the travelling salesman polytope). This is the convex hull of points in $\{0,1\}^E$ where $E = \binom{[n]}{2}$ which are the edge sets of Hamilton cycles of length n. The conclusion is that any attempt to solve the travelling salesman problem by representing it as a linear program in a natural way (i.e. respecting the symmetries of the graph) is doomed to be exponential. These results launched a long study of extended formulations of combinatorial problems. Relatively recently, exponential lower bounds have been established even without the assumption of symmetry [2].

There is another way of representing a language $S \subseteq \{0,1\}^*$ by a family of polytopes that is also considered by Yannakakis. Say that S_n is *recognized* by a polytope P_n if $S_n \subseteq P_n$ and $\{0,1\}^n \setminus S_n$ is disjoint from P_n. In particular, the convex hull of S_n recognizes S_n, but it may well be that there are more succinct polytopes that also do. Indeed, Yannakakis shows that for any language S decidable in polynomial time, there is a family of polynomial-size polytopes whose shadows recognize S_n. Thus, we cannot expect to prove exponential lower bounds on such polytopes without separating P from NP. Note that the assumption of symmetry has been dropped here. What can we say about symmetric lifts of polytopes recognizing S_n? Yannakakis does not consider this question and it does not appear to have been studied in the literature. This is the question that we take up in this paper.

We consider families of symmetric polytopes for recognizing classes of graphs (or other relational structures). This gives an interesting contrast with the results of Yannakakis. Our results show that there *is* a polynomial-size family of symmetric polytopes whose shadows recognize the class of graphs that contain a perfect matching. On the other hand, there is no family of symmetric polytopes of sub-exponential size whose shadows recognize the class of graphs with a Hamiltonian cycle.

We obtain these specific upper and lower bounds by relating the power of symmetric linear programs to two other natural models of symmetric computation, based respectively on logic and circuits. To be precise, we show that families of symmetric polytope lifts for recognizing a class of structures are equivalent to families of *symmetric* Boolean circuits with threshold gates, in the sense that there are translations between them with at most a polynomial blow-up in size in either direction. This places symmetric linear programs squarely in the context of a fairly robust notion of symmetric computation that has recently emerged. It was shown in [3] that P-uniform families of symmetric circuits with threshold gates are equivalent to fixed-point logic with counting (FPC), a well-studied logic in descriptive complexity theory (see [4]).

Our translation from circuits to linear programs is based on that given by Yannakakis, but we need to preserve symmetry and, for threshold gates, this poses a significant challenge. To construct symmetric linear programs that enforce the values of threshold gates we need a sweeping generalization of the construction of symmetric lifts of the parity polytope. In the other direction, we make a detour through logic. That is, we show how a family of symmetric polytopes can be translated into a family of formulas of first-order logic with counting (FOC), with the number of variables and the size being tightly bounded based on the size of the polytopes. The translation is based on a support theorem, which allows us to interpret in the logic, given a linear program P as advice, a version of P for a particular input structure. This then allows us to use the result of [5] to the effect that solvability of linear programs is definable in FPC.

It is interesting to compare our results with the equivalence between FPC and P-uniform symmetric threshold circuits established in [3]. Our results are stated for the non-uniform model and it is not clear that they can be made uniform. While both the standard translation from formulas of counting logic to circuits and our translation from circuits to linear programs are computable in polynomial time, for the remaining direction, it is not necessarily the case. Indeed, our translation from linear programs to formulas of counting logic involves symmetry checks that are as hard as the graph isomorphism problem. On the other hand, the results in [3] were stated for polynomial-size families of circuits and we are able to extend them to sizes up to weakly exponential. The translation from circuits to formulas given in [3] was based on a support theorem proved there which only worked for circuit sizes bounded by $O(2^{n^{1/3}})$. We use a stronger support theorem (proved in Section IV-B) which enables us to prove the translation from families of symmetric linear programs to formulas of counting logic for sizes up to $O(2^{n^{1-\epsilon}})$ for arbitrarily small ϵ.

The upper and lower bounds for symmetric linear programs that we obtain (such as for the perfect matching and the Hamilton cycle problem, respectively) are direct consequences of equivalence with the non-uniform counting logic. For instance, it is known [5] that perfect matching is definable in FPC and it follows that it is recognized by a polynomial-size family of symmetric polytope lifts. Inexpressibility results for FPC are usually established by showing lower bounds on the number of variables required to express a property in counting logic, and they yield lower bounds even in the non-uniform setting. In particular, we tighten known lower bounds on Hamiltonicity to show that it cannot be expressed with a sub-linear number of variables and hence with sub-exponential size symmetric polytope lifts. Similar exponential lower bounds for other NP-complete problems (such as graph 3-colourability and Boolean satisfiability) follow from known bounds in counting logic. Indeed, exponential lower bounds for some problems in P (such as solving systems of linear equations over finite fields) also follow. It should be noted that this establishes exponential lower bounds also on symmetric threshold circuits, a problem left open in [3], where superpolynomial lower bounds were established.

Another consequence can be derived from the connection with FPC. We know that FPC can express all polynomial-time properties of *almost all structures* under a uniform distribution (see [6]). This can be used to show that FPC can solve the planted clique problem if, and only if, the problem is solvable in polynomial time. The planted clique problem is that of distinguishing a random graph from one in which a clique has been planted. It is a widely studied problem in the context of lower bounds on linear programming methods (see e.g. [7]–[10]). It is a consequence of our results that if this problem can be solved in polynomial time, then it is solvable by polynomial sized symmetric linear programs. This is significant because a number of lower bounds have been established for the planted clique problem for a variety of models of linear and semidefinite programming, notably the well-studied Lovász-Schrijver, Sherali-Adams and Lasserre hierarchies. It is noteworthy that all of these hierarchies yield symmetric linear or semidefinite programs. Our results show that these lower bounds cannot be extended to general symmetric linear programs without separating P from NP.

In Section II we establish some preliminary definitions and notation. Section III gives the translation of circuits to linear programs. This translation is carried out for a very general notion of symmetry. For the reverse translation, from linear programs to logic given in Section IV, we restrict to the natural symmetries on graphs and relational structures. The main result and its consequences, including upper and lower bounds are presented in Section V.

II. Preliminaries

For $n \in \mathbb{N}$, we write $[n]$ for $\{1, \ldots, n\}$, where $[0] = \emptyset$. For a set X and $0 \leq n \leq |X|$, by $X^{(n)}$ we denote the set of all n-tuples of *distinct* elements of X. Logarithms are base 2 with the convention that $\log(0) = 0$. For any set W, Sym_W is the group of all permutations of W and Alt_W is the group of all even permutations of W.

Group actions. Recall that for any group G, a *G-set* is a set X with an action of the group G. For any G-set U, the action of G on the set of indexed variables $\{x_u\}_{u \in U}$ is given by: $\pi \cdot x_u = x_{\pi \cdot u}$, for any $\pi \in G$ and $u \in U$. This extends to vectors of indexed variables. For vectors of variables we use the notation $\mathbf{x}^\pi$ instead of $\pi \cdot \mathbf{x}$. We also define an action of G on $\mathbb{R}^U$ as follows. Let $\{e_u\}_{u \in U}$ be the standard basis. For any $\pi \in G$ and any real vector $\mathbf{a} = \sum_{u \in U} a_u e_u$, we put $\pi \cdot \mathbf{a} = \sum_{u \in U} a_u e_{\pi \cdot u}$. Here again we use $\mathbf{a}^\pi$ instead of $\pi \cdot \mathbf{a}$. This extends to subsets of real vector spaces: for $P \subseteq \mathbb{R}^U$ we write P^π instead of $\pi \cdot P$.

For any G-set U and any H-set W, the product group $G \times H$ acts on the disjoint union $U \dot\cup W$: given $\pi \in G$ and $\sigma \in H$, we have $(\pi, \sigma) \cdot u = \pi \cdot u$, for $u \in U$, and $(\pi, \sigma) \cdot w = \sigma \cdot w$, for $w \in W$. Of particular interest to us is the induced action of $G \times H$ on $\mathbb{R}^U \times \mathbb{R}^W$ and on sets of variables indexed by $U \dot\cup W$.

Logic and structures. For definitions of (relational) vocabularies, first-order logic and its various extensions with counting and fixed-point operators see e.g. [11].

Rational numbers are represented by structures of a single-sorted vocabulary $L_{\mathbb{Q}}$ with three monadic relation symbols and one binary relation symbol $\leq$. If $q = (-1)^b n/d$, where $n, d \in \mathbb{N}$ and $b \in \{0, 1\}$, then the domain of an $L_{\mathbb{Q}}$-structure that represents q is $\{0, \ldots, N\}$ where $N \in \mathbb{N}$ is large enough to represent the numerator and denominator with N bits. The binary relation $\leq$ is interpreted by the natural linear order on $\{0, \ldots, N\}$. The first of the monadic relation symbols of $L_{\mathbb{Q}}$ is used to represent the sign b of q by having it empty if, and only if, $b = 0$. The other two monadic relation symbols of $L_{\mathbb{Q}}$ are used to represent the bit positions on which the numerator n and the denominator d have a one. We use zero denominator to represent $\pm\infty$.

If $I_1, \ldots, I_d$ are index sets, tensors $u \in \mathbb{Q}^{I_1 \times \cdots \times I_d}$ are represented by many-sorted structures, with one sort $\bar{I}$ for each index set, and one sort $\bar{B}$ for a domain $\{0, \ldots, N\}$ of bit positions. The vocabulary $L_{\mathrm{vec},d}$ of these structures has a binary relation symbol $\leq$ for the natural linear order on $\{0, \ldots, N\}$ and three $d+1$-ary relation symbols P_s, P_n and P_d for encoding the signs and the bits of the numerators and the denominators of the entries of the tensor. Matrices $\mathbf{A} \in \mathbb{Q}^{I \times J}$ and vectors $\mathbf{a} \in \mathbb{Q}^I$ are special cases of these.

Polytopes, lifts, and shadows. A polytope is a set of the form $P = \{\mathbf{x} \in \mathbb{R}^U : \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$, where $\mathbf{A} \in \mathbb{R}^{V \times U}$ is a constraint matrix, and $\mathbf{b} \in \mathbb{R}^V$ is a vector. Typically $\mathbf{A}$ and $\mathbf{b}$ can be chosen to have rational entries, in which case the *size* of P is $(|U| + 1)|V|b$, where b is the maximum number of bits it takes to write all the numerators and all the denominators of the entries of $\mathbf{A}$ and $\mathbf{b}$ in binary.

If we think of $\mathbf{x} = (x_u)_{u \in U}$ as a sequence of variables, then P is represented by a sequence of linear constraints $(\gamma_v)_{v \in V}$ each of the form $\mathbf{a}_v^{\mathrm{T}}\mathbf{x} \leq b_v$. If U is a G-set, then for any γ_v of the form $\mathbf{a}_v^{\mathrm{T}}\mathbf{x} \leq b_v$, we write γ_v^π for the linear constraint $\mathbf{a}_v^{\mathrm{T}}\mathbf{x}^\pi \leq b_v$. Note that the sequence $(\gamma_v^\pi)_{v \in V}$ defines $P^\pi \subseteq \mathbb{R}^U$, which is again a polytope.

If $P \subseteq \mathbb{R}^U \times \mathbb{R}^W$ is a polytope, and Q is its projection into $\mathbb{R}^U$, then we say that Q is a *shadow* of P, and that P is a *lift* of Q. If $A, B \subseteq \{0, 1\}^U$ are disjoint, then Q *separates* A from B if $A \subseteq Q$ and $B \subseteq \mathbb{R}^U \setminus Q$. We also say that P is a lift that separates A from B. If Q separates A from its complement $\overline{A} = \{0, 1\}^U \setminus A$, then Q *recognizes* A, and P is a lift that recognizes A.

Let U be a G-set. A polytope $P \subseteq \mathbb{R}^U \times \mathbb{R}^W$ is *G-symmetric* if for every $\pi \in G$ there exists $\sigma \in \mathrm{Sym}_W$ such that $P^{(\pi,\sigma)} = P$. If additionally we are given an action of the group G on W such that $P^{(\pi,\pi)} = P$, for any $\pi \in G$, then the polytope P is G-symmetric *with respect to this action*. A pair of permutations $(\pi, \sigma) \in \mathrm{Sym}_U \times \mathrm{Sym}_W$ such that $P^{(\pi,\sigma)} = P$ is an *automorphism* of P.

For $n \in \mathbb{N}$, if $[n]^2$ comes with the natural action of the symmetric group Sym_n, then any Sym_n-symmetric polytope $P \subseteq \mathbb{R}^{[n]^2} \times \mathbb{R}^W$ is said to be *graph-symmetric*. Any set $A \subseteq \{0,1\}^{[n]^2}$ recognised by a graph-symmetric polytope lift $P \subseteq \mathbb{R}^{[n]^2} \times \mathbb{R}^W$ is invariant with respect to the action of the group Sym_n, i.e., for any $\mathbf{a} \in A$ and any $\pi \in \mathrm{Sym}_n$, we have $\mathbf{a}^\pi \in A$. P can be therefore seen as recognising a class of graphs with n-vertices. If we take a graph G with the set of vertices V of size n, fix a bijection f from $[n]$ to V, and define $\mathbf{a} = (a_{ij})_{i,j \in [n]} \in \{0,1\}^{[n]^2}$ by: $a_{ij} = 1$ if, and only if, there is an edge from $f(i)$ to $f(j)$ in G, then G belongs to the class recognised by P if and only if $\mathbf{a} \in A$. Since A is a Sym_n-set, this does not depend on the choice of f.

More generally, we consider Sym_n-symmetric polytope lifts recognising properties of arbitrary L-structures. For any $n \in \mathbb{N}$ and any single-sorted vocabulary L, let $L(n)$ be the disjoint union of $[n]^{\mathrm{ar}(R)}$ over all relation symbols R in L. Since $L(n)$ comes with the natural action of the group $\mathrm{Sym}(n)$, we can talk about Sym_n-symmetric polytopes over $\mathbb{R}^{L(n)} \times \mathbb{R}^W$. Any such polytope is called *L-symmetric*. Similarly as in the previous paragraph, L-symmetric polytope lifts over $\mathbb{R}^{L(n)} \times \mathbb{R}^W$ recognise classes of L-structures with n-element domains.

Boolean circuits. A *Boolean threshold circuit* is one whose gates are labelled by NOTs, unbounded degree ANDs, unbounded degree ORs, or unbounded degree thresholds $\mathrm{TH}_{n,k}$, where $\mathrm{TH}_{n,k}(z_1, \ldots, z_n)$ outputs 1 if, and only if, the number of 1's in the input $z_1, \ldots, z_n$ is at least k.

If U is a G-set, then a circuit C with inputs $(x_u)_{u \in U}$ and the set W of gates is *G-symmetric* if for every $\pi \in G$ there exists $\sigma \in \mathrm{Sym}_W$ such that $C^{(\pi,\sigma)} = C$, where by $C^{(\pi,\sigma)}$ we mean that the gates of the circuit are permuted according to σ, the labels from $\{x_u\}_{u \in U}$ are permuted according to π, and none of the other labels is moved. A circuit with $U = L(n)$ is called L-symmetric if it is Sym_n-symmetric, with the natural action of Sym_n on $L(n)$. As for polytopes, we consider L-symmetric circuits as recognizing classes of L-structures on abstract sets V of vertices through bijections $f : [n] \to V$. In the case of graphs, for example, in which $L(n) = [n]^2$, we say that such a circuit accepts a graph G with the set of vertices V of size n if for some, and hence every, bijection $f : [n] \to V$ it holds that $C(\mathbf{a}) = 1$, where $\mathbf{a}$ is the vector that describes the image of G under f^{-1}, as in the previous section.

III. From Circuits to LPs

In this section we prove the half of the equivalence that takes symmetric circuits with threshold gates into symmetric LPs. That is:

Lemma 1. *If $\mathscr{C}$ is a class of L-structures that is recognized by a family of L-symmetric Boolean threshold circuits of size $s(n)$, then $\mathscr{C}$ is recognized by a family of L-symmetric LP lifts of size $s(n)^{O(1)}$. In addition, if the Boolean circuits do not have threshold gates, then the size of the LP lifts is $O(s(n))$.*

The main step in the construction is the simulation of the threshold gates. The naïve approach by which each threshold gate is replaced by an equivalent AND-OR-NOT circuit will not work: it is known that any symmetric such circuit that computes the majority function must have superpolynomial size. This follows from Theorem 2 in [3] and a standard Ehrenfeucht-Fraissé argument. We need an alternative approach. As a step towards our goal, first we need to generalize the so-called Parity Polytope from Yannakakis [1].

A. The truncated parity polytope

Yannakakis gives a polynomial-size symmetric polytope lift of the parity polytope PP_n defined as the convex hull of all binary strings of length n with an odd number of ones. We adapt the construction to what we call the truncated parity polytope.

For each $n \geq 1$ and $t \in [n]$, let $\mathrm{EX}(n,t)$ be the convex hull of all strings in $\{0,1\}^n$ with exactly t ones. By a standard $\pm\epsilon$ argument, or a standard dimension argument, one can see that $\mathrm{EX}(n,t)$ is defined by its direct LP relaxation: $\sum_{k=1}^n x_k = t$ with $0 \leq x_k \leq 1$ for each $k \in [n]$. For each integer $n \geq 1$, let $|n|$ be the number of bits it takes to write n in binary. For each pair of integers $n \geq 1$ and $q \in \{0, \ldots, |n|-1\}$ let $\mathrm{PP}(n,q)$ be the convex hull of all strings $\mathbf{x} \in \{0,1\}^n$ satisfying $\lfloor 2^{-q} \sum_{k=1}^n x_k \rfloor \equiv 1 \pmod 2$. Note that a vector $\mathbf{x} \in \mathbb{R}^n$ is in $\mathrm{PP}(n,q)$ if, and only if, $\mathbf{x} = \sum_{t,r} w_{t,r} \mathbf{y}_{t,r}$ where each vector $\mathbf{y}_{t,r}$ is in $\mathrm{EX}(n, 2^q(2t+1)+r)$, and the $w_{t,r}$ are non-negative coefficients that add up to one, with (t,r) ranging over the set of pairs of integers with $t \in \{0, \ldots, \lfloor n/2^{q+1} \rfloor\}$ and $r \in \{0, \ldots, 2^q - 1\}$. We introduce variables $w_{t,r}$ and $z_{t,r,i}$ for each $(t,r) \in T$ and each $i \in N$, where $T = \{0, \ldots, \lfloor n/2^{q+1} \rfloor\} \times \{0, \ldots, 2^q - 1\}$ and $N = \{1, \ldots, n\}$, with the intention that $z_{t,r,i} = w_{t,r} y_{t,r,i}$ for appropriate values $y_{t,r,i}$ that we do not care to actually get. The linear program that achieves this is the following:

$$\begin{array}{ll} \sum_{(t,r)\in T} w_{t,r} = 1 & \\ 0 \le w_{t,r} \le 1 & (t,r) \in T \\ \sum_{(t,r)\in T} z_{t,r,i} = x_i & i \in N \\ \sum_{i\in N} z_{t,r,i} = (2^q(2t+1)+r)w_{t,r} & (t,r) \in T \\ 0 \le z_{t,r,i} \le w_{t,r} & t \in T, i \in N \end{array}$$

The symmetry of this linear program with respect to the x-variables is obvious: given $\pi \in \mathrm{Sym}_n$, let σ map $z_{t,r,i}$ to $z_{t,r,\pi(i)}$ and leave every other variable in place. The polytope $\mathrm{PP}(n,q)$ has the following interesting feature:

Claim: If $x_1, \dots, x_{n-1} \in \{0,1\}$ and $\sum_{k=1}^{n-1} x_k \equiv -1 \pmod{2^q}$, then there exists a unique x_n in $\mathbb{R}$ such that $(x_1, \dots, x_n)$ is in $\mathrm{PP}(n,q)$, and moreover x_n is the unique bit in $\{0,1\}$ that makes the truncation $\lfloor 2^{-q} \sum_{k=1}^{n} x_k \rfloor$ odd.

For the existence just take $x_n \in \{0,1\}$ so that the truncation $\lfloor 2^{-q} \sum_{k=1}^{n} x_k \rfloor$ is odd, which must exist by the assumption that $\sum_{k=1}^{n-1} x_k \equiv -1 \pmod{2^q}$. The uniqueness follows once we show that every x_n for which the extension vector $(x_1, \dots, x_n)$ belongs to $\mathrm{PP}(n,q)$ is in $\{0,1\}$, and again the assumption that $\sum_{k=1}^{n-1} x_k \equiv -1 \pmod{2^q}$. For a proof that such an x_n is in $\{0,1\}$ it suffices to show that if $(x_1, \dots, x_n) \in \mathrm{PP}(n,q)$ satisfies the conditions, then it is an extremal point. If it were not an extremal point then it would be a non-trivial combination of at least two extremal points and, whenever $x_1, \dots, x_{n-1}$ are all in $\{0,1\}$, only two candidates remain: $(x_1, \dots, x_{n-1}, 0)$ and $(x_1, \dots, x_{n-1}, 1)$; otherwise some x_i with $1 \le i \le n-1$ would be strictly between 0 and 1. However, assuming that $\sum_{k=1}^{n-1} x_k \equiv -1 \pmod{2^q}$, at least one of these extremal points must have even truncation $\lfloor 2^{-q} \sum_{k=1}^{n} x_k \rfloor$, and hence not even belong to $\mathrm{PP}(n,q)$; a contradiction.

B. Counting gates

The goal in this subsection is to write a polynomial-size symmetric linear program that can be used to simulate the truth-table of an exact counting gate $\mathrm{EX}_{n,t}(x_1, \dots, x_n)$, which outputs 1 if the sum of the n input bits is exactly t, and 0 otherwise. We use the truncated parity polytopes to compute the bits of the binary representation of $\sum_{k=1}^{n} x_i$, and then compare the result with the bits of the binary representation of t.

First consider the following sequence of linear programs which depend only on n and not on t:

$$(\mathbf{x}, 1^{(2^q)}, z_1^{(1)}, \dots, z_q^{(2^{q-1})}, \bar{z}_{q+1}) \in \mathrm{PP}(n + 2^{q+1}, q)$$

for $q = 0, \dots, |n| - 1$, where $\bar{z}_{q+1} = 1 - z_{q+1}$ and, for $\ell \ge 1$, the notation $a^{(\ell)}$ denotes the string $a, a, \dots, a$ of length ℓ. We claim the following property:

Claim: If $x_1, \dots, x_n \in \{0,1\}$, then there is a unique vector $(z_1, \dots, z_{|n|}) \in \mathbb{R}^{|n|}$ which together with $x_1, \dots, x_n$ is a solution, and in this solution we have $z_k \in \{0,1\}$ for all k, and $\sum_{k=1}^{n} x_k = \sum_{k=1}^{|n|} (1 - z_k) 2^{k-1}$; in other words, $z_1, \dots, z_{|n|}$ are the flips of the bits of the binary representation of $\sum_{k=1}^{n} x_k$, listed from least to most significant bit.

From now on in this proof, let $X = \sum_{k=1}^{n} x_k$. The first part of the claim follows from the corresponding property of $\mathrm{PP}(n,q)$'s, and induction on q. The second part is proved by showing that $z_1, \dots, z_{q+1}$ are all in $\{0,1\}$ and

$$X \equiv \sum_{k=1}^{q+1} (1 - z_k) 2^{k-1} \pmod{2^{q+1}} \tag{1}$$

also by induction on q. For $q = 0$ the claim follows from $\mathrm{PP}(n+2, 0) = \mathrm{PP}(n+2)$. For larger q it is a matter of putting things together and using the induction hypothesis in the right place.

Now, exact-t counting gates can be expressed using an additional linear program that simulates an AND gate to compare the bits $z_1, \dots, z_{|n|}$ with (the flips of) the bits of the binary representation of t. The LP for the AND gate is described in the next section. We encapsulate the main property of the resulting linear program $\mathrm{EX}_{n,t}(x_1, \dots, x_n, y)$:

Lemma 2. *The linear program* $\mathrm{EX}_{n,t}(x_1, \dots, x_n, y)$ *has size polynomial in* n*, is symmetric with respect to the group of permutations of* $x_1, \dots, x_n, y$ *that fix* y*, and has the following property: If* $x_1, \dots, x_n$ *are all in* $\{0,1\}$*, then there is a unique* $y \in \mathbb{R}$ *such that* $(x_1, \dots, x_n, y)$ *can be extended to a feasible solution, and this* y *is the unique output bit of the corresponding gate evaluated on inputs* $x_1, \dots, x_n$*.*

Proof. The bound on the size follows by inspection. The symmetry with respect to the group of permutations of $x_1, \dots, x_n, y$ that fix y follows from the symmetry claims for the truncated parity polytopes, together with the extension that keeps each z_i-variable in place. For the main property we rely on the claim and the main property of the LP for AND, stated below. □

C. The construction

Let us recall how AND and NOT gates are represented by LPs. Define:

$$\begin{array}{l} \underline{\mathrm{AND}(x_1, \dots, x_n, y)} \\ y \ge \sum_{i=1}^{n} x_i - n + 1 \\ y \le x_i \\ 0 \le x_i \le 1 \\ 0 \le y \le 1. \end{array} \qquad \begin{array}{l} \underline{\mathrm{NOT}(x, y)} \\ y = 1 - x \\ 0 \le x \le 1 \\ 0 \le y \le 1. \\ \end{array}$$

The main properties are summarized:

Lemma 3. *The linear programs* $\mathrm{AND}(x_1,\dots,x_n,y)$ *and* $\mathrm{NOT}(x_1,y)$ *have size linear in* n*, are symmetric with respect to the group of permutations of its variables that fix* y*, and have the following property: If* $x_1,\dots,x_n$ *are all in* $\{0,1\}$*, with* $n=1$ *for* NOT*, then there is a unique* $y\in\mathbb{R}$ *that makes* $(x_1,\dots,x_n,y)$ *feasible, and this* y *is the unique output bit of the corresponding gate evaluated on inputs* $x_1,\dots,x_n$*.*

Proof. For NOT it is obvious. For AND, easy to see. □

We define the conversion from an AND-NOT-TH circuit C to a linear program $\mathrm{LP}(C)$. Let C' be the circuit that results from replacing each k-threshold gate with inputs $y_1,\dots,y_m$ by $\neg\bigwedge_{t=k}^{m}\neg\mathrm{EX}_{m,t}(y_1,\dots,y_m)$, where $\mathrm{EX}_{m,t}(y_1,\dots,y_m)$ denotes an exact counting gate with inputs $y_1,\dots,y_m$. For each gate i in C', let y_i be a variable constrained by the inequalities $0\le y_i\le 1$. Let $\mathrm{G}(y_1,\dots,y_m,z)$ be the LP for a gate of type G, with auxiliary variables not shown. For each gate o in C' add the constraints:

$$\begin{array}{ll}
y_o = x_u & \text{if } o \text{ has input } x_u,\\
\mathrm{G}(y_{i_1},\dots,y_{i_m},y_o) & \text{if } o \text{ is G input } i_1,\dots,i_m,\\
y_o = 1 & \text{if } o \text{ is the output gate.}
\end{array}$$

By Lemmas 2 and 3, all six cases have size polynomial in the number of inputs, hence the total size is polynomial in the size of C'. In case C does not have TH gates, the step for replacing them by EX gates is not done, and all gates are AND, NOT, so the total size is linear.

Lemma 4. *If* U *is a* G*-set and* C *is* G*-symmetric, then* $\mathrm{LP}(C)$ *is* G*-symmetric and recognizes the same subset of* $\{0,1\}^U$ *as* C*.*

Proof. The claim that $\mathrm{LP}(C)$ recognizes the same subset of $\{0,1\}^U$ as C follows from Lemmas 2 and 3. We prove the symmetry. First note that the intermediate circuit C' is also G-symmetric. Now fix $\pi\in G$. Let σ be a permutation of the gates of C' so that the pair (π,σ) leaves C' in place. In particular, for each gate o of C' with inputs $i_1,\dots,i_m$, if $p=\sigma(o)$, then p is the same type of gate as o, has the same fan-in m, and if o is an input gate fed by x_u, then p is an input gate fed by $x_{\pi(u)}$. Moreover, if $j_1,\dots,j_m$ are the inputs of gate p, then there is a permutation $\tau_o\in\mathrm{Sym}_m$ so that $\sigma(i_k)=j_{\tau_o(k)}$ for every $k\in[m]$. Now use the symmetry claims of the gate programs to extend the permutation to an automorphism gate by gate. □

Proof of Lemma 1. Fix n, let $U=L(n)$, let $G=\mathrm{Sym}_n$ with the natural action on $L(n)$, and use $\mathrm{LP}(C_n)$. □

IV. From LPs to Logic

By C^k we denote the k-variable fragment of counting logic. A function $s(n)$ is at most weakly exponential if there is $\epsilon>0$ such that $s(n)\le 2^{n^{1-\epsilon}}$ for each sufficiently large n. In this section we establish the translation which takes families of symmetric linear programs to families of formulas of counting logic. That is:

Lemma 5. *If* $\mathscr{C}$ *is a class of* L*-structures that is recognized by a family of* L*-symmetric polytope lifts of size* $s(n)$*, then* $\mathscr{C}$ *is recognized by a family of* $\mathrm{C}^{k(n)}$ *formulas, where* $k(n)=O(\log(s(n))/(\log(n)-\log\log(s(n))))$*. Moreover, if* $s(n)$ *is at most weakly exponential, then the formulas have size* $s(n)^{O(1)}$*.*

Together with the standard translation of formulas to circuits, this establishes the second half of the equivalence between symmetric linear programs and symmetric circuits.

Consider a family $(P_n)_{n\in\mathbb{N}}$ of L-symmetric LP lifts. Subsection IV-A below implies that from each P_n one can construct a polytope lift $\widehat{P}_n$ which recognises the same property of structures with n-element domains but comes with an action of the group Sym_n witnessing its symmetry. Further, in Subsection IV-B we prove that the action of Sym_n on each of the constraints and auxiliary variables of $\widehat{P}_n$ depends on a subset of $[n]$ of bounded size called its *support*. This implies, in Subsection IV-C, that each $\widehat{P}_n$ after a small modification becomes a *manageable* LP lift $\bar{P}_n$, that is, one whose auxiliary variables and constraints are indexed by tuples of integers from $[n]$ of bounded length. Finally, in Subsection IV-D based on $\bar{P}_n$ we construct a FOC-interpretation that given an L-structure $\mathbb{A}$ over an n-element domain outputs a linear program which has a solution if and only if $\mathbb{A}$ belongs to the class of interest. Since solving linear programs is expressible in FPC [5], we can conclude the proof.

A. Rigid polytopes

In this subsection we consider general G-symmetric LPs, i.e., not necessarily L-symmetric.

Let U be a G-set and let $P\subseteq\mathbb{R}^U\times\mathbb{R}^W$ be a G-symmetric polytope given by a sequence of linear constraints $(\gamma_v)_{v\in V}$ where each γ_v is of the form $\mathbf{a}^{\mathrm{T}}\mathbf{x}+\mathbf{b}^{\mathrm{T}}\mathbf{y}\le c$, with $\mathbf{x}=(x_u)_{u\in U}$ and $\mathbf{y}=(y_w)_{w\in W}$. We say that the polytope P is *rigid* if for every $\pi\in G$ there exists a unique element of Sym_W, let us denote it by σ_π, such that $P^{(\pi,\sigma_\pi)}=P$.

If P is rigid, then the mapping from G to Sym_W given by $\pi\mapsto\sigma_\pi$ is a group homomorphism. Hence, there is an action of the group G on the set of auxiliary variables $\{y_w\}_{w\in W}$ such that for any $\pi\in G$ and $w\in W$ applying π to y_w gives $y_{\sigma_\pi(w)}$, and the polytope P is G-symmetric with respect to this action. Moreover, this induces an action of G on the set of constraints $\{\gamma_v\}_{v\in V}$ in the obvious way: for any $\pi\in G$ and $v\in V$ applying π to γ_v of the form $\mathbf{a}^{\mathrm{T}}\mathbf{x}+\mathbf{b}^{\mathrm{T}}\mathbf{y}\le c$ gives $\mathbf{a}^{\mathrm{T}}\mathbf{x}^\pi+\mathbf{b}^{\mathrm{T}}\mathbf{y}^{\sigma_\pi}\le c$, and the symmetry of P guarantees that this is also a constraint. For rigid G-symmetric polytopes, we write $\mathbf{y}^{\pi}$ to mean $\mathbf{y}^{\sigma_\pi}$, we use γ_v^{π} to denote $\mathbf{a}^{\mathrm{T}}\mathbf{x}^{\pi} + \mathbf{b}^{\mathrm{T}}\mathbf{y}^{\sigma_\pi} \leq c$, and P^{π} to denote $P^{(\pi,\sigma_\pi)}$.

Suppose that a subset A of $\{0,1\}^U$ is recognised by a G-symmetric polytope lift P. We show that there exists a rigid G-symmetric polytope lift $\widehat{P}$ of size polynomial in the size of P recognising A.

The construction of $\widehat{P}$ goes as follows. For the subgroup of Sym_W consisting of all permutations σ such that $P^{(\mathrm{id},\sigma)} = P$, consider the orbits of the set of auxiliary variables $\{y_w\}_{w \in W}$ under the action of this subgroup. By identifying the variables in each of those orbits we obtain a new G-symmetric polytope lift recognising A with potentially smaller number of auxiliary variables. This procedure needs to be iterated until the polytope is rigid. The number of variables and constraints in the obtained polytope $\widehat{P}$ is at most the corresponding number in P. Moreover, each coefficient in $\widehat{P}$ is the sum of distinct coefficients in P. The conclusion is the following:

Lemma 6. *For every G-symmetric polytope of size s, there is a rigid G-symmetric polytope of size not more than $s\log(s)$ which recognises the same set.*

B. Bounded supports

For a Sym_n-set Y, a subset S of $[n]$ is said to be a *support* of an element $y \in Y$ if for every $\pi \in \mathrm{Sym}_n$ that fixes S pointwise, it holds that $\pi \cdot y = y$. And it is said to be an *even support* of $y \in Y$ if for every $\pi \in \mathrm{Alt}_n$ that fixes S pointwise, we have $\pi \cdot y = y$.

An (even) support S is *k-bounded* if $|S| \leq k$. A Sym_n-set Y is *k-supported* if each element of Y has a k-bounded support. An L-symmetric polytope P is *k-supported* if the set of auxiliary variables and the set of constraints of P are k-supported. We show the following:

Lemma 7. *There exists a positive integer n_0 such that for any positive integers s and n satisfying $s \geq n \geq n_0$, the following holds: If P is a rigid L-symmetric LP lift of size s for structures with n elements, then P is k-supported, where $k = O(\log(s)/(\log(n) - \log\log(s)))$. Moreover, if $s \leq 2^{n/3}$, then the size of P is at most n^k.*

Proof. For simplicity we give the proof for the case of graphs. The general case is completely analogous.

Consider a rigid graph-symmetric polytope lift $P \subseteq \mathbb{R}^{[n]^2} \times \mathbb{R}^W$ of size s given by a sequence of linear constraints $(\gamma_v)_{v \in V}$ of the form $\mathbf{a}^{\mathrm{T}}\mathbf{x} + \mathbf{b}^{\mathrm{T}}\mathbf{y} \leq c$.

If $s > 2^{n/3}$, we can take $k = n$. Indeed, in this case $\log(s(n))/(\log(n) - \log\log(s(n)) \geq n/3\log(3)$. Hence, $n = O(\log(s)/(\log(n) - \log\log(s)))$. Since any element of a Sym_n-set is supported by $[n]$, each auxiliary variable and constraint of P has an n-bounded support.

In the case $s \leq 2^{n/3}$ the argument is more involved. First we obtain bounded even supports. Take $t = \log(s)/(\log(n) - \log\log(s))$ and $k = \lceil t \rceil$. Observe that the denominator in the definition of t is non-zero, since $s \leq 2^{n/3} < 2^n$. Also, we have $0 < t \leq k \leq n/3\log(3) < n/4 < n/e$, to be used later in the proof. Let us start by noting that $t\log\left(\frac{n}{t}\right) > \log(s)$, which follows from the fact that expanding $t\log\left(\frac{n}{t}\right)$ results in $\log(s)\frac{\log(n) - \log\log(s) + \log(\log(n) - \log\log(s))}{\log(n) - \log\log(s)}$ and since $s \leq 2^{n/3}$, the big fraction is strictly bigger than 1.

For any $S \subseteq [n]$, let $\mathrm{Alt}_{(S)}$ denote the group of all even permutations of $[n]$ that fix the set S pointwise. We use the following fact.

Lemma 8 (Theorem 5.2B in [12])**.** *If $n > 8$ and $1 \leq k \leq n/4$, and $G \leq \mathrm{Sym}_n$ such that $[\mathrm{Sym}_n : G] < \binom{n}{k}$, then there is $S \subseteq [n]$ with $|S| < k$ such that $\mathrm{Alt}_{(S)} \leq G$.*

For $w \in W$, let St_w denote the stabilizer of y_w in Sym_n, i.e., the subgroup of Sym_n defined by $\mathrm{St}_w = \{\pi \in \mathrm{Sym}_n : \pi \cdot y_w = y_w\}$. Since $[\mathrm{Sym}_n : \mathrm{St}_w]$ is the size of the orbit of y_w under the action of Sym_n and the total number of auxiliary variables is bounded by the size of P, we have $[\mathrm{Sym}_n : \mathrm{St}_i] \leq s < (n/t)^t \leq (n/k)^k \leq \binom{n}{k}$ with the second following from $t\log\left(\frac{n}{t}\right) > \log(s)$ showed above, and the third from $0 < t \leq k < n/4 < n/e$ and the fact that $f(x) = (n/x)^x$ is an increasing function of x in the interval $(0, n/e)$. Lemma 8 implies that, if n is large enough, there exist $S \subseteq [n]$ with $|S| < k$ and $\mathrm{Alt}_{(S)} \leq \mathrm{St}_w$. This is a k-bounded even support of y_w. An analogous argument yields a k-bounded even support for each constraint in $\{\gamma_v\}_{v \in V}$.

To obtain supports in place of even supports we look at polytopes as graphs. We define a graph called the *graph representation* of P. Its automorphism group is isomorphic to the automorphism group of P and its number of vertices is bounded by $O(s^2)$.

Now, for any $S \subseteq [n]$, let $\mathrm{Sym}_{(S)}$ denote the group of all permutations of $[n]$ that fix the set S pointwise. Take some $w \in W$ and let S be a k-bounded even support of y_w. Since $\mathrm{Alt}_{(S)} \leq \mathrm{St}_w$, we have $\mathrm{Alt}_{(S)} \leq \mathrm{St}_w \cap \mathrm{Sym}_{(S)} \leq \mathrm{Sym}_{(S)}$. Hence, $\mathrm{St}_w \cap \mathrm{Sym}_{(S)} = \mathrm{Alt}_{(S)}$ or $\mathrm{St}_w \cap \mathrm{Sym}_{(S)} = \mathrm{Sym}_{(S)}$. We argue it is the latter case that holds using the following theorem.

Lemma 9 (Theorem A in [13])**.** *If $n > 22$, then the number of vertices of any graph whose full automorphism group is isomorphic to Alt_n is at least $1/2\binom{n}{\lfloor n/2 \rfloor} \sim 2^n/\sqrt{2\pi n}$.*

Assume that $\mathrm{St}_w \cap \mathrm{Sym}_{(S)} = \mathrm{Alt}_{(S)}$. A simple modification of the graph representation $\mathbb{P}$ yields a graph $\mathbb{P}_w$ whose automorphism group is isomorphic to $\mathrm{St}_w \cap \mathrm{Sym}_{(S)}$, and therefore to $\mathrm{Alt}_{(S)}$, which in turn is isomorphic to the alternating group on $[n - |S|]$. Once again, the number of vertices of $\mathbb{P}_w$ is $O(s^2)$. Thus, if n is large enough, we have $s^2 \leq 2^{2n/3} < 1/2\binom{n}{\lfloor n/2 \rfloor}$. Hence, by Lemma 9, we obtain the desired contradiction.

$\mathrm{St}_w \cap \mathrm{Sym}_{(S)} = \mathrm{Sym}_{(S)}$ implies $\mathrm{Sym}_{(S)} \leq \mathrm{St}_w$, thus S is a k-bounded support of y_w. An analogous argument yields a k-bounded support for each constraint in $\{\gamma_v\}_{v \in V}$. Note also that $s < \binom{n}{k} \leq n^k$. This finishes the proof of Lemma 7. □

We now show that it is possible to (non-uniquely) represent the auxiliary variables and constraints of k-supported polytopes by tuples of integers from $[n]$ of length k in a way that is consistent with the group action. In order for the representation to be uniform across all n, we extend the definition of the set $[n]^{(k)}$ to the case when $k > n$. For $0 < n < k$, the set $[n]^{(k)}$ consists of k-tuples of elements of $[n]$ with the first n components pairwise distinct and the last $k - n$ components equal to the n-th component. We have the following:

Lemma 10. *If Y is a single-orbit k-supported Sym_n-set, there is a surjective homomorphism from $[n]^{(k)}$ to Y.*

Once a surjective homomorphism f from a Sym_n-set $[n]^{(k)}$ to a Sym_n-set Y is fixed, the family $\{f^{-1}(y)\}_{y \in Y}$ forms a partition of $[n]^{(k)}$. Hence, for any $y \in Y$, each tuple $(i_1, \ldots, i_k)$ from $f^{-1}(y)$ uniquely identifies y, and is called an *identifier* of y. In most cases each element of Y has several identifiers.

To represent elements of a k-supported Sym_n-sets with potentially more than one orbit, we need to introduce several copies of the set $[n]^{(k)}$, one for each orbit.

Corollary 1. *Let Y be a k-supported Sym_n-set. There is a surjective homomorphism from $Q \times [n]^k$ to Y, where the size of Q is equal to the number of orbits of Y.*

The definition of an identifier extends to the general case discussed in the corollary above. Note that if a tuple $(q, i_1, \ldots, i_k)$ is an identifier of $y \in Y$, then the tuple $(q, \pi(i_1), \ldots, \pi(i_k))$ is an identifier of $\pi \cdot y$.

C. Manageable polytopes

For $k \geq 0$, a polytope $P \subseteq \mathbb{R}^{L(n)} \times \mathbb{R}^W$ is called k-*manageable* if: (1) the set of constraints of P is indexed by $V = Q \times [n]^k$, (2) the set of auxiliary variables of P is indexed by $W = T \times [n]^k$, (3) the sets Q and T come with the trivial action of the group Sym_n, (4) P is L-symmetric with respect to the natural action of Sym_n on W, and the induced action of Sym_n on the set of constraints is exactly the natural action of Sym_n on V.

The following key property of k-manageable polytopes allows us to use them in the translation from families of linear programs to logic.

Lemma 11. *If P is a k-manageable polytope with constraints indexed by $Q \times [n]^{(k)}$ and auxiliary variables indexed by $T \times [n]^{(k)}$, then for any relation symbol $R \in L$, and each $q \in Q$, $t \in T$, $\mathbf{i}, \mathbf{i}', \mathbf{j}, \mathbf{j}' \in [n]^{(k)}$, $\mathbf{k}, \mathbf{k}' \in [n]^{\mathrm{ar}(R)}$: (1) the constant terms of the linear constraints $\gamma_{(q,\mathbf{i})}$ and $\gamma_{(q,\mathbf{i}')}$ are the same, (2) if the equality types of the tuples $(\mathbf{j}, \mathbf{i})$ and $(\mathbf{j}', \mathbf{i}')$ are the same, then the coefficient of the variable $y_{(t,\mathbf{j})}$ in the linear constraint $\gamma_{(q,\mathbf{i})}$ is the same as the coefficient of the variable $y_{(t,\mathbf{j}')}$ in the linear constraint $\gamma_{(q,\mathbf{i}')}$, (3) if the equality types of the tuples $(\mathbf{k}, \mathbf{i})$ and $(\mathbf{k}', \mathbf{i}')$ are the same, then the coefficient of the variable $x_{(R,\mathbf{k})}$ in the linear constraint $\gamma_{(q,\mathbf{i})}$ is the same as the coefficient of the variable $x_{(R,\mathbf{k}')}$ in the linear constraint $\gamma_{(q,\mathbf{i}')}$.*

Now, suppose that a k-supported rigid L-symmetric LP lift $P \subseteq \mathbb{R}^{L(n)} \times \mathbb{R}^W$ recognizes some property of L-structures. We argue that there exists a k-manageable polytope lift $\bar{P}$ recognising A. Since the polytope P is k-supported, by applying Lemma 10 we obtain two sets of identifiers: $Q \times [n]^k$ for the constraints and $T \times [n]^k$ for the auxiliary variables. Let us introduce a new variable of the form $y_{(t,\mathbf{j})}$ for any identifier $(t, \mathbf{j}) \in T \times [n]^k$. We obtain a manageable polytope $\bar{P}$ from P by first, replacing each auxiliary variable y_w by the sum of variables $y_{(t,\mathbf{j})}$ over the set of all identifiers $(t, \mathbf{j})$ of y_w; and secondly, replacing every constraint γ_v, by several copies of this constraint, one for every identifier $(q, \mathbf{i})$ of γ_v. The obtained polytope lift $\bar{P}$ is clearly k-manageable and it is easy to see that it recognizes the same property of L-structures.

D. From manageable polytopes to counting logic

We now put everything together in the proof of Lemma 5. For simplicity we give the proof for the case of graphs. The general case is completely analogous.

Let $P \subseteq \mathbb{R}^{[n]^2} \times \mathbb{R}^W$ be a graph-symmetric LP lift of size s recognising some property of graphs with n vertices, that is, a subset A of $\{0,1\}^{[n]^2}$, and let $\widehat{P}$ be a rigid graph-symmetric LP lift recognising A. Recall that the size s' of $\widehat{P}$ is at most $s \log(s)$ where s is the size of P. In particular, $s' \leq s^2$.

If $s > 2^{n/6}$, we have $n = O(\log(s)/(\log(n) - \log\log(s)))$. Since every class of graphs with n vertices is definable in C^n, we complete the proof of the lemma in this case by taking $k = n$.

If $s \leq 2^{n/6}$, then $s' \leq s^2 \leq 2^{n/3}$. Hence, by Lemma 7, for some $k = O(\log(s')/(\log(n) - \log\log(s')))$, $\widehat{P}$ is k-supported, has at most n^k auxiliary variables, at most n^k constraints, and all its coefficients and constant terms can be encoded using at most n^k bits. Moreover, any such k clearly satisfies $k = O(\log(s)/(\log(n) - \log\log(s)))$.

Let $\bar{P}$ be a k-manageable polytope lift recognising A with the set of constraints indexed by $Q \times [n]^k$, and the set of auxiliary variables indexed by $T \times [n]^k$, as described in Subsection IV-C above. Note that it follows from the construction of $\bar{P}$ that the number of elements in the sets T and Q is bounded, respectively, by the number of auxiliary variables and the number of constraints in $\widehat{P}$. Hence, $|Q|, |T| \leq n^k$.

Suppose now that we are given a graph G with the set of vertices V of size n and the set of edges E. If we could fix a bijection from $[n]$ to V, we could then compute from $\bar{P}$ and G a linear program $\bar{P}_G$ with the set of constraints $\{\gamma_{(q,\mathbf{v})} : q \in Q, \mathbf{v} \in V^k\}$, and the set of variables $\{x_{vw} : v, w \in V\} \cup \{y_{(t,\mathbf{v})} : t \in T, \mathbf{v} \in V^k\}$. In order to decide if G has the property of interest we would then check if the partial valuation $\mathbf{a}_G$ defined by: $x_{vw} := 1$ if $(v, w) \in E$, and $x_{vw} := 0$ otherwise, can be extended to a solution. This in turn can be done in logic using the following consequence of the results in [5].

Lemma 12. *There exists an* FPC *formula ϕ which given a matrix $\mathbf{A} \in \mathbb{Q}^{I \times J}$ and a pair of vectors $\mathbf{b} \in \mathbb{Q}^I$, and $\mathbf{a} \in \mathbb{Q}^{J'}$, where $J' \subseteq J$, decides if $\mathbf{a}$ can be extended to a solution of the linear program $\mathbf{A}\mathbf{x} \leq \mathbf{b}$.*

Our goal is to use Lemma 11 to show that the linear program $\bar{P}_G$ can be computed without fixing a bijection from $[n]$ to V. Based on $\bar{P}$ we construct a FOC-interpretation Ψ which takes as input a graph G with n vertices and outputs a relational encoding of the linear program $\bar{P}_G$ together with the partial valuation $\mathbf{a}_G$.

Given a graph G with the set of vertices V of size n and the set of edges E the FOC-interpretation Ψ has access to the domain V of the graph and the naturally ordered number domain $\{0, \ldots, n\}$.

To represent the bit encodings of the coefficients and constant terms in $\bar{P}_G$ and $\mathbf{a}_G$ we use tuples from $[n]^k \subseteq \{0, \ldots, n\}^k$ ordered lexicographically (with respect to the natural order of the number domain). For a fixed k, such an order can be easily defined in first-order logic.

Tuples from $[n]^k \subseteq \{0, \ldots, n\}^k$ are also used to represent elements of Q and T. For simplicity, assume that $|Q| = |T| = n^k$ and let us think about the correspondence between the tuples in $[n]^k$ and the elements of Q and T as being fixed. Recall that the constraints and variables of the linear program $\bar{P}_G$ are indexed by $Q \times V^k$ and $V^2 \cup T \times V^k$, respectively. The relational encoding of $\bar{P}_G$ in the output of Ψ has the set of constraints $[n]^k \times V^k$ and the set of variables $V^2 \cup [n]^k \times V^k$.

The partial valuation $\mathbf{a}_G$ is simply a vector with index set V^2. For any $v, w \in V$, the entry (v, w) of this vector is defined to be 1 if $(v, w) \in E$, and 0 otherwise. Clearly, this can be done in first-order logic.

It remains to argue that the relations defining the coefficients and constant terms of $\bar{P}_G$ are FOC-definable from G. Recall from Lemma 11 that in the linear program $\bar{P}$ for any $(q, \mathbf{i}) \in Q \times [n]^k$ and $(t, \mathbf{j}) \in T \times [n]^k$, the coefficient of the auxiliary variable $y_{(t,\mathbf{j})}$ in the constraint $\gamma_{(q,\mathbf{i})}$ depends on t, q and the equality type of the tuple $(\mathbf{j}, \mathbf{i}) \in [n]^{2k}$. This implies that in the linear program $\bar{P}_G$ for any $(q, \mathbf{v}) \in Q \times V^k$ and $(t, \mathbf{w}) \in T \times V^k$, the coefficient of the variable $y_{(t,\mathbf{w})}$ in the constraint $\gamma_{(q,\mathbf{v})}$ is determined by t, q and the equality type of the tuple $(\mathbf{w}, \mathbf{v}) \in V^{2k}$. Hence, once the correspondence between the tuples in $[n]^k$ and the elements of Q and T is fixed, for any $(\mathbf{s}, \mathbf{v}) \in [n]^k \times V^k$ and $(\mathbf{z}, \mathbf{w}) \in [n]^k \times V^k$, the interpretation Ψ can define the coefficient of the variable $(\mathbf{z}, \mathbf{w})$ in the constraint $(\mathbf{s}, \mathbf{v})$. This follows from the fact that any element of $[n]^k$ as well as an equality type of a $2k$-tuple can be defined in first-order logic. An analogous reasoning shows that all the other coefficients and constant terms of $\bar{P}_G$ are FOC-definable.

The interpretation Ψ has $O(k)$ variables. Its size is polynomial in n^k, in k, and in the number of equality types of $2k$ tuples, that is, polynomial in n^k, k, and $(2k)^{2k}$. Since $k = O(n)$, the size of Ψ is simply $n^{O(k)}$.

By composing Ψ with the FPC formula ϕ from Lemma 12 we obtain an FPC formula ψ which given a graph G with n vertices decides if G has the property of interest. The formula ψ has $l = O(k)$ variables and size $n^{O(k)}$. We translate it into a formula θ of C^{2l} such that ψ is equivalent to θ on all graphs of size at most n and θ is of size polynomial in the size of ψ, l, and n^l. Hence, θ has $O(k)$ variables and size $n^{O(k)}$.

We have therefore shown that a property of graphs with n vertices recognized by a graph-symmetric polytope lift of size s is defined by a C^k formula, where $k = O(\log(s)/(\log(n) - \log\log(s)))$. Moreover, if s is at most weakly exponential, then for some positive real ϵ we have $k = O(\log(s)/(\log(n) - \log\log(s))) = O(\log(s)/(\epsilon \log(n))) = O(\log(s)/\log(n))$. Hence, in this case the size of θ is $n^{O(k)} = s^{O(1)}$. This finishes the proof of Lemma 5 and this section.

V. Results and Applications

In this section, we start by establishing the main theorem of the paper, which characterizes the expressive power of symmetric linear programs. We then derive from it upper and lower bounds on families of symmetric linear programs for many classical combinatorial problems. Finally, we observe that such families are as powerful as general Boolean circuits on almost all graphs and relate this to work on the planted clique problem.

A. Equivalence of Models

If $\mathscr{C}$ is a class of finite L-structures of some single-sorted vocabulary L, and n is a positive integer, we write $\mathscr{C}_n$ for the set of all structures in $\mathscr{C}$ of cardinality n. We write $\mathrm{s}_{\mathscr{C}}(n)$ for the size of a smallest L-symmetric Boolean circuit that recognizes $\mathscr{C}_n$, and $\mathrm{lp}_{\mathscr{C}}(n)$ for the size of a smallest L-symmetric LP lift that recognizes $\mathscr{C}_n$. Similarly, we write $\mathrm{w}_{\mathscr{C}}(n)$ for the *counting-width* of $\mathscr{C}_n$, i.e., the smallest number of variables k of a C^k-formula that defines $\mathscr{C}_n$ on L-structures of cardinality n, and $\mathrm{sw}_{\mathscr{C}}(n)$ for the *counting size-width* of $\mathscr{C}_n$, i.e., the smallest k such that there is a C^k-formula of size at most n^k that defines $\mathscr{C}_n$ on L-structures of cardinality n. As motivation for the $\mathrm{sw}_{\mathscr{C}}$ measure, observe that $\mathrm{w}_{\mathscr{C}}(n) \leq \mathrm{sw}_{\mathscr{C}}(n)$ and that, unlike for $\mathrm{w}_{\mathscr{C}}$, bounded $\mathrm{sw}_{\mathscr{C}}$ implies polynomial size.

Theorem 1. *If $\mathscr{C}$ is a class of finite L-structures, $\mathrm{lp}_{\mathscr{C}}(n)$ is at most weakly exponential, and $\mathrm{t}_{\mathscr{C}}(n) := \log(\mathrm{lp}_{\mathscr{C}}(n))/\log(n)$, then*

1) $\mathrm{s}_{\mathscr{C}}(n)^{\Omega(1)} \leq \mathrm{lp}_{\mathscr{C}}(n) \leq \mathrm{s}_{\mathscr{C}}(n)^{O(1)}$,
2) $\Omega(\mathrm{sw}_{\mathscr{C}}(n)) \leq \mathrm{t}_{\mathscr{C}}(n) \leq O(\mathrm{sw}_{\mathscr{C}}(n))$.

Proof. The upper bound in 1) is a direct consequence of Lemma 1. The upper bound in 2) follows: $\mathrm{t}_{\mathscr{C}}(n) \leq O(\log(\mathrm{s}_{\mathscr{C}}(n))/\log(n)) \leq O(\mathrm{sw}_{\mathscr{C}}(n))$ since $\mathrm{s}_{\mathscr{C}}(n) \leq n^{O(\mathrm{sw}_{\mathscr{C}}(n))}$ (see [11]). The lower bounds in 1) and 2) follow from Lemma 5: Write $s = \mathrm{lp}_{\mathscr{C}}(n)$ and choose $\ell = c\log(s)/(\log(n) - \log\log(s))$ for a large c to be specified later. By assumption $s \leq 2^{n^{1-\epsilon}}$ for some $\epsilon > 0$ and large enough n. Hence $\ell = O(\log(s)/\log(n))$ with the hidden constant in the big-oh notation dependent on ϵ. For the appropriate constant in the big-oh in $\ell = O(\log(s)/\log(n))$, Lemma 5 says that there is a C^ℓ-formula that defines $\mathscr{C}$ and has size polynomial in s, since $\mathrm{lp}_{\mathscr{C}}(n)$ is at most weakly exponential. If c is big enough we get that $s^{O(1)} \leq n^\ell$, so $\mathrm{sw}_{\mathscr{C}}(n) \leq O(\log(s)/\log(n))$. Hence $\Omega(\mathrm{sw}_{\mathscr{C}}(n)) \leq \mathrm{t}_{\mathscr{C}}(n)$ and $\mathrm{s}_{\mathscr{C}}(n)^{\Omega(1)} \leq \mathrm{lp}_{\mathrm{C}}(n)$ through, again, $\mathrm{s}_{\mathscr{C}}(n) \leq n^{O(\mathrm{sw}_{\mathscr{C}}(n))}$ (see [11]). □

B. Upper and Lower Bounds

By using the strongest known lower bounds on counting width, Lemmas 1 and 5 provide exponential lower bounds of type $2^{\Omega(n)}$. We discuss these here.

a) Lower bounds on symmetric lifts and circuits: In the sequel, let 3-XOR refer to the constraint satisfaction problem of deciding whether a system of 3-variable parity constraints on $\{0,1\}$-valued variables is satisfiable, and let 3-SAT refer to the satisfiability problem for 3-CNF formulas. In both cases, an instance is presented as a finite structure that encodes the incidence structure of the constraints: the domain is the disjoint union of the set of variables and the set of constraints, there is one monadic relation for each type of constraint that indicates which constraints are of that type, and three binary relations that indicate the three variables that participate in each constraint. Note that the instances for these problems are not plain graphs but graphs with coloured vertices and edges. It is understood that, when the graphs are coloured, the term *graph-symmetric* refers to symmetry with respect to the expanded vocabulary.

Theorem 2. *Every graph-symmetric LP lift or Boolean threshold circuit that recognizes the class of Hamiltonian graphs with n vertices, or the class of 3-colourable graphs with n vertices, or the class of satisfiable 3-SAT instances with n variables, or the class of satisfiable 3-XOR instances with n variables, has size $2^{\Omega(n)}$. Moreover, for 3-colouring, 3-SAT, and 3-XOR, the lower bound holds even on the class of instances with $O(n)$ edges, $O(n)$ clauses, and $O(n)$ constraints, respectively.*

We note that these $2^{\Omega(n)}$ lower bounds for 3-colouring, 3-XOR and 3-SAT are optimal up to the multiplicative constant in the exponent: there are symmetric Boolean circuits and LP lifts of size $2^{O(n)}$. This follows from their definability in Monadic Second-Order Logic: the monadic second-order quantifers translate into conjunctions or disjunctions of size $2^{O(n)}$, and the first-order parts translate into polynomial-size symmetric circuits in the usual way.

By Lemma 5, for obtaining the $2^{\Omega(n)}$ lower bound for LP lifts it suffices to show that any C^k-sentence that defines the class of n-vertex 3-colourable graphs has $k = \Omega(n)$: indeed, whenever $s \leq 2^{n/d}$, we have $\log(s)/(\log(n) - \log\log(s)) \leq n/(d\log(d))$. With the lower bound for LP lifts in hand, the lower bound for Boolean threshold circuits would then follow from Lemma 1. A result from the literature that is quite close to the $k = \Omega(n)$ that we need can be found in Section 4.2 in [14], but the analysis in there gives $k = \Omega(\sqrt{n})$, and not $k = \Omega(n)$. While it should be possible to modify the construction in [14] to get what we need, we refer to a more recent construction that achieves what we want for the problems 3-XOR and 3-SAT, and then proceed by reduction. These intermediate steps will also be useful when we discuss Hamiltonicity.

Theorem 3 (see Theorem 3.7 and 3.8 in [15] and Lemmas 22 and 23 in [16])**.** *There exist $c, d > 0$ such that, for every k and every sufficiently large n, every C^k-sentence that separates the class of satisfiable 3-XOR (resp. 3-SAT) instances with n variables and cn constraints from the class of unsatisfiable ones has $k \geq dn$.*

Neither [15] nor [16] state the linear bound cn on the number of constraints, but it easily follow from both proofs. Concretely, it follows from Lemma 3.3 in [15], in which the bound *is* stated. It is easy to see that the textbook (e.g., [17]) reduction from 3-SAT to 3-colouring is a first-order interpretation that produces a linear size output. The textbook reduction from 3-SAT to Hamiltonicity is not so, but can easily be converted into one. Combined with Theorem 3 we get Theorem 2.

b) Lower Bound on the TSP Polytope: Yannakakis proved that the travelling salesman polytope does not have subexponential symmetric LP lifts. We show that the same follows from Theorem 2. Let TSP_n denote the convex hull of all the vectors $(x_{ij})_{i,j\in[n]}$ that represent Hamilton cycles on n vertices.

Theorem 4 (Theorem 2 in [1])**.** *Every graph-symmetric LP lift that has* TSP_n *as shadow has size* $2^{\Omega(n)}$*.*

Proof. If P were such a lift, with principal and auxiliary variables $\mathbf{y}$ and $\mathbf{z}$, then the program with constraints $0 \leq y_{ij} \leq x_{ij}$ and $(\mathbf{y}, \mathbf{z}) \in P$ would recognize the class of Hamiltonian graphs and clash with Theorem 2. □

c) Upper bounds: Surprisingly, the type of argument of Theorem 4 cannot be adapted for the matching polytope: Theorem 1 says that any problem that is definable in FPC has polynomial-size symmetric LP-lifts, and graphs that have perfect matchings are definable in FPC [3].

Corollary 2. *There is a (polynomial-time uniform) family of graph-symmetric LP lifts of polynomial size that recognizes the class of graphs with a perfect matching.*

This should be contrasted with the fact, proved by Yannakakis, that any symmetric LP lift of the perfect matching polytope PM_n has size $2^{\Omega(n)}$. Capturing PM_n by an LP lift or recognizing the class of graphs that have a perfect matching by an LP lift are different tasks. Both objects could be used for deciding whether a given graph has a perfect matching, but capturing PM_n has a demanding structural requirement that has no analogue in the other task. We do not know whether there is any route for deriving lower bounds for PM_n from our results.

C. Problems on Erdős-Rényi Random Graphs

Let $G \sim \mathscr{G}(n,p)$ mean that G is distributed as in the Erdős-Rényi distribution on n-vertex labelled graphs with edge probability p. We argue that, for average-case problems with respect to the uniform distribution $\mathscr{G}(n,1/2)$, as well as for the type of problems that ask to distinguish $\mathscr{G}(n,1/2)$ from some other distribution, polynomial-size symmetric LPs are as powerful as arbitrary not necessarily symmetric Boolean circuits. For average-case problems, this is a direct consequence of our main result and the following well-known fact in descriptive complexity theory:

Theorem 5 (Corollary 4.8 in [6])**.** *For every polynomial-time decidable class of graphs* $\mathscr{C}$ *there is an* FPC*-definable class of graphs* $\mathscr{C}'$ *for which the probability that a random graph* $G \sim \mathscr{G}(n,1/2)$ *falls in the symmetric difference* $\mathscr{C} \Delta \mathscr{C}'$ *is* $o(1)$*.*

The point of Theorem 5 is that the FPC formula that defines $\mathscr{C}'$ does not require any order on the input graph, hence our Theorem 1 applies. Theorem 5 is indeed a consequence of the Immerman-Vardi Theorem [18], [19] and the fact that a linear order is, asymptotically almost surely on $\mathscr{G}(n,1/2)$, definable in FPC. We return to this later. For the rest of this section we focus on the problem of distinguishing $\mathscr{G}(n,1/2)$ from some other distribution of random graphs, to which a direct application of Theorem 5 does not look possible.

Let $\mathscr{G}(n,p,k)$ denote the distribution that results from drawing a random graph from $\mathscr{G}(n,p)$ and then *planting* a random k-clique in it, i.e., adding the edges of a k-clique on a uniformly chosen subset of k vertices. Following [20], the planted clique problem, also known as the *hidden* clique problem, comes in three flavours: search, refutation, and decision. We focus on the decision version which, formally, can be stated as follows. We say that $\mathscr{C}$ solves the planted clique problem with parameters $p = p(n)$ and $k = k(n)$ and advantage $\epsilon > 0$ if for every large enough n we have: 1) if $G \sim \mathscr{G}(n,p)$, then G is in $\mathscr{C}$ with probability at least $1/2 + \epsilon$, 2) if $G \sim \mathscr{G}(n,p,k)$, then G is in $\mathscr{C}$ with probability at most $1/2 - \epsilon$. It is solvable in polynomial time if $\mathscr{C}$ is.

The planted clique problem has an interesting history starting at [21], [22]. In the range $k(n) = \Omega(\sqrt{n})$, algorithms were found to solve it in polynomial time [22]–[24]. For $k(n) = o(\sqrt{n})$ the status of the problem is famously open, but lower bounds are known in restricted models, including certain models of (symmetric) linear and semidefinite program formulations. The clique number is the maximum of $\sum_{v\in V} y_v$ subject to the constraints that $y_u y_v = 0$ for each non-edge $(u,v) \notin E$, and $y_v^2 - y_v = 0$ for each $u \in V$. In the decision version we replace the objective by $\sum_{v\in V} y_v \geq k$. Thinking of G as given by the $\{0,1\}$-vector $(x_{uv})_{u,v\in[n]}$ in the usual way, the program can be made uniform for all G by turning the constraint into $y_u y_v \leq x_{uv}$ for all $u, v \in [n]$. This is a hard-to-solve quadratic program, but there are systematic methods for generating tractable relaxations as introduced by Lovász and Schrijver in [25], and Sherali and Adams in [26]. These lead to hierarchies of symmetric LP lifts that project to tighter and tighter approximations of the convex hull of solutions of the quadratic program. The limitations of the LS and SA hierarchies (and beyond) for the planted clique problem have been the object of recent study [8]–[10]

In view of such success in proving lower bounds on the size of symmetric LP lifts, starting with Yannakakis, and including the discussion above on hierarchies for the planted clique problem, and also given our own lower bounds from Section V-B, the following consequence of Theorem 1 may come as a surprise:

Theorem 6. *If the planted clique problem with parameters $p = 1/2$ and $k = k(n)$ is solvable in polynomial time with advantage $\epsilon > 0$, then it is also solvable with advantage $\epsilon - o(1)$ in* FPC*, by polynomial-size graph-symmetric LP lifts, and by polynomial-size graph-symmetric threshold circuits.*

Proof. Using almost sure graph canonization from [27], an order is almost surely definable on $G \sim \mathscr{G}(n, 1/2)$ by a FOC-formula ϕ (see [6]). Let $\Psi(G)$ be "$\phi(G)$ defines an order and $\Phi(G, \phi)$ holds", where $\Phi(G, <)$ is the FP formula given by the Immerman-Vardi Theorem applied to the assumption. Note: Ψ is an FPC formula over unordered graphs. If $G \sim \mathscr{G}(n, 1/2)$, then the probability that ϕ does not define a linear order is $o(1)$, and the probability that some and hence every ordered expansion of G satisfies Φ is at least $1/2 + \epsilon$, so the probability that G satisfies Ψ is at least $1/2 + \epsilon - o(1)$. If $G \sim \mathscr{G}(n, 1/2, k)$, then the probability that some and hence every ordered expansion of G satisfies Φ is at most $1/2 - \epsilon$, so the probability that G satisfies Ψ is even smaller, and $1/2 - \epsilon \leq 1/2 - \epsilon + o(1)$. □

Acknowledgment

First author partially funded by European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme, grant agreement ERC-2014-CoG 648276 (AUTAR) and MICCIN grant TIN2016-76573-C2-1P (TASSAT3). The second author was partially supported by a Fellowship of the Alan Turing Institute under the EPSRC grant EP/N510129/1. The third author funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 795936. Some of the work reported here was initiated at the Simons Institute for the Theory of Computing during the programme on Logical Structures in Computation in autumn 2016. We are grateful to Matthew Anderson for some very stimulating discussions on this topic. In particular, he suggested using Lemma 9 to prove the existence of supports. We also thank Wied Pakusa for sharing his manuscript on the FPC-completeness of linear programming with us.

References

[1] M. Yannakakis, "Expressing combinatorial optimization problems by linear programs," *J. Comput. Syst. Sci.*, vol. 43, no. 3, pp. 441–466, 1991.

[2] T. Rothvoss, "The matching polytope has exponential extension complexity," *J. ACM*, vol. 64, no. 6, pp. 41:1–41:19, 2017.

[3] M. Anderson and A. Dawar, "On symmetric circuits and fixed-point logics," *Theory of Computing Systems*, vol. 60, no. 3, pp. 521–551, Apr 2017.

[4] A. Dawar, "The nature and power of fixed-point logic with counting," *ACM SIGLOG News*, vol. 2, no. 1, pp. 8–21, 2015.

[5] M. Anderson, A. Dawar, and B. Holm, "Solving linear programs without breaking abstractions," *J. ACM*, vol. 62, no. 6, pp. 48:1–48:26, Dec. 2015.

[6] L. Hella, P. G. Kolaitis, and K. Luosto, "Almost everywhere equivalence of logics in finite model theory," *Bulletin of Symbolic Logic*, vol. 2, no. 4, pp. 422–443, 1996.

[7] N. Alon, M. Krivelevich, and B. Sudakov, "Finding a large hidden clique in a random graph," in *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '98. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1998, pp. 594–598.

[8] U. Feige and R. Krauthgamer, "The probable value of the Lovász–Schrijver relaxations for maximum independent set," *SIAM J. Comput.*, vol. 32, no. 2, pp. 345–370, 2003.

[9] B. Barak, S. B. Hopkins, J. A. Kelner, P. Kothari, A. Moitra, and A. Potechin, "A nearly tight sum-of-squares lower bound for the planted clique problem," in *FOCS*. IEEE Computer Society, 2016, pp. 428–437.

[10] S. B. Hopkins, P. Kothari, A. H. Potechin, P. Raghavendra, and T. Schramm, "On the integrality gap of degree-4 sum of squares for planted clique," *ACM Trans. Algorithms*, vol. 14, no. 3, pp. 28:1–28:31, 2018.

[11] M. Otto, *Bounded Variable Logics and Counting: A Study in Finite Models*, ser. Lecture Notes in Logic. Cambridge University Press, 2017, vol. 9.

[12] J. D. Dixon and B. Mortimer, *Permutation Groups*, ser. Graduate Texts in Mathematics. Springer-Verlag New York, 1996, vol. 163.

[13] M. W. Liebeck, "On graphs whose full automorphism group is an alternative group or a finite classical group," *Proceedings of the London Mathematical Society*, vol. s3-47, no. 2, pp. 337–362, 1983.

[14] A. Dawar, "A restricted second order logic for finite structures," *Inf. Comput.*, vol. 143, no. 2, pp. 154–174, 1998.

[15] A. Atserias and A. Dawar, "Definable inapproximability: New challenges for duplicator," in *27th EACSL Annual Conference on Computer Science Logic, CSL 2018, September 4-7, 2018, Birmingham, UK*, 2018, pp. 7:1–7:21.

[16] A. Dawar and P. Wang, "Definability of semidefinite programming and lasserre lower bounds for csps," in *LICS*. IEEE Computer Society, 2017, pp. 1–12.

[17] C. H. Papadimitriou, *Computational complexity*. Academic Internet Publ., 2007.

[18] N. Immerman, "Languages that capture complexity classes," *SIAM J. Comput.*, vol. 16, no. 4, pp. 760–778, 1987.

[19] M. Y. Vardi, "The complexity of relational query languages (extended abstract)," in *STOC*. ACM, 1982, pp. 137–146.

[20] B. Barak and D. Steurer, "Proofs, beliefs, and algorithms through the lens of sum-of-squares," 2016, last accessed Jan 8, 2019.

[21] M. Jerrum, "Large cliques elude the metropolis process," *Random Struct. Algorithms*, vol. 3, no. 4, pp. 347–360, 1992.

[22] L. Kučera, "Expected complexity of graph partitioning problems," *Discrete Applied Mathematics*, vol. 57, no. 2, pp. 193 – 212, 1995, combinatorial optimization 1992.

[23] N. Alon, M. Krivelevich, and B. Sudakov, "Finding a large hidden clique in a random graph," *Random Structures & Algorithms*, vol. 13, no. 34, pp. 457–466, 1998.

[24] U. Feige and R. Krauthgamer, "Finding and certifying a large hidden clique in a semirandom graph," *Random Structures & Algorithms*, vol. 16, no. 2, pp. 195–208, 2000.

[25] L. Lovász and A. Schrijver, "Cones of matrices and set-functions and 0-1 optimization," *SIAM Journal on Optimization*, vol. 1, no. 2, pp. 166–190, 1991.

[26] H. D. Sherali and W. P. Adams, "A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems," *SIAM J. Discrete Math.*, vol. 3, no. 3, pp. 411–430, 1990.

[27] L. Babai, P. Erdös, and S. M. Selkow, "Random graph isomorphism," *SIAM J. Comput.*, vol. 9, no. 3, pp. 628–635, 1980.