Definable Ellipsoid Method, Sums-of-Squares Proofs, and the Isomorphism Problem

Albert Atserias
Universitat Politècnica de Catalunya

Joanna Ochremiak
Université Paris Diderot

Abstract

The ellipsoid method is an algorithm that solves the (weak) feasibility and linear optimization problems for convex sets by making oracle calls to their (weak) separation problem. We observe that the previously known method for showing that this reduction can be done in fixed-point logic with counting (FPC) for linear and semidefinite programs applies to any family of explicitly bounded convex sets. We use this observation to show that the exact feasibility problem for semidefinite programs is expressible in the infinitary version of FPC. As a corollary we get that, for the graph isomorphism problem, the Lasserre/Sums-of-Squares semidefinite programming hierarchy of relaxations collapses to the Sherali-Adams linear programming hierarchy, up to a small loss in the degree.

CCS Concepts • **Theory of computation → Complexity theory and logic**; **Convex optimization**; **Semidefinite programming**; **Proof complexity**; **Finite Model Theory**;

Keywords ellipsoid method, fixed-point logic, semidefinite programming, sums-of-squares, graph isomorphism problem

ACM Reference Format:
Albert Atserias and Joanna Ochremiak. 2018. Definable Ellipsoid Method, Sums-of-Squares Proofs, and the Isomorphism Problem. In *LICS '18: LICS '18: 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, July 9–12, 2018, Oxford, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3209108.3209186

1 Introduction

Besides the well-known fact of being the first algorithm to be discovered that could solve linear programs (LPs) in polynomial time, the ellipsoid method has at least two other aspects that make it an important tool for the computer science theoretician. The first is that the algorithm is able to handle not only high-dimensional explicit LPs, but even certain implicitly given LPs that are described by exponentially many, or even infinitely many, linear inequalities. These include some of the most celebrated groundwork pieces of combinatorial optimization, such as the weighted matching problem on general graphs, and the submodular function minimization problem, among others. The second important feature of the ellipsoid method is that its polynomial running time in the bit-model of computation, taking into account potential issues of numeric instability, is since a long time ago well understood and developed [9].

There is a third emerging and to some extent surprising feature of the ellipsoid method that is of particular significance for the logician and the descriptive complexity theorist. The starting point is the important breakthrough result of Anderson, Dawar and Holm [2] who developed a method called *folding* for dealing with symmetries in an LP. They used this method for showing that, for the special case of LPs, the ellipsoid method can be implemented in fixed-point logic with counting (FPC), and hence in polynomial time, but *choicelessly*, i.e., in a way that the symmetries from the input are respected all along the computation, as well as in the output. As the main application of their result, they proved that the class of graphs that have a perfect matching could be defined in FPC, thus solving one of the well-known open problems raised by Blass, Gurevich and Shelah in their work on Choiceless Polynomial Time [6]. The method of folding was extended further by Dawar and Wang for dealing with explicitly bounded and full-dimensional semidefinite programs (SDPs) [7].

The first contribution of our work is the observation that the abovementioned method of *folding* from [2] is general enough to capture the power of the ellipsoid method in its full strength. We observe that the general polynomial-time reduction that solves the weak feasibility problem given a weak separation oracle for an explicitly bounded convex set can be implemented, choicelessly, in FPC. As in the earlier works that employed the folding method, our implementation uses the reduction algorithm as described in [9] as a black-box. The black-box is made into a choiceless procedure through a sequence of runs of the algorithm along a refining sequence of suitable quotients of the given convex set. It should be pointed out that while all the main ideas for doing this were already implicit in the earlier works [2, 7], working out the details requires a certain degree of care. For one thing, when we started this work it was not clear whether the earlier methods would be able to deal with separation oracles for families of convex sets that are *not* closed under the folding-quotient operations. We observe that such closure conditions, which happen to hold for LPs and SDPs, are indeed not required.

With this observation in hand, we develop three applications.

Our first application concerns the semidefinite programming exact feasibility problem. A semidefinite set, also known as a spectrahedron, is a subset of Euclidean space that is defined as the intersection of the cone of positive semidefinite matrices with an affine subspace. Thus, semidefinite sets are the feasible regions of SDPs, and the SDP exact feasibility problem asks, for an SDP given as input, whether its feasible region is non-empty. While the approximate and explicitly bounded version of this problem is solvable in polynomial-time by the ellipsoid method, the computational complexity of *exact* feasibility is a well-known open problem in mathematical programming; it is decidable in polynomial space, but its precise position in the complexity hierarchy is unknown. It has been shown that the problem is at least as hard as PosSLP,

LICS '18, July 9–12, 2018, Oxford, United Kingdom

ACM ISBN 978-1-4503-5583-4/18/07…$15.00
https://doi.org/10.1145/3209108.3209186

the positivity problem for integers represented as arithmetic circuits [16], and hence at least as hard as the famous square-root sum problem, but the exact complexity of these two problems is also largely unknown (see [1]). Our result on the SDP exact feasibility problem is that, when its input is represented suitably as a finite structure, it is definable in the logic $\mathrm{C}^{\omega}_{\infty\omega}$, i.e. bounded-variable infinitary logic with counting. In more recent terminology, we say that the SDP exact feasibility problem has *bounded counting width*: there is a fixed bound k so that the set of YES (and NO) instances of the problem is closed under indistinguishability by formulas of k-variable counting logic. This is perhaps an unexpected property for the SDP exact feasibility problem to have.

Although this definability result does not seem to have any direct algorithmic consequences for the SDP exact feasibility problem itself, we are able to use the gained knowledge to produce new results on the strength of isomorphism relaxations.

A variety of mathematical programming relaxations of the graph isomorphism problem have been proposed in the literature: from the fractional isomorphism relaxation of Tinhofer [17], through its strengthening via the Sherali-Adams hierarchy of LP relaxations [3, 12], to its further strengthening via the Lasserre hierarchy of SDP relaxations [13], its relaxation via Groebner basis computations [5], and a few others. While all these hierarchies of LP, SDP or Groebner-based relaxations are now known to stay proper relaxations of isomorphism, their relative strength, besides the obvious relationships, was not fully understood. Since SDP is a proper generalization of LP, one may be tempted to guess that the Lasserre SDP hierarchy could perhaps distinguish more graphs than its LP sibling. Interestingly, we prove this not to be the case: for the isomorphism problem, the strength of the Lasserre hierarchy collapses to that of the Sherali-Adams hierarchy, up to a small loss in the level of the hierarchy.

Concretely, we show that there exists a constant $c \geq 1$ such that if two given graphs are distinguishable in the k-th level of the Lasserre hierarchy, then they must also be distinguishable in the ck-th level of the Sherali-Adams hierarchy. The constant c loss comes from the number of variables for expressing the SDP exact feasibility problem in bounded-variable counting logic. It should be noted that our proof is indirect as it relies on the characterization of the indistinguishability in k-variable counting logic via the k-th level Sherali-Adams relaxation of graph isomorphism [3]. The question whether the collapse can be shown to hold by directly *lifting* LP-feasible solutions into SDP-feasible ones remains an interesting one.

By moving to the duals of the Lasserre and the Sherali-Adams hierarchies our results can be interpreted in terms of Sums-of-Squares proofs (SOS) and Sherali-Adams proofs (SA) and, as a side bonus, they can be used to derive consequences for Polynomial Calculus proofs (PC). In terms of proofs, we show that if there is a degree-k SOS proof that two graphs are not isomorphic, then there is also a degree-ck SA proof. In turn, it was already known from before, by combining the results in [3] and [5], that if there is a degree-ck SA proof then there is also a degree-ck (monomial) PC proof (over the reals), which is known to imply that there is a degree-$2ck$ SOS proof by the recent result in [4]. Thus, our result completes a full cycle of implications to show that, for the graph isomorphism problem, SA, monomial PC, PC, and SOS are equally powerful, up to a factor loss of $2c$ in the degree. It also confirms the belief expressed in [5] that the gap between PC and monomial PC is not large (a result obtained independently in [8]). It is remarkable that we proved these statements purely about the relative strength of proof systems through an excursion into the descriptive complexity of the ellipsoid method, the SDP exact feasibility problem, and bounded-variable infinitary logics.

2 Preliminaries

Vectors and matrices. We use $[n]$ to denote the set $\{1, \ldots, n\}$. If I is a non-empty index set, then an I-vector is an element of $\mathbb{R}^I$. The components of $u \in \mathbb{R}^I$ are writen $u(i)$ or u_i, for $i \in I$. We identify $\mathbb{R}^n$ with $\mathbb{R}^{[n]}$. For I-vectors u and v, the *inner product* of u and v is $\langle u, v\rangle = \sum_{i\in I} u_i v_i$. We write $\|u\|_2 = \sqrt{\langle u, u\rangle}$ for the L_2-norm, and $\|u\|_\infty = \max\{|u_i| : i \in I\}$ for the L_∞-norm. For $K \subseteq \mathbb{R}^I$ and $\delta > 0$, we define the *δ-ball* around K by $S(K, \delta) := \{x \in \mathbb{R}^I : \|x - y\|_2 \leq \delta \text{ for some } y \in K\}$. For $K = \{x\}$, we set $S(x, \delta) := S(\{x\}, \delta)$. We define also $S(K, -\delta) := \{x \in \mathbb{R}^I : S(x, \delta) \subseteq K\}$.

If I and J are two non-empty index sets, then an $I \times J$-matrix is simply an $I \times J$-vector; i.e., an element of $\mathbb{R}^{I\times J}$. The L_2- and L_∞-norms of a matrix $X \in \mathbb{R}^{I\times J}$ are defined as the respective norms of X seen as an $I \times J$-vector, and the inner product of the matrices $X, Y \in \mathbb{R}^{I\times J}$ is $\langle X, Y\rangle = \sum_{i\in I}\sum_{j\in J} X_{ij}Y_{ij}$. Matrix product is written by concatenation. A square matrix $X \in \mathbb{R}^{I\times I}$ is positive definite, denoted $X > 0$, if it is symmetric and satisfies $z^T Xz > 0$, for every non-zero $z \in \mathbb{R}^I$. If it is symmetric but satisfies the weaker condition that $z^T Xz \geq 0$, for every $z \in \mathbb{R}^I$, then it is positive semidefinite, which we denote by $X \geq 0$. Equivalently, X is positive semidefinite if and only if $X = Y^T Y$ for some matrix $Y \in \mathbb{R}^{J\times I}$ if and only if all its eigenvalues are non-negative. By I we denote the square identity matrix of appropriate dimensions.

Let I and J be two non-empty index sets and let $\sigma : I \to J$ be a function. If v is a J-vector, then we write $[v]^{-\sigma}$ for the I-vector defined by $[v]^{-\sigma}(i) = v(\sigma(i))$ for every $i \in I$. The notation extends to sets S of J-vectors in the natural way: $[S]^{-\sigma} = \{[v]^{-\sigma} : v \in S\}$. If P is a set of I-vectors and Q is a set of J-vectors, then we say that P and Q are *isomorphic*, denoted $P \cong Q$, if there exists a bijection $\sigma : I \to J$ such that $P = [Q]^{-\sigma}$.

Vocabularies, structures and logics. A many-sorted (relational) vocabulary L is a set of sort symbols $D_1, \ldots, D_s$ together with a set of relation symbols $R_1, \ldots, R_m$. Each relation symbol R in the list has an associated *type* of the form $D_{i_1} \times \cdots \times D_{i_r}$, where $r \geq 0$ is the *arity* of the symbol, and $i_1, \ldots, i_r \in [s]$ are not necessarily distinct. A structure $\mathbb{A}$ of vocabulary L, or an L-structure, is given by s disjoint sets $D_1, \ldots, D_s$ called *domains*, one for each sort symbol $D_i \in L$, and one relation $R \subseteq D_{i_1} \times \cdots \times D_{i_r}$ for each relation symbol $R \in L$ of type $D_{i_1} \times \cdots \times D_{i_r}$. We use $D(\mathbb{A})$ or D to denote the domain associated to the sort symbol D, and $R(\mathbb{A})$ or R to denote the relation associated to the relation symbol R. In practice, the overloading of the notation should never be an issue. The domain of a sort symbol is also called a *sort*.

A logic for a many-sorted vocabulary L has an underlying set of *individual variables* for each different sort in L. When interpreted on an L-structure, the variables are supposed to range over the domain of its sort; i.e., the variables are typed. Besides the equalities $x = y$ between variables of the same type, the atomic L-formulas are the formulas of the form $R(x_1, \ldots, x_r)$, where R is a relation symbol of arity r and $x_1, \ldots, x_r$ are variables of types that match the type of R. The formulas of first-order logic over L are built from the atomic formulas by negations, disjunctions, conjunctions, and existential and universal quantification of individual variables.

The syntax of Infinitary Logic with Counting $C_{\infty\omega}$ extends the syntax of first-order logic by all quantifiers of the form $\exists^{\geq i} x(\varphi)$, where i is a natural number, and formulas of the form $\bigvee_{i\in I}\phi_i$ and $\bigwedge_{i\in I}\phi_i$ where I is a possibly infinite index set, and $\{\phi_i : i \in I\}$ is an indexed set of formulas. The meaning of $\exists^{\geq i} x(\varphi)$ is that there exist at least i many witnesses a for the variable x within its sort such that assignment $x \mapsto a$ satisfies the formula φ. The fragment $C^k_{\infty\omega}$ of $C_{\infty\omega}$ is the set of formulas that use at most k variables of any type. We write $C^\omega_{\infty\omega}$ for the union of the $C^k_{\infty\omega}$ over all natural numbers k. For the definition of Fixed-Point Logic with Counting FPC see [14]. It is known that for every natural number k, every many-sorted vocabulary L, and every L-formula φ of FPC that uses k variables, there exists an L-formula ψ of $C^k_{\infty\omega}$ such that φ and ψ define the same relations over all finite L-structures (see, e.g. [14]).

Interpretations and reductions. Let L and K be two many-sorted vocabularies, and let Θ be a class of K-formulas. A Θ-interpretation of L in K is given by: two Θ-formulas $\delta_D(x)$ and $\epsilon_D(x,y)$ for each sort symbol D of L, and one Θ-formula $\psi_R(x_1,\ldots,x_r)$ for each relation symbol $R \in L$ of arity r. In all these formulas, the displayed x's and y's are tuples of distinct variables of the same length m, called the arity of the interpretation. We say that the interpretation takes a K-structure $\mathbb{A}$ as input and produces an L-structure $\mathbb{B}$ as output if for each sort symbol D in L there exists a surjective partial map $f_D : A^m \to D(\mathbb{B})$, where A is the domain of $\mathbb{A}$, such that $f_D^{-1}(D(\mathbb{B})) = \{a \in A^m : \mathbb{A} \models \delta_D(a)\}$, $f_D^{-1}(\{(b,b) : b \in D(\mathbb{B})\}) = \{(a,b) \in (A^m)^2 : \mathbb{A} \models \epsilon_D(a,b)\}$, and $f_R^{-1}(R(\mathbb{B})) = \{(a_1,\ldots,a_r) \in (A^m)^r : \mathbb{A} \models \psi_R(a_1,\ldots,a_r)\}$ where $f_R = f_{D_1} \times \ldots \times f_{D_r}$ and $D_1 \times \cdots \times D_r$ is the type of R. The composition of two interpretations, one of L in K, and another one of K in J, is an interpretation of L in J defined in the obvious way. Similarly, the composition of an interpretation of L in K with an L-formula is a K-formula defined in the obvious way. In all these compositions, the number of variables in the resulting formulas *multiply*. For example, the composition of a $C^k_{\infty\omega}$-interpretation with a $C^\ell_{\infty\omega}$-formula is a $C^{k\ell}_{\infty\omega}$-formula. A reduction from a problem to another is an interpretation that takes (a representation of) an input x for the first problem and produces (a representation of) an input y for the second problem, in such a way that (a representation of) a solution for y is also (a representation of) a solution for x. The reduction is called a Θ-reduction if it can be produced by a Θ-interpretation.

Numbers, vectors and matrices as structures. We represent natural numbers, integers and rational numbers as finite relational structures in the following way. A natural number $n \in \mathbb{N}$ is represented by a finite structure, with a domain $\{0,\ldots,N-1\}$ of *bit positions* where $N \geq \lfloor \log_2(n+1) \rfloor$, of a vocabulary $L_{\mathbb{N}}$ that contains a binary relation symbol $\leq$ for the natural *linear order* on the bit positions, and a unary relation symbol P for the *actual bits*, i.e., the bit positions i that carry a 1-bit in the unique binary representation of n of length N. Single bits $b \in \{0,1\}$ are represented as natural numbers with at least one bit position. Thus $L_{\mathbb{B}}$ is really the same as $L_{\mathbb{N}}$, but we still give it a separate name. Rationals $q \in \mathbb{Q}$ are represented by structures of the vocabulary $L_{\mathbb{Q}} = L_{\mathbb{B}} \dot\cup L_{\mathbb{N}} \dot\cup L_{\mathbb{N}}$, with a domain $\{0,\ldots,N-1\}$ that is large enough to encode both the numerator and the denominator of q in binary. If $q = (-1)^b n/d$, where $b \in \{0,1\}$ and $n,d \in \mathbb{N}$, then the P-relation from $L_{\mathbb{B}}$ is used to encode the sign b in the least significant bit-position, the P-relation from the first copy of $L_{\mathbb{N}}$ is used to encode the bits of the numerator n, and the P-relation from the second copy of $L_{\mathbb{N}}$ is used to encode the bits of the denominator d. Each $\leq$ is the natural linear order on the bit positions. Zero denominator means $\pm\infty$.

If $I_1,\ldots,I_d$ denote index sets that are not necessarily pairwise distinct, then the 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 I for as many different index sets as there are in the list $I_1,\ldots,I_d$, plus one sort $\bar B$ for the bit positions. The vocabulary $L_{\mathrm{vec},d}$ of these structures has one unary relation symbol I for each index sort $\bar I$, one binary relation symbol $\leq$ for the natural linear order on the bit positions $\bar B$, and three $d+1$-ary relation symbols P_s, P_n and P_d, each of type $\bar I_1\times\cdots\times\bar I_d\times\bar B$, for encoding the signs and the bits of the numerators and the denominators of the entries of the tensor. Vectors $u \in \mathbb{Q}^I$, matrices $A \in \mathbb{Q}^{I\times J}$ and square matrices $A \in \mathbb{Q}^{I\times I}$ are special cases of these, and so are indexed sets of vectors $\{u_i : i \in K\} \subseteq \mathbb{Q}^I$ and index sets of matrices $\{A_i : i \in K\} \subseteq \mathbb{Q}^{I\times J}$. We let $L_{\mathrm{vec}} := L_{\mathrm{vec},1}$.

3 Definable Ellipsoid Method

In this section we show that the ellipsoid method can be implemented in FPC for any family of explicitly bounded convex sets. We begin by defining the problems involved.

Geometric problems and the ellipsoid method. Let $\mathscr{C}$ be a class of convex sets, each of the form $K \subseteq \mathbb{R}^I$ for some non-empty index set I. The class $\mathscr{C}$ comes with an associated encoding scheme. We assume that the encoding of a set $K \subseteq \mathbb{R}^I$ carries within it enough information to determine the set I. If the encoding also carries information about a rational R satisfying $K \subseteq S(0^I, R)$, then we say that K is *circumscribed*, and we write $(K; I, R)$ to refer to it. We write $(K; n, R)$ whenever $I = [n]$.

The *exact feasibility problem* for $\mathscr{C}$ takes as input the encoding of a set $K \subseteq \mathbb{R}^I$ in $\mathscr{C}$ and asks for a bit $b \in \{0,1\}$ that is 1 if K is non-empty, and 0 if K is empty. The *weak feasibility problem* for $\mathscr{C}$ takes as input the encoding of a set $K \subseteq \mathbb{R}^I$ in $\mathscr{C}$ and a rational $\epsilon > 0$ and asks for a bit $b \in \{0,1\}$ and a vector $x \in \mathbb{Q}^I$ such that:

1. $b = 1$ and $x \in S(K,\epsilon)$, or
2. $b = 0$ and $\mathrm{vol}(K) \leq \epsilon$.

The reason why the exact feasibility problem is formulated as a decision problem and does not ask for a feasible point is that K could well be a single point with non-rational components. In the weak feasibility problem this is not an issue because if K is non-empty, then the ball $S(K,\epsilon)$ surely contains a rational point. The *not-so-weak separation problem* for $\mathscr{C}$ takes as input the encoding of a set $K \subseteq \mathbb{R}^I$ in $\mathscr{C}$, a vector $y \in \mathbb{Q}^I$, and a rational $\delta > 0$ and asks as output for a bit $b \in \{0,1\}$ and a vector $s \in \mathbb{Q}^I$ such that $||s||_\infty = 1$ and:

1. $b = 1$ and $y \in S(K,\delta)$, or
2. $b = 0$ and $\langle s,y\rangle + \delta \geq \sup\{\langle s,x\rangle : x \in K\}$.

The qualification *not-so-weak* serves the purpose of distinguishing the problem from the *weak(er)* version in which condition 2. is replaced by the looser requirement that $\langle s,y\rangle + \delta \geq \sup\{\langle s,x\rangle : x \in S(K,-\delta)\}$. It turns out that the main procedure of the ellipsoid method, as stated in the monograph [9] and in Theorem 1 below, requires the *not-so-weak* version. Recall that an ellipsoid in $\mathbb{R}^I$ is a set of form $E(A,a) = \{x \in \mathbb{R}^I : (x-a)^T A(x-a) \leq 1\}$, where $a \in \mathbb{R}^I$ is the center, and A is an $I \times I$ positive definite matrix.

Theorem 1 (Theorem 3.2.1 in [9]). *There is an oracle polynomial-time algorithm, the central-cut ellipsoid method (CC), that solves the*

following problem: Given a rational number $\epsilon > 0$ and a circumscribed closed convex set $(K; n, R)$ given by an oracle that solves the not-so-weak separation problem for K, outputs one of the following: either a vector $x \in S(K, \epsilon)$, or a positive definite matrix $A \in \mathbb{Q}^{n\times n}$ and a vector $a \in \mathbb{Q}^n$ such that $K \subseteq E(A, a)$ and $\mathrm{vol}(E(A, a)) \leq \epsilon$.

We plan to use algorithm CC from Theorem 1 almost as a black-box, except for its three aspects stated below. Although they are not stated in Theorem 3.2.1 in [9], inspection of the proof shows that they hold: (1) the input to the algorithm is the triple given by ϵ, n and R; (2) the rationals ϵ and R are represented in binary, the natural n is represented in unary; (3) the algorithm makes at least one oracle query, and the output is determined by the answer to the last oracle call in the following way: if this last call was (y, δ) and the answer was the pair (b, s), then $\delta \leq \epsilon$ and the output vector x of CC is y itself whenever $b = 1$, and there exists a positive definite matrix A and a vector a so that $K \subseteq E(A, a)$ and $\mathrm{vol}(E(A, a)) \leq \epsilon$ whenever $b = 0$. The last point implies, in particular, that CC solves the weak feasibility problem for the given K. Note also that CC solves the feasibility problem for K by making oracle calls to the separation problem for *the same* K.

Definability of ellipsoid. In our case, since we want to refer to definability in a logic, the encoding scheme for $\mathscr{C}$ will encode each set K through a finite relational structure, and we will require it to be invariant under isomorphisms. Such encodings we call *representations*. Formally, a *representation* of $\mathscr{C}$ is a surjective partial map r from the class of all finite L-structures onto $\mathscr{C}$, where L is a finite vocabulary that contains at least one unary relation symbol I, that satisfies the following conditions:

1. if $\mathbb{A}, \mathbb{B} \in \mathrm{Dom}(r)$ and $\mathbb{A} \cong \mathbb{B}$, then $r(\mathbb{A}) \cong r(\mathbb{B})$,
2. if $\mathbb{A} \in \mathrm{Dom}(r)$, then $r(\mathbb{A}) \subseteq \mathbb{R}^I$ where $I = I(\mathbb{A})$.

A *circumscribed representation* of $\mathscr{C}$ is a surjective partial map r from the class of all finite L-structures onto $\mathscr{C}$, where L is a finite vocabulary containing at least one unary relation symbol I as well as a copy of the vocabulary $L_{\mathbb{Q}}$, that satisfies 1. and 2. above, and:

3. if $\mathbb{A} \in \mathrm{Dom}(r)$, then $r(\mathbb{A}) \subseteq S(0^I, R)$ where $R = L_{\mathbb{Q}}(\mathbb{A})$.

A circumscribed representation of $\mathscr{C}$ exists only if every K in $\mathscr{C}$ is bounded. For a representation r of $\mathscr{C}$, any of the existing preimages $\mathbb{A} \in r^{-1}(K)$ of a set $K \in \mathscr{C}$ is called a *representation* of K. If $\mathscr{C}$ has a representation in some vocabulary L, we say that $\mathscr{C}$ is a *represented class of sets*, and if it has a circumscribed representation, we say that it is a *represented class of circumscribed sets*.

If $\mathscr{C}$ is a represented class of convex sets and Φ is a class of formulas, we say that the weak feasibility for $\mathscr{C}$ is Φ-definable if there exists a Φ-interpretation that, given an input represented as a structure over the vocabulary of the input, produces a valid output represented also as a structure over the vocabulary of the output.

The following is the main result of this section.

Theorem 2. *Let $\mathscr{C}$ be a represented class of circumscribed closed convex sets. If the not-so-weak separation problem for $\mathscr{C}$ is* FPC*-definable, then the weak feasibility problem for $\mathscr{C}$ is also* FPC*-definable.*

Although all the main ideas of the proof that we are going to present were already present in the works [2] and [7], we present a detailed proof for completeness.

At an intuitive level, the main difficulty for simulating the ellipsoid method within a logic is that one needs to make sure that the execution of the algorithm stays *canonical*; i.e., invariant under the isomorphisms of the input structure. The principal device to achieve this is the following clever idea from [2]: instead of running the ellipsoid method directly over the given set $K \subseteq \mathbb{R}^I$, the algorithm is run over certain *folded* versions $[K]^\sigma \subseteq \mathbb{R}^{\sigma(I)}$ of K, where $\sigma(I)$ is an *ordered subset* of I. If the execution of the ellipsoid does not detect the difference between K and $[K]^\sigma$, then an appropriately defined *unfolding* of the solution for $[K]^\sigma$ will give the right solution for K. If, on the contrary, the ellipsoid detects the difference in the form of a vector $u \in \mathbb{Q}^I$ whose folding $[u]^\sigma$ does not unfold appropriately, then the knowledge of u is exploited in order to *refine* the current folding into a strictly larger ordered $\sigma'(I) \subseteq I$, and the execution is restarted with the new $[K]^{\sigma'} \subseteq \mathbb{R}^{\sigma'(I)}$. After no more than $|I|$ many refinements the folding will be indistinguishable from K, and the execution will be correct.

The crux of the argument that makes the procedure definable in FPC is that the ellipsoid algorithm is always operating over an *ordered* set $\sigma(I)$. In particular, the algorithm stays canonical, and the polynomially many steps of its execution are expressible in fixed-point logic FP by the Immerman-Vardi Theorem. The counting of FPC is required only for the folding/unfolding/refining steps.

Before we move to the actual proof, we discuss the required material for the method of foldings.

Folding operations. Let I and J be non-empty index sets. Let $\sigma : I \to J$ be an onto map. The *folding* $[u]^\sigma$ of an I-vector u and the *unfolding* $[v]^{-\sigma}$ of a J-vector v are defined by $[u]^\sigma(j) := \sum_{i\in\sigma^{-1}(j)} u(i)/|\sigma^{-1}(j)|$ and $[v]^{-\sigma}(i) := v(\sigma(i))$ for every $j \in J$ and every $i \in I$, respectively. For sets $K \subseteq \mathbb{R}^I$ and $L \subseteq \mathbb{R}^J$, define $[K]^\sigma := \{[u]^\sigma : u \in K\}$ and $[L]^{-\sigma} := \{[v]^{-\sigma} : v \in L\}$. The map σ is said to *respect* a vector $u \in \mathbb{R}^I$ if $u_i = u_{i'}$ whenever $\sigma(i) = \sigma(i')$. The following lemma collects a few important properties of foldings. See Propositions 17 and 18 in [7] in which properties (4) and (5) from the lemma are also proved for all sets but stated only for convex sets.

Lemma 3. *Let $\sigma : I \to J$ be an onto map, let u and v be I-vectors, and let K be a set of I-vectors. Then the following hold: (1) $[au+bv]^\sigma = a[u]^\sigma + b[v]^\sigma$ for every $a, b \in \mathbb{R}$, (2) $\|[u]^\sigma\|_2 \leq \|u\|_2$, (3) $K \subseteq S(0^I, R)$ implies $[K]^\sigma \subseteq S(0^J, R)$, (4) $u \in S(K, \delta)$ implies $[u]^\sigma \in S([K]^\sigma, \delta)$, (5) if σ respects u, then $\langle u, v\rangle + \delta \geq \sup\{\langle u, x\rangle : x \in K\}$ implies $\langle [u]^\sigma, [v]^\sigma\rangle + \delta \geq \sup\{\langle [u]^\sigma, x\rangle : x \in [K]^\sigma\}$, and (6) if K is convex, then $[K]^\sigma$ is convex.*

There is one further important property of foldings that we will need. We extend the definition of the set $E(A, a)$ to arbitrary positive semidefinite matrices A. It should be noted that if A is positive semidefinite but not positive definite, then at least one of the semi-axes of $E(A, a)$ is infinite and hence the set is unbounded. In this case we call $E(A, a)$ an *unbounded ellipsoid*.

Lemma 4. *Let $K \subseteq \mathbb{R}^I$ be a set, let $\sigma : I \to J$ be an onto map, and let $R \in \mathbb{R}^{J\times I}$ and $L \in \mathbb{R}^{I\times J}$ be the matrices that define the linear maps $u \mapsto [u]^\sigma$ and $v \mapsto [v]^{-\sigma}$, respectively. If there is a positive definite matrix $A \in \mathbb{R}^{J\times J}$ and a vector $a \in \mathbb{R}^J$ such that $[K]^\sigma \subseteq E(A, a)$, then $K \subseteq E(R^T AR, La)$. Moreover, for every $\epsilon > 0$ and $r > 0$, if $\mathrm{vol}(E(A, a)) \leq \epsilon$, then $\mathrm{vol}(E(R^T AR, La) \cap S(0^I, r)) \leq 2^n r^{n-1} nk\epsilon^{1/k}$, where $n = |I|$ and $k = |J| \geq 1$.*

From now on, all maps $\sigma : I \to J$ will be onto and have $J = [k]$ for some positive integer k. Such maps define a preorder $\leq_\sigma$ on I with exactly k equivalence classes, defined by $i \leq_\sigma i'$ if and only

if $\sigma(i) \leq \sigma(i')$. A second map $\sigma' : I \to [k']$ is a *refinement* of σ if $\sigma'(i) \leq \sigma'(i')$ implies $\sigma(i) \leq \sigma(i')$. The refinement is *proper* if there exist $i, i' \in I$ such that $\sigma'(i) < \sigma'(i')$ and $\sigma(i) = \sigma(i')$. Recall that $\sigma : I \to [k]$ *respects* a vector $v \in \mathbb{R}^I$ if $v(i) = v(i')$ whenever $\sigma(i) = \sigma(i')$. Since any bijective map respects any vector, observe that if σ does not respect v, then there exists a least one proper refinement of σ that does respect v. Moreover, a coarsest proper refinement that respects v is FPC-definable by counting the number of distinct coordinates in each equivalence class, and splitting accordingly. We write σ^v for this FPC-definable proper refinement of σ. The next lemma collects a few computation tasks about foldings.

Lemma 5. *The following have* FPC*-interpretations: given I, an onto $\sigma : I \to [k]$, $u \in \mathbb{Q}^I$ and $v \in \mathbb{Q}^k$, output $[u]^\sigma$, $[v]^{-\sigma}$, $\sigma^u : I \to [k']$, and $b = 1$ if σ respects u, and $b = 0$ otherwise.*

Proof of Theorem 2. Let Ψ be an FPC-interpretation that witnesses that the not-so-weak separation problem for $\mathscr{C}$ is FPC-definable. We start by showing that there is an FPC-interpretation Ψ' that takes as input a representation of a set $K \subseteq \mathbb{R}^I$ in $\mathscr{C}$, an onto mapping $\sigma : I \to [k]$ where k is an integer that satisfies $1 \leq k \leq |I|$, a vector $y \in \mathbb{Q}^k$, and a rational $\delta > 0$ and outputs an integer $b \in \{-1, 0, 1\}$ and a vector $s \in \mathbb{Q}^I$ such that $\|s\|_\infty = 1$ and one of these holds:

1. $b = 1$, σ respects s, $[y]^{-\sigma} \in S(K, \delta)$ and $y \in S([K]^\sigma, \delta)$,
2. $b = 0$, σ respects s, $\langle [s]^\sigma, y\rangle + \delta \geq \sup\{\langle [s]^\sigma, x\rangle : x \in [K]^\sigma\}$,
3. $b = -1$, σ does not respect s.

Concretely, let Ψ' be the interpretation that, given a representation of $K \subseteq \mathbb{R}^I$ in $\mathscr{C}$, an onto $\sigma : I \to [k]$, a $y \in \mathbb{Q}^k$ and a rational $\delta > 0$, does the following:

01. given K, σ, y and δ as specified,
02. compute $y^- := [y]^{-\sigma}$ and $(b, s) := \Psi(K; y^-, \delta)$,
03. if σ respects s, output the same (b, s),
04. if σ does not respect s, output $(-1, s)$.

The claim that Ψ' is FPC-definable follows from Lemma 5. The claim that Ψ' satisfies the required conditions follows from the correctness of Ψ, together with the fact that $[[y]^{-\sigma}]^\sigma = y$, and properties (4) and (5) in Lemma 3. For later use, let us note that if the given $\sigma : I \to [k]$ is a bijection, then the third type of output $b = -1$ cannot occur.

Next we show how to use Ψ' in order to implement, in FPC, the algorithm CC from Theorem 1. Consider the following variant CC' of CC:

01. given a rational $\epsilon > 0$ and a representation of $K \subseteq \mathbb{R}^I$,
02. compute R with $K \subseteq S(0, R)$ from the representation of K,
03. $n := |I|$, $k := 1$, and $\sigma := 1_I : I \to [1]$ (the constant 1 map),
04. start CC on (γ, k, R) with $\gamma := \min\{(\epsilon/(2^n R^{n-1} nk))^k, \epsilon\}$,
05. given a query (y, δ), replace it by $(b, s) := \Psi'(K; \sigma, y, \delta)$,
06. if σ respects s, then
07. compute $[s]^\sigma$, take $(b, [s]^\sigma)$ as valid answer to (y, δ),
08. if the run of CC makes a new query (y, δ), goto 05,
09. if the run of CC makes no more queries, goto 13,
10. else
11. compute the canonical refinement $\sigma^s : I \to [k']$,
12. abort this run and goto 04 with $\sigma := \sigma^s$ and $k := k'$,
13. let (b, s) be the output of Ψ' for the last oracle call (y, δ),
14. output $(b, [y]^{-\sigma})$.

A key aspect of CC that makes this algorithm well-defined is that, for steps 04, 05, 08 and 09, the only knowledge that the algorithm needs about the targeted set $[K]^\sigma$ are its dimension k, its bounding radius R, and correct answers to earlier queries (see point 1. immediately following the statement of Theorem 1). In particular, the algorithm will be well-defined even if the class $\mathscr{C}$ is *not* closed under foldings, as long as the gathered knowledge about the alleged $[K]^\sigma$ remains consistent with the assumption that the convex set given by the oracle *is* $[K]^\sigma$. Note that properties (2), (4), and (5) in Lemma 3 guarantee so, as long as all s-vectors are respected by σ. As soon as this is detected to not be the case, σ is refined, and the run of CC is restarted with the new k for the new σ.

After no more than $|I|$ many refinements of σ, the simulation of the run of CC will be executed until the end. Indeed, this happens at latest once σ becomes the totally refined map because at that point σ is a bijection that surely respects every s. Whenever the run is executed until the end, the algorithm reaches step 13 with a pair (b, s) and a σ that respects s. We use this to show that CC' solves the weak feasibility problem for $\mathscr{C}$.

The claim that CC' solves the weak feasibility problem for $\mathscr{C}$ is proved as follows. Let (b, s) be the output of Ψ' for the last oracle call (y, δ) of the execution of CC. As noted above, $\sigma : I \to [k]$ respects s and hence $b \in \{0, 1\}$ by Property 3 in the description of Ψ'. If $b = 1$, then $[y]^{-\sigma} \in S(K, \delta)$ by Property 1 in the description of Ψ', and $S(K, \delta) \subseteq S(K, \epsilon)$ because $\delta \leq \gamma \leq \epsilon$. This shows that $(b, [y]^{-\sigma})$ is a correct output for the weak feasibility problem for ϵ and K in case $b = 1$. In case $b = 0$ we have $[K]^\sigma \subseteq E(A, a)$ for a positive definite matrix A and a vector a, with $\mathrm{vol}(E(A, a)) \leq \gamma \leq (\epsilon/(2^n R^{n-1} nk))^k$, by point 3. immediately following the statement of Theorem 1. Since $K \subseteq S(0, R)$, by Lemma 4 this means that the volume of K is at most ϵ and the answer $b = 0$ is a correct output.

For the implementation in FPC, we note that CC' is a relational WHILE algorithm that halts after at most $|I|$ iterations all whose steps can be computed through FPC-interpretations without quotients. Step 01 is the description of the input. Step 02 follows from the fact that K has a circumsbribed representation: just take the $L_\mathbb{Q}$-reduct of the representation of K, where $L_\mathbb{Q}$ is the copy of the vocabulary that is used for representing the rational radius R. Step 03 follows from Lemma 5. Step 04 follows from the Immerman-Vardi Theorem on the fact that the representation of $[k]$ is an ordered structure and the computation of CC in between oracle calls runs in polynomial time. Step 05 is just a control statement. Step 06 follows from Lemma 5. Step 07 follows from Lemma 5 and the fact that σ respects s. Step 08 follows, again, from the Immerman-Vardi Theorem on the fact that the representation of $[k]$ is an ordered structure and the computation of CC in between oracle calls runs in polynomial time. Step 09 follows from the same reason as Step 08. Step 10 is a control statement. Step 11 follows from Lemma 5. Step 12 and 13 are just control statements. Step 14 follows from Lemma 5.

4 Feasibility of SDPs

In this section we use Theorem 2 to show that the exact feasibility of semidefinite programs is definable in $\mathrm{C}^\omega_{\infty\omega}$.

Semidefinite sets. A *semidefinite set* $K_{A,b} \subseteq \mathbb{R}^I$ is the set of matrices $X \in \mathbb{R}^{J \times J}$ that satisfy

$$\langle A_i, X\rangle \leq b_i \quad \text{for } i \in M \quad \text{and} \quad X \succeq 0, \tag{1}$$

where $A \in \mathbb{R}^{M\times(J\times J)}$ is an indexed set of $J \times J$ matrices, $b \in \mathbb{R}^M$ is an indexed set of reals, X is a $J \times J$ symmetric matrix of formal variables $x_{ij} = x_{ji} = x_{\{i,j\}}$ for $i, j \in J$, and $I = \{\{i, j\} : i, j \in J\}$ is the set of variable indices. A *circumscribed semidefinite set* is a pair $(K_{A,b} \subseteq \mathbb{R}^I, R)$, where $K_{A,b} \subseteq \mathbb{R}^I$ is a semidefinite set as defined above and R is a rational satisfying $K_{A,b} \subseteq S(0^I, R)$. By $\mathscr{C}_{\mathrm{SDP}}$ and $\mathscr{C}^C_{\mathrm{SDP}}$ we denote the class of semidefinite sets and the class of circumscribed semidefinite sets, respectively.

When A and b have rational coefficients, the semidefinite set $K_{A,b} \subseteq \mathbb{R}^I$ is represented by a four-sorted structure, with one sort $\bar{I}$ for the set I of indices of variables, two sorts $\bar{J}$ and $\bar{M}$ for the index sets J and M, and one sort $\bar{B}$ for a domain $\{0, \ldots, N-1\}$ of bit positions that is large enough to encode all the numbers in binary. The vocabulary L_{SDP} includes the following relation symbols: three unary symbols $\bar{I}, \bar{J}$ and $\bar{M}$, for $\bar{I}, \bar{J}$ and $\bar{M}$, respectively, one ternary symbol P of type $\bar{I}\times\bar{J}\times\bar{J}$ that indicates the two indices of each variable, one binary symbol $\leq$ for the natural linear order on $\bar{B}$, three 4-ary symbols $P_{A,s}, P_{A,n}, P_{A,d}$ for the set of matrices $\{A_i : i \in M\}$, three binary symbols $P_{b,s}, P_{b,n}, P_{b,d}$ for the set of rationals $\{b_i : i \in M\}$. The representation of the circumscribed semidefinite set $(K_{A,b} \in \mathbb{R}^I, R)$ is a structure over the vocabulary $L_{\mathrm{SDP}} \dot\cup L_{\mathbb{Q}}$ whose L_{SDP}-reduct is the representation of $K_{A,b} \in \mathbb{R}^I$, and whose $L_{\mathbb{Q}}$-reduct is the representation of R.

In [7] Dawar and Wang show the following:

Theorem 6 ([7]). *The weak feasibility problem for circumscribed semidefinite sets is* FPC*-definable.*

In order to do so they prove Theorem 2 for the special case of semidefinite sets and propose an FPC-interpretation for the not-so-weak separation oracle. For completeness, in the full version we work out the details of a variant of their construction, indicating the precise place where our procedures differ, and why.

Exact feasibility. We use Theorem 6 to prove the main result of this section:

Theorem 7. *The exact feasibility problem for semidefinite sets is* $\mathrm{C}^\omega_{\infty\omega}$*-definable.*

We begin by relating the exact feasibility to the weak feasibility problem for circumscribed semidefinite sets. For any $R > 0$, the *R-restriction* of a semidefinite set $K_{A,b}$ is the set of all those points in $K_{A,b}$ whose L_∞-norm is bounded by R, i.e:

$$\langle A_i, X\rangle \leq b_i \ \text{ for } i \in M \quad |X_{\{i,j\}}| \leq R \ \text{ for } i, j \in J \quad \text{and} \quad X \succeq 0.$$

For any $\epsilon > 0$, the *ϵ-relaxation* of a semidefinite set $K_{A,b}$ is the semidefinite set given by:

$$\langle A_i, X\rangle \leq b_i + \epsilon \ \text{ for } i \in M \ \text{ and } \ X \succeq 0,$$

The question of emptiness for ϵ-relaxations of R-restrictions of semidefinite sets is linked to the problem under consideration. Recall the Cantor Intersection Theorem: If $K_1 \supseteq K_2 \supseteq \cdots$ is a decreasing nested sequence of non-empty compact subsets of $\mathbb{R}^n$, then the intersection $\bigcap_{i\geq 1} K_i$ is non-empty. We use it for the following lemma.

Lemma 8. *A semidefinite set $K_{A,b}$ is non-empty if and only if there exists a positive rational R such that for every positive rational ϵ it holds that the ϵ-relaxation of the R-restriction of $K_{A,b}$ is non-empty.*

It follows from Theorem 6 that the emptiness problem for ϵ-relaxations of R-restrictions of semidefinite sets is definable in FPC in the following sense.

Proposition 1. *There exists a formula ψ of* FPC *such that if $\mathbb{A}$ is a structure representing a semidefinite set $K_{A,b} \subseteq \mathbb{R}^I$ and two positive rational numbers R and ϵ, then: if $\mathbb{A} \models \psi$, the ϵ-relaxation of the R-restriction of $K_{A,b}$ is non-empty, and if $\mathbb{A} \not\models \psi$, the R-restriction of $K_{A,b}$ is empty.*

Proof. Let Φ be an FPC-interpretation that witnesses that the weak feasibility problem for the class of circumscribed semidefinite sets is FPC-definable. The formula ψ takes as input the representation of a semidefinite set $K_{A,b} \subseteq \mathbb{R}^I$, a rational $\epsilon > 0$ and a rational $R > 0$, and does the following:

01. given $K_{A,b} \subseteq \mathbb{R}^I$, ϵ and R as specified,
02. compute $k := |I|$,
03. compute $R' := \lceil\sqrt{k(R+\epsilon)^2}\rceil$,
04. compute K, the ϵ-relaxation of the R-restriction of $K_{A,b}$,
05. compute $m := \max\{\|A_i\|_2 : i \in M\} \cup \{1\}$,
06. compute $\delta = \epsilon^k/(k!(2km)^k)$,
07. compute $(b, x) := \Phi((K, R'), \delta)$,
08. if $b = 1$ output $\top$,
09. if $b = 0$ output $\bot$.

This procedure is clearly FPC-definable. In order to prove correctness we will need the following lemma.

Lemma 9. *Let $A \in \mathbb{R}^{M\times(J\times J)}$, $b \in \mathbb{R}^M$, $I = \{\{i, j\} : i, j \in J\}$, $k = |I|$, and $m = \max\{\|A_i\|_2 : i \in M\} \cup \{1\}$. For any $\epsilon > 0$, if the semidefinite set $K_{A,b} \in \mathbb{R}^I$ is non-empty, then its ϵ-relaxation has volume greater than $\delta = \epsilon^k/(k!(2km)^k)$.*

Proof. Take $\epsilon_1 = \epsilon/2km$ and let $Y \in K_{A,b}$. We show that $S(Y + \epsilon_1 \mathrm{I}, \epsilon_1)$ is included in the ϵ-relaxation of $K_{A,b}$. It follows that the volume of the ϵ-relaxation of $K_{A,b}$ is at least $\epsilon_1^k V_k$, where V_k is the volume of a 1-ball in the k-dimensional real vector space. Since $V_k > 1/k!$ this finishes the proof. □

We are ready to conclude the proof. Observe that the L_∞-norm of any point that belongs to the ϵ-relaxation of the R-restriction of a semidefinite set is bounded by $R + \epsilon$, therefore the pair (K, R') computed in Steps 03 and 04 is a representation of a circumscribed semidefinite set. Let (b, x) be the pair computed in Step 07.

If $b = 1$ then there exists a point in $S(K, \delta)$, which in particular means that K is non-empty, so the output in Step 08 is correct. If $b = 0$, then we know that the volume of K is at most δ. The inequalities that define K have the form $\langle A_i, X\rangle \leq b_i + \epsilon$ for $i \in M$, and $X_{\{i,j\}} \leq R + \epsilon$ or $-X_{\{i,j\}} \leq R + \epsilon$ for $i, j \in J$. The maximum 2-norm of the normals of these inequalities and 1 is $m = \max\{\|A_i\|_2 : i \in M\} \cup \{1\}$, so Lemma 9 applies. This means that K is empty, and the output in Step 09 is correct. □

To finish the proof of Theorem 7 we show a technical lemma that may sound a bit surprising at first: it sounds as if it was stating that $\mathrm{C}^k_{\infty\omega}$-definability is closed under second-order quantification over unbounded domains, which cannot be true. However, on closer look, the lemma states this *only* if the vocabularies of the quantified and the body parts of the formula are totally disjoint. In particular, this means that the domains of the sorts in the quantified and body parts of the formula stay unrelated *except* through the counting mechanism of $\mathrm{C}^k_{\infty\omega}$.

If $\mathscr{A}$ is a class of $L \cup K$-structures and $\mathscr{B}$ is a class of K-structures, we use the notation $\exists\mathscr{B} \cdot \mathscr{A}$ to denote the class of all finite L-structures $\mathbb{A}$ for which there exists a structure $\mathbb{B} \in \mathscr{B}$ such that $\mathbb{A} \dot\cup \mathbb{B} \in \mathscr{A}$. Similarly, we use $\forall\mathscr{B} \cdot \mathscr{A}$ to denote the class of all finite L-structures $\mathbb{A}$ such that for all structures $\mathbb{B} \in \mathscr{B}$ we have that $\mathbb{A} \dot\cup \mathbb{B} \in \mathscr{A}$.

Lemma 10. *Let L and K be many-sorted vocabularies with disjoint sorts, let $\mathscr{A}$ be a class of finite $L \cup K$-structures, and let $\mathscr{B}$ be a class of finite K-structures. If $\mathscr{A}$ is $\mathrm{C}^k_{\infty\omega}$-definable, then the classes of L-structures $\exists\mathscr{B} \cdot \mathscr{A}$ and $\forall\mathscr{B} \cdot \mathscr{A}$ are also $\mathrm{C}^k_{\infty\omega}$-definable.*

Proof. The proof is a simple *Booleanization* trick to replace the finite quantifiers $\exists^{\geq i}$ over the sorts in K by finite propositional formulas, followed by replacing $\exists\mathscr{B}$ and $\forall\mathscr{B}$ by infinite disjunctions and conjunctions, respectively, indexed by the structures in $\mathscr{B}$. □

Proof of Theorem 7. Let ψ be the $L_{\mathrm{SDP}} \dot\cup L_{\mathbb{Q}} \dot\cup L_{\mathbb{Q}}$-formula of FPC defined in Proposition 1. Let l be the number of variables in ψ. By the translation from l-variable FPC to $\mathrm{C}^l_{\infty\omega}$ (see Section 2), there exists an $L_{\mathrm{SDP}} \dot\cup L_{\mathbb{Q}} \dot\cup L_{\mathbb{Q}}$-formula τ of $\mathrm{C}^l_{\infty\omega}$ defining the same class $\mathscr{A}$ of finite structures. The vocabulary of $\mathscr{A}$ has disjoint sorts. Let $\mathscr{B}_R$ and $\mathscr{B}_\epsilon$ be classes of finite structures which are representations of positive rational numbers over the first and second copy of $L_{\mathbb{Q}}$, respectively. By Lemma 10 the class $\forall\mathscr{B}_\epsilon \cdot \mathscr{A}$, and hence $\exists\mathscr{B}_R \cdot \forall\mathscr{B}_\epsilon \cdot \mathscr{A}$, is also $\mathrm{C}^l_{\infty\omega}$-definable. Let ϕ be the L_{SDP}-formula of $\mathrm{C}^l_{\infty\omega}$ defining this last class. Lemma 8 implies that ϕ defines the exact feasibility problem for semidefinite sets. □

5 SOS Proofs and Lasserre Hierarchy

In this section we develop the descriptive complexity of the problem of deciding the existence of Sums-of-Squares proofs. Along the way we discuss the relationship between the Lasserre hierarchy of SDP relaxations and SOS, and how 0/1-valued variables ensure strong duality. We use the strong duality to argue the equivalence between the existence of SOS refutations and the existence of a notion of SOS *approximate* refutations that we introduce.

Descriptive Complexity of SOS Proofs. Let $x_1, \ldots, x_n$ be a set of variables. In the following when we talk about polynomials or monomials we mean polynomials and monomials over the set of variables $x_1, \ldots, x_n$ and real or rational coefficients. For a set $Q = \{q_1, \ldots, q_k\}$ of polynomials and a polynomial q, a *Sums-of-Squares proof* of $q \geq 0$ from Q is an identity:

$$\sum_{j\in[m]} p_j s_j = q, \tag{2}$$

where, for every $j \in [m]$, the polynomial s_j is a sum of squares of polynomials, and the polynomial p_j is either in Q or in the set B_n defined as follows:

$$1, \quad x_i, \quad 1 - x_i, \quad x_i^2 - x_i, \quad x_i - x_i^2, \quad \text{for every } i \in [n]. \tag{3}$$

We refer to the inequalities $p \geq 0$ for $p \in B_n$ as *Boolean axioms*. The degree of the proof is defined as $\max\{\deg(p_j s_j) : j \in [m]\}$, where, for a polynomial p, the notation $\deg(p)$ denotes the degree of p.

One should think about the set of polynomials Q as representing a system of polynomial inequalities $\{q_i \geq 0 : i \in [k]\}$. The identity (2) implies that any 0/1-solution to this system satisfies also the inequality $q \geq 0$. Therefore, if $q = -1$, a proof certifies that the system $\{q_i \geq 0 : i \in [k]\}$ has no 0/1-solutions. This is why we call it a *refutation* of Q. Sometimes we allow the system to include equations $q_i = 0$, which we think of as the set of two inequalities $q_i \geq 0$ and $-q_i \geq 0$, i.e., $\{q_i, -q_i\} \subseteq Q$.

We consider the problem of deciding the existence of SOS proofs and refutations of a fixed degree $2d$ for a set of polynomials given as input. The first easy observation is that the proof-existence problem can be reduced to the exact feasibility problem for semidefinite sets, and the reduction can be done in FPC. Then we ask whether the exactness condition in the feasibility problem for semidefinite sets can be relaxed, and we achieve this for refutations. In other words:

1. Proof-existence reduces in FPC to exact SDP feasibility.
2. Refutation-existence reduces in FPC to weak SDP feasibility.

We note that, in both cases, the semidefinite sets in the outcome of this reduction are not circumbscribed. As stated, point 1. above is almost a reformulation of the problem. In order to prove point 2. we develop a notion of approximate refutation, and combine it with a strong duality theorem that characterizes the existence of SOS refutations in terms of so-called *pseudoexpectations*. We note that the strong duality theorem that we need relies on the assumption that the Boolean axioms are allowed for free in the definition of SOS. Finally, we combine these FPC reductions with the results of the previous section in order to get the following:

Corollary 1. *For every fixed positive integer d, the problems of deciding the existence of SOS proofs of degree $2d$, and SOS refutations of degree $2d$, are $\mathrm{C}^\omega_{\infty\omega}$-definable. Moreover, there exists a constant c, independent of d, such that the defining formulas are in $\mathrm{C}^{cd}_{\infty\omega}$.*

As usual with descriptive complexity results like these, we need to fix an encoding of the input as finite relational structures. In this case the inputs are indexed sets of polynomials. The exact choice of encoding is not essential, but we propose one for concreteness.

Let I be an index set and let $\{x_i : i \in I\}$ be a set of formal variables. A *monomial* is a product of variables. We use the notation x^α, where $\alpha \in \mathbb{N}^I$, to denote the monomial that has *degree* α_i on variable x_i. We write $|\alpha|$ for the degree $\sum_{i\in I} \alpha_i$ of the monomial x^α. A *polynomial* $\sum_\alpha c_\alpha x^\alpha$ is a finite linear combination of monomials, i.e. all but finitely many of the coefficients c_α are zero. A polynomial p with rational coefficients is represented by a three-sorted structure, with a sort $\bar{I}$ for the index set I, a second sort $\bar{M}$ for the finite set of monomials that have non-zero coefficient in p, and a third sort $\bar{B}$ for a domain $\{0, \ldots, N-1\}$ of bit positions, where N is large enough to encode all the coefficients of p and all the degrees of its monomials in binary. The vocabulary L_{pol} of this structure has one unary relation symbol I for $\bar{I}$, one binary relation symbol $\leq$ for the natural linear order on $\bar{B}$, three binary relations symbols P_s, P_n, and P_d of type $\bar{M} \times \bar{B}$ that encode, for each monomial, the sign, the bits of the numerator, and the bits of the denominator of its coefficient, respectively, and a ternary relation symbol D of type $\bar{M} \times \bar{I} \times \bar{B}$ that encodes, for each monomial and each variable, the bits of the degree of this variable in the monomial.

Lasserre hierarchy. For a set of polynomials $\{q_0, q_1, \ldots, q_k\}$, by $\mathrm{POP}(q_0; \{q_1, \ldots, q_k\})$ we denote the polynomial optimisation problem :

$$\text{(POP)} \quad : \quad \inf_x q_0 \ \text{ s.t. } q_i \geq 0 \ \text{ for } i \in [k], \tag{4}$$

Take $d > 0$. By M_d we denote the matrix indexed by monomials of degree at most d over the variables $x_1, \ldots, x_n$ where $(M_d)_{\alpha,\beta} = x^{\alpha+\beta}$. For every monomial x^α, we introduce a variable y_α and by $M_d(y)$ we denote the corresponding matrix of variables, i.e., $(M_d(y))_{\alpha,\beta} = y_{\alpha+\beta}$. More generally, for any polynomial

$q = \sum_\gamma c_\gamma x^\gamma$, the matrix $M_{q,d}$, indexed by monomials of degree at most d, is defined by $M_{q,d} = qM_d$, i.e., $(M_{q,d})_{\alpha,\beta} = qx^{\alpha+\beta}$. The corresponding matrix $M_{q,d}(y)$ is defined by $(M_{q,d}(y))_{\alpha,\beta} = \sum_\gamma c_\gamma y_{\alpha+\beta+\gamma}$. Observe that the entries of the matrix $M_{q,d}$ are polynomials of degree at most $2d + \deg q$, while the entries of the matrix $M_{q,d}(y)$ are the corresponding linear combinations of variables. Note also that $M_{1,d} = M_d$ and $M_{1,d}(y) = M_d(y)$. For every variable y_α, consider the coefficients of y_α in the matrix $M_{q,d}(y)$. Those coefficients form a matrix which we denote by $A_{q,d,\alpha}$. Formally, for $|\alpha| \leq 2d + \deg q$, the matrices $A_{q,d,\alpha}$ are defined as the real matrices satisfying $M_{q,d}(y) = \sum_\alpha y_\alpha A_{q,d,\alpha}$ or equivalently $M_{q,d} = \sum_\alpha x^\alpha A_{q,d,\alpha}$. Finally, for any polynomial q, by d_q we denote the biggest integer satisfying $2d_q + \deg q \leq 2d$.

Let Q be a set of polynomials. For any positive integer d, the *Lasserre SDP relaxation* of the polynomial optimisation problem $\mathrm{POP}(\sum_\alpha a_\alpha x^\alpha; Q)$ of *order* d is the pair of semidefinite programs (P_d, D_d), where P_d is the *primal* semidefinite program:

$$\begin{aligned} \inf_y \ & \textstyle\sum_\alpha a_\alpha y_\alpha \\ & y_\emptyset = 1 \\ & M_{q,d_q}(y) \succeq 0, \text{ for every } q \in Q \end{aligned} \tag{5}$$

and D_d is the *dual* semidefinite program:

$$\begin{aligned} \sup_{z,Z} \ & z \\ & \textstyle\sum_{q\in Q} \langle A_{q,d_q,\emptyset}, Z_q\rangle = a_\emptyset - z \\ & \textstyle\sum_{q\in Q} \langle A_{q,d_q,\alpha}, Z_q\rangle = a_\alpha, \text{ for } 1 \leq |\alpha| \leq 2d \\ & Z_q \succeq 0, \text{ for every } q \in Q \end{aligned} \tag{6}$$

Weak SDP duality implies that the optimal value of P_d is always greater or equal to the optimal value of D_d. In [10] the authors establish a condition which guarantees strong duality for primal and dual SDP problems in the Lasserre hierarchy.

Theorem 11 ([10]). *If $\mathrm{POP}(q_0; Q)$ is a polynomial optimisation problem where one of the inequalities describing the feasibility region is $R^2 - \sum_{i\in[n]} x_i^2 \geq 0$, then for every positive integer d, the optimal values of P_d and D_d are equal.*

The polynomial optimisation problem $\mathrm{POP}(q_0; Q)$ is called *encircled* if a polynomial $R^2 - \sum_{i\in[n]} x_i^2$ can be obtained as a positive linear combination of polynomials from Q of degree at most 2. The following lemma implies strong duality for primal and dual SDP problems in the Lasserre hierarchy for encircled polynomial optimisation problems.

Lemma 12. *Let Q be a set of polynomials and let $p = \sum_{q\in Q} c_q q$ be a positive linear combination of polynomials from Q, such that $\deg p = \max\{\deg q : c_q > 0\}$. For some polynomial q_0, let (P_d, D_d) and (P'_d, D'_d) be the order d Lasserre SDP relaxations of $\mathrm{POP}(q_0; Q)$ and $\mathrm{POP}(q_0; Q \cup \{p\})$, respectively. The optimal values of P_d and P'_d, as well as the optimal values of D_d and D'_d are equal.*

SOS proofs as semidefinite sets. Fix a set of polynomials Q and a further polynomial $p = \sum_\alpha a_\alpha x^\alpha$. Let $\bar{Q} = Q \cup B_n$. A polynomial s of degree at most $2t$ is a sum of squares if and only if there exists a positive semidefinite matrix Z indexed by monomials of degree at most t such that $s = \langle M_t, Z\rangle$. Therefore, there exists a degree-$2d$ SOS proof of the polynomial inequality $p \geq 0$ from Q if and only if, for every $q \in \bar{Q}$, there exists a positive semidefinite matrix Z_q indexed by monomials of degree at most d_q such that

$$\begin{aligned} \textstyle\sum_{q\in\bar{Q}} q\langle M_{d_q}, Z_q\rangle &= \textstyle\sum_{q\in\bar{Q}} \langle M_{q,d_q}, Z_q\rangle = \\ &= \textstyle\sum_{q\in\bar{Q}} \langle \sum_\alpha x^\alpha A_{q,d_q,\alpha}, Z_q\rangle = \\ &= \textstyle\sum_\alpha x^\alpha \sum_{q\in\bar{Q}} \langle A_{q,d_q,\alpha}, Z_q\rangle = \sum_\alpha a_\alpha x^\alpha. \end{aligned} \tag{7}$$

Observe that the existence of a set of positive semidefinite matrices $\{Z_q : q \in \bar{Q}\}$ satisfying the identity (7) is exactly the same as non-emptiness of the semidefinite set $K_d(Q,p) \subseteq \mathbb{R}^{I_d}$ given by:

$$\textstyle\sum_{q\in\bar{Q}} \langle A_{q,d_q,\alpha}, Z_q\rangle = a_\alpha \ \text{ for } |\alpha| \leq 2d \ \text{ and } \ X \succeq 0, \tag{8}$$

where $J_d = \{(q, x^\alpha) : q \in \bar{Q}, |\alpha| \leq d_q\}$ is a set of indices, X is a $J_d \times J_d$ symmetric matrix of formal variables, $I_d = \{\{(q,x^\alpha),(q',x^{\alpha'})\} : (q,x^\alpha),(q',x^{\alpha'}) \in J_d\}$ is a set of variable indices, and for every $q \in \bar{Q}$, the matrix Z_q is the principal submatrix of X corresponding to the rows and columns indexed by $\{(q,x^\alpha) : |\alpha| \leq d_q\}$.

Indeed, from every feasible point $X \in K_d(Q,p)$ we get a set of positive semidefinite matrices $\{Z_q : q \in \bar{Q}\}$ satisfying the identity (7) by setting Z_q be the principal submatrix of X corresponding to the rows and columns indexed by $\{(q,x^\alpha) : |\alpha| \leq d_q\}$. On the other hand, any set of positive semidefinite matrices $\{Z_q : q \in \bar{Q}\}$ satisfying the identity (7) can be extended to a point in $K_d(Q,p)$ by setting all remaining variables to 0.

The representation of the semidefinite set $K_d(Q,p)$ can be easily obtained from the representation of the set of polynomials Q and the polynomial p by means of FPC-interpretations:

Fact 1. *For every positive integer d, there is an* FPC*-interpretation that takes a set of polynomials Q and a polynomial p as input and outputs a representation of the semidefinite set $K_d(Q,p)$. Moreover, there exists a constant c, independent of d, such that the formulas in the* FPC *interpretation have at most cd variables.*

Therefore, as a consequence of Theorem 7 we obtain Corollary 1.

Proof of Corollary 1. Let us fix a positive integer d and let Φ be the FPC-interpretation from Fact 1. We compose Φ with the $\mathrm{C}^\omega_{\infty\omega}$-sentence from Theorem 7 that decides the exact feasibility of semidefinite sets. The resulting sentence ψ decides the existence of an SOS proof of degree $2d$. It is a sentence of $\mathrm{C}^k_{\infty\omega}$, where $k = cd$, for an integer c that is independent of d. A $\mathrm{C}^\omega_{\infty\omega}$-sentence deciding the existence of an SOS refutation of degree $2d$ is obtained analogously by starting with an FPC-interpretation which takes as input a set of polynomials Q and outputs the semidefinite set $K_d(Q,-1)$. □

Approximate SOS refutations. For any $\epsilon > 0$, an *ϵ-approximate* degree-$2d$ SOS refutation of a set of polynomials Q is an identity:

$$\textstyle\sum_{q\in\bar{Q}} qs_q = \sum_\alpha a_\alpha x^\alpha, \tag{9}$$

where for every $q \in \bar{Q}$, the polynomial s_q is a sum of squares, for each α of degree at least 1, we have $|a_\alpha| \leq \epsilon$, and $|1 + a_\emptyset| \leq \epsilon$. The same way as the degree-$2d$ SOS refutations correspond to the points in $K_d(Q,-1)$, the ϵ-approximate degree-$2d$ SOS refutations correspond to the points in the ϵ-relaxation of $K_d(Q,-1)$.

We relate the existence of SOS refutations to primal and dual problems in the Lasserre hierarchy for the problem $\mathrm{POP}(0;\bar{Q})$. The goal is to use strong duality for showing that, for small enough ϵ depending on the degree and the number of variables, the existence of SOS refutations is equivalent to the existence of ϵ-approximate ones.

It follows that the problem of deciding the existence of SOS refutations of a fixed degree reduces, by means of FPC-interpretations, to the weak feasibility problem for semidefinite sets.

For any set of polynomials Q, the polynomial optimisation problem $\mathrm{POP}(0;\bar{Q})$ will be denoted by $\mathrm{SOL}(Q)$:

$$(\mathrm{SOL}(Q)) \quad : \quad \inf_x 0 \quad \text{s.t. } q \geq 0 \ \text{ for } q \in \bar{Q}. \tag{10}$$

Clearly, the problem $\mathrm{SOL}(Q)$ is feasible if and only if the system of polynomial inequalities $\{q \geq 0 : q \in Q\}$ has a 0/1-solution.

For a positive integer d, by $(P_d(Q), D_d(Q))$ we denote Lasserre SDP relaxation of the polynomial optimisation problem $\mathrm{SOL}(Q)$ of order d. Observe that degree-$2d$ SOS refutations of Q correspond precisely to the feasible solutions to $D_d(Q)$ with value 1 (see identity (7)). The following lemma summarizes the relationship between degree-$2d$ SOS refutations of Q and solutions to the program $D_d(Q)$. The second equivalence follows from the fact that by multiplying a solution to $D_d(Q)$ with value v by any $p \geq 0$ we obtain another solution with value pv.

Lemma 13. *There exists an* SOS *refutation of Q of degree $2d$ if and only if $D_d(Q)$ has a solution with value 1 if and only if the optimal value of $D_d(Q)$ is $+\infty$.*

For a system of polynomials Q, a *pseudoexpectation for Q of degree $2d$* is a linear mapping F from the set of polynomials of degree at most $2d$ over the set of variables $x_1, \ldots, x_n$ to the reals such that $F(1) = 1$, and for every $q \in \bar{Q}$ and every sum of squares polynomial s of degree at most $2d_q$, we have $F(qs) \geq 0$.

A linear mapping from the set of polynomials of degree at most $2d$ to the reals is uniquely defined by its restriction to monomials. Therefore, there is a natural one-to-one correspondence between linear functions from the set of polynomials of degree at most $2d$ to the reals and assignments to the set of variables $\{y_\alpha : |\alpha| \leq 2d\}$ of the program $P_d(Q)$, given by $G(y_\alpha) = F(x^\alpha)$. It is easy to see that an assignment G to the variables of $P_d(Q)$ is a feasible solution if and only if F is a pseudoexpectation of degree $2d$.

Lemma 14. *There exists a degree-$2d$ pseudoexpectation for Q if and only if the program $P_d(Q)$ is feasible.*

By summing the inequalities $1 - x_1 \geq 0, \ldots, 1 - x_n \geq 0$, together with $x_1 - x_1^2 \geq 0, \ldots, x_n - x_n^2 \geq 0$ one obtains $n - \sum_{i\in[n]} x^2 \geq 0$, which witnesses the fact that the problem $\mathrm{SOL}(Q)$ is encircled. Hence, by Lemma 12 for the problem $\mathrm{SOL}(Q)$ there is no duality gap between primal and dual SDP problems in the Lasserre hierarchy and the optimal value of $D_d(Q)$ is $+\infty$ if and only if $P_d(Q)$ is infeasible. As a consequence of Lemmas 13 and 14 we get:

Corollary 2. *There exists an* SOS *refutation of Q of degree $2d$ if and only if there is no pseudoexpectation for Q of degree $2d$.*

Suppose that Q has no degree-$2d$ SOS refutation. By strong duality this implies the existence of a degree-$2d$ pseudoexpectation. This in turn, as we will show now, precludes even the existence of ϵ-approximate refutations, for small enough ϵ. The key is the following lemma.

Lemma 15. *If F is a degree-$2d$ pseudoexpectation for Q, then $0 \leq F(m) \leq 1$ for every monomial m of degree at most d, and $-1 \leq F(m) \leq 1$ for every monomial m of degree at most $2d$.*

Proof. First we show that if m is a monomial of degree at most $2d$ and $\bar{m}$ denotes its multilinearization, then $F(\bar{m}) = F(m)$. We do this by showing that $F(x^2 m) = F(xm)$ for every variable x and every monomial m of degree at most $2d-2$. Fix such a monomial m and let r and s be monomials of degree at most $d-1$ such that $m = rs$. Note that $m = p^2 - q^2$ where $p = (r+s)/2$ and $q = (r-s)/2$, and both p^2 and q^2 have degree at most $2d-2$. It holds that $F((x^2-x)m) = F((x^2-x)p^2) + F((x-x^2)q^2) \geq 0$ and $F((x^2-x)m) = -F((x^2-x)q^2) - F((x-x^2)p^2) \leq 0$. This shows $F((x^2-x)m) = 0$ and hence $F(x^2 m) = F(xm)$.

We show that $0 \leq F(m) \leq 1$ for every m of degree at most d. By the previous paragraph we have $F(m) = F(m^2)$, and $F(m^2) \geq 0$, since m^2 is a square of degree at most $2d$. The other inequality is shown by induction on the degree. For the empty monomial 1 we have $F(1) = 1$. Now let m be a monomial of degree at most $d-1$ such that $F(m) \leq 1$ and let x be a variable. It holds that $F(m) - F(xm) = F((1-x)m) = F((1-x)m^2) \geq 0$, so $F(xm) \leq F(m) \leq 1$.

Finally, let m be a monomial of degree at most $2d$ and let r and s be monomials of degree at most d such that $m = rs$. We have $F(r^2) + 2F(rs) + F(s^2) = F((r+s)^2) \geq 0$. Therefore, $2F(rs) \geq -F(r^2) - F(s^2) \geq -2$, so $F(m) \geq -1$. Similarly $F(r^2) - 2F(rs) + F(s^2) = F((r-s)^2) \geq 0$. Hence, $2F(rs) \leq F(r^2) + F(s^2) \leq 2$, so $F(m) \leq 1$. □

The number of monomials of degree at most $2d$ over the set of n variables is $\binom{n+2d}{2d}$. Let $\epsilon_{n,d} = 1/(3\binom{n+2d}{2d})$. We are now ready to show that the existence of a degree-$2d$ SOS refutation of a system of polynomial inequalities with n variables is equivalent to the existence of an $\epsilon_{n,d}$-approximate such refutation.

Proposition 2. *Let Q be a set of polynomials with at most n variables. The set Q has an* SOS *refutation of degree $2d$ if and only if it has an $\epsilon_{n,d}$-approximate* SOS *refutation of degree $2d$.*

Proof. The left-to-right implication is clear. Now assume that Q has no SOS refutation of degree $2d$. Therefore, by Corollary 2 there exists a pseudoexpectation of degree $2d$. Let us denote it by F. Suppose that Q has an $\epsilon_{n,d}$-approximate SOS refutation of degree $2d$, i.e., there exists a set of sum of squares polynomials $\{s_q : q \in \bar{Q}\}$ such that $\sum_{q\in\bar{Q}} qs_q = \sum_\alpha a_\alpha x^\alpha$, where for each α of degree at least 1, we have $|a_\alpha| \leq \epsilon_{n,d}$, and $|1 + a_\emptyset| \leq \epsilon_{n,d}$.

Now, observe that $F\big(\sum_{q\in\bar{Q}} qs_q\big) = \sum_{q\in\bar{Q}} F(qs_q) \geq 0$, while $F(\sum_\alpha a_\alpha x^\alpha) = a_\emptyset + \sum_{\alpha\neq\emptyset} a_\alpha F(x^\alpha) \leq -1 + \epsilon_{n,d} + \binom{n+2d}{2d}\epsilon_{n,d}$, which is at most $-\frac{1}{3}$. Obtained contradiction finishes the proof. □

An ϵ-relaxation of a convex set K is either empty, which implies the emptiness of the set K itself, or it has volume greater than δ, where δ can be easily computed by means of FPC-interpretations from the representation of K and ϵ (see Lemma 9). Therefore:

Corollary 3. *For every positive integer d, there is an* FPC*-definable reduction from the problem of deciding the existence of* SOS *refutations of degree $2d$, to the weak feasibility problem for semidefinite sets.*

Proof. The FPC-interpretation takes a set of polynomials Q with n variables as input and outputs the $\epsilon_{n,d}$-relaxation of $K_d(Q,-1)$ and a rational $\delta > 0$, such that either the $\epsilon_{n,d}$-relaxation of $K_d(Q,-1)$ is empty, or it has volume greater than δ. □

6 Isomorphism

We formulate the isomorphism problem for two graphs G and H as a system $\mathrm{ISO}(G,H)$ of quadratic polynomial equations over $\mathbb{R}$, with 0/1-valued variables. Let U and V denote the sets of vertices of G and H. For $u_1, u_2 \in U$, we write $\mathrm{tp}_G(u_1,u_2)$ for the atomic

type of (u_1, u_2) in G. Similarly, for $v_1, v_2 \in V$, we write $\mathrm{tp}_H(v_1, v_2)$ for the atomic type of (v_1, v_2) in H. The system of equations has one variable x_{uv} for each pair of vertices $u \in U$, $v \in V$; the intended meaning of x_{uv} is that vertex u is mapped to v by an alleged isomorphism. The set of equations of $\mathrm{ISO}(G, H)$ is the following:

$$\begin{array}{ll} \sum_{v \in V} x_{uv} - 1 = 0 & \text{for each } u \in U, \\ \sum_{u \in U} x_{uv} - 1 = 0 & \text{for each } v \in V, \\ x_{u_1 v_1} x_{u_2 v_2} = 0 & \text{for each } u_1, u_2 \in U, v_1, v_2 \in V \\ & \text{with } \mathrm{tp}_G(u_1, u_2) \neq \mathrm{tp}_H(v_1, v_2). \end{array}$$

It is straightfoward to check that the set of equations $\mathrm{ISO}(G, H)$ can be produced from G and H by an FPC-interpretation:

Fact 2. *There is an* FPC*-interpretation that takes two graphs G and H as input and outputs the set of equations* $\mathrm{ISO}(G, H)$.

If graphs G and H are isomorphic, we write $G \cong H$. An SOS proof of $G \ncong H$ is an SOS refutation of $\mathrm{ISO}(G, H)$. A Sherali-Adams (SA) proof of $G \ncong H$ is an SA proof of $-1 \geq 0$ from $\mathrm{ISO}(G, H)$, where an SA proof is an identity of the type (2) in which the polynomials s_j are not sums-of-squares but extended monomials, i.e., polynomials of the form $c \cdot \prod_{j \in J} x_j \prod_{k \in K} (1 - x_k)$ where c is positive real. A (monomial) Polynomial Calculus (PC) proof of $G \ncong H$ is a (monomial) PC proof of $-1 = 0$ from the system of polynomial equations $\mathrm{ISO}(G, H)$; for definitions see [5].

We rely on the following facts from [3] and [5]:

Theorem 16. *Let G and H be graphs and let k be a positive integer. The following are equivalent:*

1. *$G \equiv^k H$, i.e., G and H satisfy the same $\mathrm{C}^k_{\infty\omega}$-sentences,*
2. *there is no degree-k* SA *proof of $G \ncong H$,*
3. *there is no degree-k monomial* PC *proof of $G \ncong H$.*

Indeed, [3] uses a slightly different encoding of the isomorphism problem as a system of polynomial equations, but using methods from [3] and [5] Theorem 16 follows for the encoding we use here. For the collapse result we are about to prove, we use *2* implies *1* and Corollary 1.

Theorem 17. *There exist an integer constant c such that, for all pairs of graphs G and H and all positive integers d, if there is a degree-$2d$* SOS *proof of $G \ncong H$, then there is a degree-cd* SA *proof of $G \ncong H$.*

Proof. Fix a positive integer d. Let Φ be the FPC-interpretation from Fact 2 and compose it with the $\mathrm{C}^\omega_{\infty\omega}$-sentence from Corollary 1 that decides the existence of an SOS proof of degree $2d$. The resulting sentence ϕ is a sentence of $\mathrm{C}^k_{\infty\omega}$, where $k = cd$ for an integer c that is independent of d. The sentence ϕ was designed in such a way that for every pair of graphs G and H it holds that $(G, H) \models \phi$ if and only if there is a degree-$2d$ SOS proof of $G \ncong H$. In particular, since there certainly is no degree-$2d$ SOS proof that G is not isomorphic to itself, we have $(G, G) \models \neg\phi$. Now assume that there is no degree-k SA proof of $G \ncong H$. We get $G \equiv^k H$ by Theorem 16, from which it follows that $(G, H) \equiv^k (G, G)$. Since ϕ is a $\mathrm{C}^k_{\infty\omega}$-sentence and $(G, G) \models \neg\phi$ we get $(G, H) \models \neg\phi$. Therefore, by design of ϕ, there is no degree-$2d$ SOS proof of $G \ncong H$. □

Next we use the following recent result of Berkholz [4] unexpectedly showing that SOS simulates PC; we remark that this result holds only for systems of equations with 0/1-values.

Theorem 18 ([4])**.** *Let Q be a system of polynomial equations over $\mathbb{R}$ with 0/1-valued variables. If Q has a* PC *refutation of degree d, then Q has an* SOS *refutation of degree $2d$.*

For two non-isomorphic graphs G and H, let $\mathrm{sos}(G, H)$, $\mathrm{sa}(G, H)$, $\mathrm{monpc}(G, H)$ and $\mathrm{pc}(G, H)$ denote the smallest degrees for which SOS, SA, monomial PC and PC can prove that G and H are not isomorphic, respectively. For isomorphic graphs let us take all three quantities to be ∞. Combining Theorems 18, 16, 17, we get the following full cycle of implications:

$$\tfrac{1}{2} \cdot \mathrm{sos}(G, H) \leq \mathrm{pc}(G, H) \leq \mathrm{monpc}(G, H) \leq \mathrm{sa}(G, H) \leq \tfrac{c}{2} \cdot \mathrm{sos}(G, H).$$

where c is the constant in Theorem 17. By returning to the *primals*, the same results can be stated in terms of the number of levels that are required for the Lasserre [11] and the Sherali-Adams [15] hierarchies to become infeasible. The result says that, for any pair of graphs G and H, the first levels at which the relaxations for $\mathrm{ISO}(G, H)$ become infeasible are separated by no more than a constant $c/2$-factor.

Acknowledgments

We are grateful to Christoph Berkholz, Anuj Dawar, and Wied Pakusa, for useful discussions at an early stage of this work. We are also grateful to Aaron Potechin for pointing out that the ability of the Lasserre hierarchy to capture spectral arguments was relevant for our result. 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). Second author supported by the French Agence Nationale de la Recherche, AGGREG project reference ANR-14-CE25-0017-01. Part of this work was done while the second author was visiting UPC funded by AUTAR.

References

[1] E. Allender, P. Bürgisser, J. Kjeldgaard-Pedersen, and P. B. Miltersen. 2009. On the Complexity of Numerical Analysis. *SIAM J. Comput.* 38, 5 (2009), 1987–2006.
[2] M. Anderson, A. Dawar, and B. Holm. 2015. Solving Linear Programs Without Breaking Abstractions. *J. ACM* 62, 6 (2015), 48:1–48:26.
[3] A. Atserias and E. Maneva. 2013. Sherali–Adams Relaxations and Indistinguishability in Counting Logics. *SIAM J. Comput.* 42, 1 (2013), 112–137.
[4] C. Berkholz. 2017. The Relation between Polynomial Calculus, Sherali-Adams, and Sum-of-Squares Proofs. *ECCC* (2017).
[5] C. Berkholz and M. Grohe. 2015. Limitations of Algebraic Approaches to Graph Isomorphism Testing. In *ICALP*. 155–166.
[6] A. Blass, Y. Gurevich, and S. Shelah. 2002. On polynomial time computation over unordered structures. *J. Symbolic Logic* 67, 3 (2002), 1093–1125.
[7] A. Dawar and P. Wang. 2017. Definability of semidefinite programming and Lasserre lower bounds for CSPs. In *LICS*. 1–12.
[8] E. Grädel, M. Grohe, B. Pago, and W. Pakusa. 2018. A Finite-Model-Theoretic View on Propositional Proof Complexity. *CoRR* abs/1802.09377 (2018).
[9] M. Grötschel, L. Lovász, and A. Schrijver. 1993. *Geometric Algorithms and Combinatorial Optimization.* Springer-Verlag.
[10] C. Josz and Didier Henrion. 2016. Strong duality in Lasserre's hierarchy for polynomial optimization. *Optimization Letters* 10, 1 (2016), 3–10.
[11] J. B. Lasserre. 2001. Global Optimization with Polynomials and the Problems of Moments. *SIAM Journal on Optimization* 11, 3 (2001), 796–817.
[12] P. N. Malkin. 2014. Sherali–Adams relaxations of graph isomorphism polytopes. *Discrete Optimization* 12 (2014), 73–97.
[13] R. O'Donnell, J. Wright, C. Wu, and Y. Zhou. 2014. Hardness of Robust Graph Isomorphism, Lasserre Gaps, and Asymmetry of Random Graphs. In *SODA*. 1659–1677.
[14] M. Otto. 1997. *Bounded variable logics and counting – A study in finite models.* Vol. 9. Springer-Verlag.
[15] H. D. Sherali and W. P. Adams. 1990. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM Journal on Discrete Mathematics* 3, 3 (1990), 411–430.
[16] S. P. Tarasov and M. N. Vyalyi. 2008. Semidefinite programming and arithmetic circuit evaluation. *Discrete Appl. Math.* 156, 11 (2008), 2070–2078.
[17] G. Tinhofer. 1986. Graph isomorphism and theorems of Birkhoff type. *Computing* 36, 4 (1986), 285–300.