SONC Optimization and Exact Nonnegativity Certificates via Second-Order Cone Programming

Victor Magron

CNRS LAAS, 7 avenue du colonel Roche 31400 Toulouse; Institute of Mathematics, 31400 Toulouse France.

Jie Wang

CNRS LAAS, 7 avenue du colonel Roche 31400 Toulouse France

Abstract

The second-order cone (SOC) is a class of simple convex cones and optimizing over them can be done more efficiently than with semidefinite programming. It is interesting both in theory and in practice to investigate which convex cones admit a representation using SOCs, given that they have a strong expressive ability. In this paper, we prove constructively that the cone of sums of nonnegative circuits (SONC) admits a SOC representation. Based on this, we give a new algorithm for unconstrained polynomial optimization via SOC programming. We also provide a hybrid numeric-symbolic scheme which combines the numerical procedure with a rounding-projection algorithm to obtain exact nonnegativity certificates. Numerical experiments demonstrate the efficiency of our algorithm for polynomials with fairly large degree and number of variables.

Keywords: sum of nonnegative circuit polynomials, second-order cone representation, second-order cone programming, polynomial optimization, sum of binomial squares, rounding-projection algorithm, exact nonnegativity certificate

1. Introduction

A *circuit polynomial* is of the form $\sum_{\alpha\in\mathscr{T}} c_\alpha \mathbf{x}^\alpha - d\mathbf{x}^\beta \in \mathbb{R}[\mathbf{x}] = \mathbb{R}[x_1, \ldots, x_n]$, where $c_\alpha > 0$ for all $\alpha \in \mathscr{T}$, $\mathscr{T} \subseteq (2\mathbb{N})^n$ comprises the vertices of a simplex and β lies in the relative interior of this simplex. The set of *sums of nonnegative circuit polynomials (SONC)* was introduced by Iliman and De Wolff (2016a) as a new certificate of nonnegativity for sparse polynomials, which is independent of the well-known set of sums of squares (SOS). Another recently introduced alternative certificates by Chandrasekaran and Shah (2016) are sums of arithmetic-geometric-exponentials (SAGE), which can be obtained via relative entropy programming. The connection between SONC and SAGE polynomials has been recently studied in Murray et al. (2021); Wang (2018); Katthän et al. (2021). It happens that SONC polynomials and SAGE polynomials are

Email addresses: `vmagron@laas.fr` (Victor Magron), `jwang@laas.fr` (Jie Wang)
URL: `https://homepages.laas.fr/vmagron` (Victor Magron),
`https://wangjie212.github.io/jiewang` (Jie Wang)

actually equivalent, as proved by Wang (2018) and Murray et al. (2021), and that both have a cancellation-free representation in terms of generators; see prior research by Wang (2018); Murray et al. (2021).

One of the significant differences between SONC and SOS is that SONC decompositions preserve sparsity of polynomials while SOS decompositions of sparse polynomials are not necessarily sparse; see Wang (2018). The set of SONC polynomials with a given support forms a convex cone, called a *SONC cone*. Optimization problems over SONC cones can be formulated as geometric programs or more generally relative entropy programs. We refer the interested reader to Iliman and De Wolff (2016b) for the unconstrained case and to Dressler et al. (2019) for the constrained case. Numerical experiments for unconstrained POPs (polynomial optimization problems) in Seidler and de Wolff (2018) have demonstrated the advantage of the SONC-based methods compared to the SOS-based methods, especially in the high-degree but fairly sparse case. Another recent framework by Dressler et al. (2020) relies on the dual SONC cone to compute lower bounds of polynomials by means of linear programming instead of geometric programming. As emphasized in the numerical comparison from Dressler et al. (2020), there are no general guarantees that the bounds obtained with this framework are always more or less accurate than the approach based on geometric programming from Seidler and de Wolff (2018), and the same holds for performance.

In the SOS case, there have been several attempts to exploit sparsity occurring in POPs. The sparse variant developed by Waki et al. (2006) of the moment-SOS hierarchy exploits the correlative sparsity pattern among the input variables to reduce the support of the resulting SOS decompositions. Such sparse representation results have been successfully applied in many fields, such as optimal power flow by Josz (2016), roundoff error bounds by Magron et al. (2017) and recently extended to the noncommutative case by Klep et al. (2021). Another way to exploit sparsity is to consider patterns based on terms, rather than variables, yielding an alternative sparse variant of Lasserre's hierarchy. The resulting *term-sparsity* SOS (TSSOS) hierarchy has been developed for polynomial optimization by Wang et al. (2021b,c,a), combined with correlative sparsity in Wang et al. (2020), and extended to the noncommutative case in Wang and Magron (2020a).

One of the similar features shared by SOS/SONC-based frameworks is their intrinsic connections with conic programming: SOS decompositions are computed via semidefinite programming and SONC decompositions via geometric programming. In both cases, the resulting optimization problems are solved with interior-point algorithms, thus output approximate nonnegativity certificates. However, one can still obtain an exact certificate from such output via hybrid numerical-symbolic algorithms when the input polynomial lies in the interior of the SOS/SONC cone. One way is to rely on rounding-projection algorithms adapted to the SOS cone by Peyrl and Parrilo (2008) and the SONC cone by Magron et al. (2019b), or alternatively on perturbation-compensation schemes as in Magron et al. (2019a); Magron and Safey El Din (2018c). The latter algorithms are implemented within the `RealCertify` library by Magron and Safey El Din (2018a).

In this paper, we study the second-order cone representation of SONC cones. An n-dimensional *(rotated) second-order cone (SOC)* is defined as

$$\mathbf{K} := \{\boldsymbol{a} = (a_i)_{i=1}^n \in \mathbb{R}^n \mid 2a_1a_2 \geq \sum_{i=3}^n a_i^2, a_1 \geq 0, a_2 \geq 0\}.$$

The SOC is well-studied and has mature solvers. Optimizing via second-order cone programming (SOCP) can be handled more efficiently than with semidefinite programming; see the work by Ahmadi and Majumdar (2019) and Alizadeh and Goldfarb (2003). On the other hand, despite the simplicity of SOCs, they have a strong ability to express other convex cones. Many such examples can be found in Section 3.3 from Ben-Tal and Nemirovski (2001). Therefore, it is interesting in theory and also important from the view of applications to investigate which convex cones can be expressed by SOCs.

Given sets of lattice points $\mathscr{A} \subseteq (2\mathbb{N})^n$, $\mathscr{B}_1 \subseteq \mathrm{conv}(\mathscr{A}) \cap (2\mathbb{N})^n$ and $\mathscr{B}_2 \subseteq \mathrm{conv}(\mathscr{A}) \cap (\mathbb{N}^n \backslash (2\mathbb{N})^n)$ ($\mathrm{conv}(\mathscr{A})$ is the convex hull of $\mathscr{A}$) with $\mathscr{A} \cap \mathscr{B}_1 = \varnothing$, let $\mathrm{SONC}_{\mathscr{A},\mathscr{B}_1,\mathscr{B}_2}$ be the SONC cone supported on $\mathscr{A}, \mathscr{B}_1, \mathscr{B}_2$ (see Definition 14). The first main result of this paper is the following theorem.

Theorem 1. *For $\mathscr{A} \subseteq (2\mathbb{N})^n$, $\mathscr{B}_1 \subseteq \mathrm{conv}(\mathscr{A}) \cap (2\mathbb{N})^n$ and $\mathscr{B}_2 \subseteq \mathrm{conv}(\mathscr{A}) \cap (\mathbb{N}^n \backslash (2\mathbb{N})^n)$ with $\mathscr{A} \cap \mathscr{B}_1 = \varnothing$, the convex cone $\mathrm{SONC}_{\mathscr{A},\mathscr{B}_1,\mathscr{B}_2}$ admits a SOC representation.*

The fact that SONC cones admit a SOC characterization was firstly proven by Averkov (2019) (see Theorem 17 for more details). However, Averkov's result is more theoretical. Even though Averkov's proof theoretically allows one to construct a SOC representation for a SONC cone, the construction is complicated and was not explicitly given in Averkov's paper. Our proof of Theorem 1, which involves writing a SONC polynomial as a sum of binomial squares with rational exponents (Theorem 12), is totally different from Averkov's and leads to a more concise (hence more efficient) SOC representation for SONC cones. This representation also enables us to propose a practical algorithm, based on SOCP, providing SONC decompositions for a certain class of nonnegative polynomials, which in turn yields lower bounds for unconstrained POPs. We emphasize that the number of SOCs used in the algorithm is *linear* in the number of terms (Lemma 29). We test the algorithm on various randomly generated polynomials up to a fairly large size, involving $n \sim 40$ variables and of degree $d \sim 60$. The numerical results demonstrate the efficiency of our algorithm.

The rest of this paper is organized as follows. In Section 2, we list some preliminaries on SONC polynomials. In Section 3, we reveal a key connection between SONC polynomials and sums of binomial squares by introducing the notion of $\mathscr{T}$-rational mediated sets. By virtue of this connection, we obtain second-order cone representations for SONC cones in Section 4. In Section 5, we provide a numerical procedure for unconstrained polynomial optimization via SOCP. A hybrid numerical-symbolic certification algorithm, based on combining this numerical procedure with a symbolic rounding-projection scheme is given and analyzed in Section 6. We evaluate the performance of the resulting certification algorithm in Section 7.

This paper is the follow-up of our previous contribution (Wang and Magron, 2020b), published in the proceedings of ISSAC'20. The main theoretical and practical novelties are threefold: (1) we design a hybrid numeric-symbolic algorithm to compute exact nonnegativity certificates, (2) provide explicit bounds for its arithmetic complexity analysis and (3) compare its efficiency with our numerical optimization procedure. Another more minor novelty is that we provide detailed statements for the algorithms to build rational mediated sets and simplex covers, respectively in Section 3.1 and Section 5.

2. Preliminaries

Let $\mathbb{R}[\mathbf{x}] = \mathbb{R}[x_1, \ldots, x_n]$ be the ring of real n-variate polynomials, and let $\mathbb{R}_+$ be the set of positive real numbers. For a finite set $\mathscr{A} \subseteq \mathbb{N}^n$, we denote by $\mathrm{conv}(\mathscr{A})$ the convex hull of $\mathscr{A}$.

Given a finite set $\mathscr{A} \subseteq \mathbb{N}^n$, we consider polynomials $f \in \mathbb{R}[\mathbf{x}]$ supported on $\mathscr{A} \subseteq \mathbb{N}^n$, i.e., f is of the form $f(\mathbf{x}) = \sum_{\alpha\in\mathscr{A}} f_\alpha \mathbf{x}^\alpha$ with $f_\alpha \in \mathbb{R}, \mathbf{x}^\alpha = x_1^{\alpha_1}\cdots x_n^{\alpha_n}$. The support of f is $\mathrm{supp}(f) := \{\boldsymbol{\alpha} \in \mathscr{A} \mid f_\alpha \neq 0\}$ and the Newton polytope of f is defined as $\mathrm{New}(f) := \mathrm{conv}(\mathrm{supp}(f))$. For a polytope P, we use $V(P)$ to denote the vertex set of P and use P° to denote the interior of P. For a set A, we use $\#A$ to denote the cardinality of A. A polynomial $f \in \mathbb{R}[\mathbf{x}]$ which is nonnegative over $\mathbb{R}^n$ is called a *nonnegative polynomial*, or a *positive semi-definite (PSD) polynomial*. The following definition of circuit polynomials was proposed by Iliman and De Wolff in Iliman and De Wolff (2016a).

Definition 2. *A polynomial $f \in \mathbb{R}[\mathbf{x}]$ is called a* circuit polynomial *if it is of the form $f(\mathbf{x}) = \sum_{\alpha\in\mathscr{T}} c_\alpha \mathbf{x}^\alpha - d\mathbf{x}^\beta$ and satisfies the following conditions:*

(i) $\mathscr{T} \subseteq (2\mathbb{N})^n$ *comprises the vertices of a simplex;*

(ii) $c_\alpha > 0$ *for each* $\boldsymbol{\alpha} \in \mathscr{T}$*;*

(iii) $\boldsymbol{\beta} \in \mathrm{conv}(\mathscr{T})^\circ \cap \mathbb{N}^n$.

The support $(\mathscr{T}, \boldsymbol{\beta})$ satisfying $(i), (iii)$ is called a circuit.

If $f = \sum_{\alpha\in\mathscr{T}} c_\alpha \mathbf{x}^\alpha - d\mathbf{x}^\beta$ is a circuit polynomial, then from the definition we can uniquely write $\boldsymbol{\beta} = \sum_{\alpha\in\mathscr{T}} \lambda_\alpha \boldsymbol{\alpha}$ with $\lambda_\alpha > 0$ and $\sum_{\alpha\in\mathscr{T}} \lambda_\alpha = 1$. We define the corresponding *circuit number* as $\Theta_f := \prod_{\alpha\in\mathscr{T}} (c_\alpha/\lambda_\alpha)^{\lambda_\alpha}$. The nonnegativity of the circuit polynomial f is decided by its circuit number alone, that is, f is nonnegative if and only if either $\boldsymbol{\beta} \notin (2\mathbb{N})^n$ and $|d| \leq \Theta_f$, or $\boldsymbol{\beta} \in (2\mathbb{N})^n$ and $d \leq \Theta_f$, by Theorem 3.8 from Iliman and De Wolff (2016a). To provide a concise narrative, we refer to a nonnegative circuit polynomial by a nonnegative circuit and also view a monomial square as a nonnegative circuit. An explicit representation of a polynomial being a *sum of nonnegative circuits*, or *SONC* for short, provides a certificate for its nonnegativity. Such a certificate is called a *SONC decomposition*. A polynomial that admits a SONC decomposition is called a *SONC polynomial*. For simplicity, we denote the set of SONC polynomials by SONC.

For a polynomial $f = \sum_{\alpha\in\mathscr{A}} f_\alpha \mathbf{x}^\alpha \in \mathbb{R}[\mathbf{x}]$, let $\Lambda(f) := \{\boldsymbol{\alpha} \in \mathscr{A} \mid \boldsymbol{\alpha} \in (2\mathbb{N})^n \text{ and } f_\alpha > 0\}$ and $\Gamma(f) := \mathrm{supp}(f)\backslash\Lambda(f)$. Then we can write f as $f = \sum_{\alpha\in\Lambda(f)} c_\alpha \mathbf{x}^\alpha - \sum_{\beta\in\Gamma(f)} d_\beta \mathbf{x}^\beta$. For each $\boldsymbol{\beta} \in \Gamma(f)$, let

$$\mathscr{C}(\boldsymbol{\beta}) := \{\mathscr{T} \mid \mathscr{T} \subseteq \Lambda(f) \text{ and } (\mathscr{T}, \boldsymbol{\beta}) \text{ is a circuit}\}. \tag{1}$$

As a consequence of Theorem 5.5 from Wang (2018), if $f \in$ SONC then it has a decomposition

$$f = \sum_{\beta\in\Gamma(f)} \sum_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} f_{\mathscr{T}\beta} + \sum_{\alpha\in\tilde{\mathscr{A}}} c_\alpha \mathbf{x}^\alpha, \tag{2}$$

where $f_{\mathscr{T}\beta}$ is a nonnegative circuit polynomial supported on $\mathscr{T} \cup \{\boldsymbol{\beta}\}$ and $\tilde{\mathscr{A}} = \{\boldsymbol{\alpha} \in \Lambda(f) \mid \boldsymbol{\alpha} \notin \cup_{\beta\in\Gamma(f)} \cup_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} \mathscr{T}\}$.

3. SONC and sums of binomial squares

In this section, we give a characterization of SONC polynomials in terms of sums of binomial squares (SOBS) with rational exponents.

3.1. Rational mediated sets

A lattice point $\boldsymbol{\alpha} \in \mathbb{N}^n$ is *even* if it is in $(2\mathbb{N})^n$. For a subset $M \subseteq \mathbb{N}^n$, define $\overline{A}(M) := \{\frac{1}{2}(\boldsymbol{v}+\boldsymbol{w}) \mid \boldsymbol{v} \neq \boldsymbol{w}, \boldsymbol{v}, \boldsymbol{w} \in M \cap (2\mathbb{N})^n\}$ as the set of averages of distinct even points in M. A subset $\mathscr{T} \subseteq (2\mathbb{N})^n$ is called a *trellis* if $\mathscr{T}$ comprises the vertices of a simplex. For a trellis $\mathscr{T}$, we call M a $\mathscr{T}$*-mediated set* if $\mathscr{T} \subseteq M \subseteq \overline{A}(M) \cup \mathscr{T}$; see the work on mediated sets by Hartzer et al. (2020); Powers and Reznick (2021); Reznick (1989).

Theorem 3. *Let $f = \sum_{\alpha \in \mathscr{T}} c_\alpha \mathbf{x}^\alpha - d\mathbf{x}^\beta \in \mathbb{R}[\mathbf{x}]$ with $d \neq 0$ be a nonnegative circuit. Then f is a sum of binomial squares if and only if there exists a $\mathscr{T}$-mediated set containing $\boldsymbol{\beta}$. Moreover, suppose that $\boldsymbol{\beta}$ belongs to a $\mathscr{T}$-mediated set M and for each $\boldsymbol{u} \in M \backslash \mathscr{T}$, let us write $\boldsymbol{u} = \frac{1}{2}(\boldsymbol{v_u}+\boldsymbol{w_u})$ for some $\boldsymbol{v_u} \neq \boldsymbol{w_u} \in M \cap (2\mathbb{N})^n$. Then one has the decomposition $f = \sum_{\boldsymbol{u} \in M\backslash \mathscr{T}} (a_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{v_u}} - b_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{w_u}})^2$, with $a_{\boldsymbol{u}}, b_{\boldsymbol{u}} \in \mathbb{R}$.*

Proof: It follows from Theorem 5.2 in Iliman and De Wolff (2016a). □

By Theorem 3, if we want to represent a nonnegative circuit polynomial as a sum of binomial squares, we need to first decide if there exists a $\mathscr{T}$-mediated set containing a given lattice point and then to compute one if there exists. However, there are obstacles for each of these two steps: (1) there may not exist such a $\mathscr{T}$-mediated set containing a given lattice point; (2) even if such a set exists, there is no efficient algorithm to compute it. In order to overcome these two difficulties, we introduce the concept of $\mathscr{T}$-rational mediated sets as a replacement of $\mathscr{T}$-mediated sets by admitting rational numbers in coordinates.

Concretely, for a subset $M \subseteq \mathbb{Q}^n$, let us define $\widetilde{A}(M) := \{\frac{1}{2}(\boldsymbol{v}+\boldsymbol{w}) \mid \boldsymbol{v} \neq \boldsymbol{w}, \boldsymbol{v}, \boldsymbol{w} \in M\}$ as the set of averages of distinct rational points in M. Let us assume that $\mathscr{T} \subseteq \mathbb{Q}^n$ comprises the vertices of a simplex. We say that M is a $\mathscr{T}$*-rational mediated set* if $\mathscr{T} \subseteq M \subseteq \widetilde{A}(M) \cup \mathscr{T}$. We shall see that for a trellis $\mathscr{T}$ and a lattice point $\boldsymbol{\beta} \in \text{conv}(\mathscr{T})^\circ$, a $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$ always exists and moreover, there is an effective algorithm to compute it.

First, let us consider the one dimensional case. For a sequence of integer numbers $A = \{s, q_1, \ldots, q_m, p\}$ (arranged from small to large), if every q_i is an average of two distinct numbers in A, then we say A is an (s, p)*-mediated sequence*. Note that the property of (s, p)-mediated sequences is preserved under translations, that is, there is a one-to-one correspondence between (s, p)-mediated sequences and $(s + r, p + r)$-mediated sequences for any integer number r. So it suffices to consider the case of $s = 0$.

For a fixed p and an integer q with $0 < q < p$, a *minimal* $(0, p)$*-mediated sequence* containing q is a $(0, p)$-mediated sequence containing q with the least number of elements.

Example 4. *Consider the set $A = \{0, 2, 4, 5, 8, 11\}$. One can easily check by hand that A is a minimal $(0, 11)$-mediated sequence containing $2, 4, 5, 8$.*

Denote the number of elements in a minimal $(0, p)$-mediated sequence containing q by $N(\frac{q}{p})$. One can then easily show that $N(\frac{1}{p}) = \lceil \log_2(p) \rceil + 2$ by induction on p. We conjecture that this formula holds for general q.

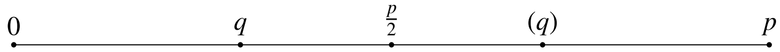
Conjecture 5. *If $\gcd(p, q) = 1$, then $N(\frac{q}{p}) = \lceil \log_2(p) \rceil + 2$.*

Generally we do not know how to compute a minimal $(0, p)$-mediated sequence containing a given q. However, we can design a procedure to compute an approximately minimal $(0, p)$-mediated sequence containing a given q, based on the following lemma.

Lemma 6. *For integers p, q such that $0 < q < p$, there exists a $(0, p)$-mediated sequence containing q with the cardinality less than $\frac{1}{2}(\log_2(p) + \frac{3}{2})^2$.*

Proof: We can assume $\gcd(p, q) = 1$ (otherwise one can consider $p/\gcd(p, q), q/\gcd(p, q)$ instead). Let us do induction on p. Assume that for any $p', q' \in \mathbb{N}, 0 < q' < p' < p$, there exists a $(0, p')$-mediated sequence containing q' with the number of elements less than $\frac{1}{2}(\log_2(p') + \frac{3}{2})^2$.

Case 1: Suppose that p is an even number. If $q = \frac{p}{2}$, then by $\gcd(p, q) = 1$, we have $q = 1$ and $A = \{0, 1, 2\}$ is a $(0, p)$-mediated sequence containing q. Otherwise, we have either $0 < q < \frac{p}{2}$ or $\frac{p}{2} < q < p$. For $0 < q < \frac{p}{2}$, by the induction hypothesis, there exists a $(0, \frac{p}{2})$-mediated sequence A' containing q. For $\frac{p}{2} < q < p$, since the property of mediated sequences is preserved under translations, one can first subtract $\frac{p}{2}$ and obtain a $(0, \frac{p}{2})$-mediated sequence containing $q - \frac{p}{2}$ by the induction hypothesis. Then by adding $\frac{p}{2}$, one obtains a $(\frac{p}{2}, p)$-mediated sequence A' containing q. It follows that $A = A' \cup \{p\}$ or $A = \{0\} \cup A'$ is a $(0, p)$-mediated sequence containing q.



Moreover, we have

$$\#A = 1 + \#A' < 1 + \frac{1}{2}(\log_2(\frac{p}{2}) + \frac{3}{2})^2 < \frac{1}{2}(\log_2(p) + \frac{3}{2})^2.$$

Case 2: Suppose that p is an odd number. Without loss of generality, assume that q is an even number (otherwise one can consider $p - q$ instead and then obtain a $(0, p)$-mediated sequence containing q through the map $x \mapsto p - x$ which clearly preserves the property of mediated sequences).

Let $q = 2^k r$ for some $k, r \in \mathbb{N}\backslash\{0\}$ and $2 \nmid r$. If $q = p - r$, then $q = \frac{q-r+p}{2}$. Since $\gcd(p, q) = 1$, we have $r = 1$. Let $A = \{0, \frac{1}{2}q, \frac{3}{4}q, \ldots, (1 - \frac{1}{2^k})q, q, p\}$. For $i = 1, \ldots, k$, we have $(1 - \frac{1}{2^i})q = \frac{1}{2}(1 - \frac{1}{2^{i-1}})q + \frac{1}{2}q$. Therefore, A is a $(0, p)$-mediated sequence containing q.

Moreover, we have

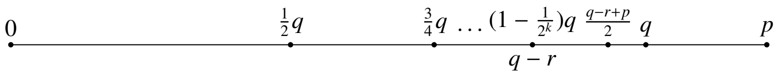
$$\#A = k + 3 < \frac{1}{2}(\log_2(2^k + 1) + \frac{3}{2})^2 = \frac{1}{2}(\log_2(p) + \frac{3}{2})^2.$$

If $q < p - r$, then q lies on the line segment between $q - r$ and $\frac{q-r+p}{2}$. Since $\frac{q-r+p}{2} - (q - r) = \frac{p+r-q}{2} < p$, then by the induction hypothesis, there exists a $(q - r, \frac{q-r+p}{2})$-mediated sequence A' containing q (using translations). It follows that $A = \{0, \frac{1}{2}q, \frac{3}{4}q, \ldots, (1 - \frac{1}{2^{k-1}})q, p\} \cup A'$ is a $(0, p)$-mediated sequence containing q.

Moreover, we have

$$\begin{aligned}\#A = k + 1 + \#A' &< \log_2(\frac{q}{r}) + 1 + \frac{1}{2}(\log_2(\frac{p+r-q}{2}) + \frac{3}{2})^2 \\ &< \log_2(p) + 1 + \frac{1}{2}(\log_2(\frac{p}{2}) + \frac{3}{2})^2 \\ &= \frac{1}{2}(\log_2(p) + \frac{3}{2})^2.\end{aligned}$$

If $q > p-r$, then q lies on the line segment between $\frac{q-r+p}{2}$ and p. Since $p-\frac{q-r+p}{2} = \frac{p+r-q}{2} < p$, then by the induction hypothesis, there exists a $(\frac{q-r+p}{2}, p)$-mediated sequence A' containing q (using translations). It follows that the set $A = \{0, \frac{1}{2}q, \frac{3}{4}q, \ldots, (1-\frac{1}{2^k})q\} \cup A'$ is a $(0, p)$-mediated sequence containing q.



As previously, we have $\#A = k + 1 + \#A' < \frac{1}{2}(\log_2(p) + \frac{3}{2})^2$. $\square$

According to the proof of Lemma 6, the procedure to compute an approximately minimal $(0, p)$-mediated sequence containing a given q is stated as Algorithm 1.

Lemma 7. *Suppose that $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$ are two rational points, and $\boldsymbol{\beta}$ is any rational point on the line segment between $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$. Then there exists an $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\}$-rational mediated set M containing $\boldsymbol{\beta}$. Furthermore, if the denominators of coordinates of $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\beta}$ are odd numbers, and the numerators of coordinates of $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2$ are even numbers, then we can ensure that the denominators of coordinates of points in M are odd numbers and the numerators of coordinates of points in $M\backslash\{\boldsymbol{\beta}\}$ are even numbers.*

Proof: Suppose $\boldsymbol{\beta} = (1-\frac{q}{p})\boldsymbol{\alpha}_1 + \frac{q}{p}\boldsymbol{\alpha}_2$, $p, q \in \mathbb{N}, 0 < q < p, \gcd(p, q) = 1$. We then construct a one-to-one correspondence between the points on the one-dimensional number axis and the points on the line across $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$ via the map: $s \mapsto (1-\frac{s}{p})\boldsymbol{\alpha}_1 + \frac{s}{p}\boldsymbol{\alpha}_2$, such that $\boldsymbol{\alpha}_1$ corresponds to the origin, $\boldsymbol{\alpha}_2$ corresponds to p and $\boldsymbol{\beta}$ corresponds to q. Then it is clear that a $(0, p)$-mediated sequence containing q corresponds to a $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\}$-rational mediated set containing $\boldsymbol{\beta}$. Hence by Lemma 6, there exists a $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\}$-rational mediated set M containing $\boldsymbol{\beta}$ with the number of elements less than $\frac{1}{2}(\log_2(p) + \frac{3}{2})^2$. Moreover, we can see that if $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\beta}$ are lattice points, then the elements in M are also lattice points.

If the denominators of coordinates of $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\beta}$ are odd numbers, and the numerators of coordinates of $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2$ are even numbers, assume that the least common multiple of denominators appearing in the coordinates of $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\beta}$ is r and then remove the denominators by multiplying the coordinates of $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\beta}$ by r such that $r\boldsymbol{\alpha}_1, r\boldsymbol{\alpha}_2$ are even lattice points. If $r\boldsymbol{\beta}$ is even, let M' be the $\{\frac{r}{2}\boldsymbol{\alpha}_1, \frac{r}{2}\boldsymbol{\alpha}_2\}$-rational mediated set containing $\frac{r}{2}\boldsymbol{\beta}$ obtained as above (the elements in M' are lattice points). Then $M = \frac{2}{r}M' := \{\frac{2}{r}\boldsymbol{u} \mid \boldsymbol{u} \in M'\}$ is an $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\}$-rational mediated set containing $\boldsymbol{\beta}$ such that the denominators of coordinates of points in M are odd numbers and the numerators of coordinates of points in $M\backslash\{\boldsymbol{\beta}\}$ are even numbers.

If $r\boldsymbol{\beta}$ is not even, assume without loss of generality that $\boldsymbol{\beta}$ lies on the line segment between $\boldsymbol{\alpha}_1$ and $\frac{\boldsymbol{\alpha}_1+\boldsymbol{\alpha}_2}{2}$. Let $\boldsymbol{\beta}' = 2\boldsymbol{\beta} - \boldsymbol{\alpha}_1$ with $r\boldsymbol{\beta}'$ an even lattice point. Let M' be the $\{\frac{r}{2}\boldsymbol{\alpha}_1, \frac{r}{2}\boldsymbol{\alpha}_2\}$-rational mediated set containing $\frac{r}{2}\boldsymbol{\beta}'$ obtained as above (note that the elements in M' are lattice points).

Algorithm 1 `MedSeq`(p, q)

Input: $p, q \in \mathbb{N}, 0 < q < p$

Output: A set of triples $\{(u_i, v_i, w_i)\}_i$ with $u_i = \frac{1}{2}(v_i + w_i)$ such that $\{0, p\} \cup \{u_i\}_i$ is a $(0, p)$-mediated sequence containing q

$u := p$, $v := q$, $w :=$gcd(p, q);
$u := \frac{u}{w}, v := \frac{v}{w}$;
if $2|u$ **then**
  if $v = \frac{u}{2}$ **then**
    $A := \{(1, 0, 2)\}$;
  else
    if $v < u/2$ **then**
      $A :=$ `MedSeq`$(\frac{u}{2}, v) \cup \{(\frac{u}{2}, 0, u)\}$;
    else
      $A := \{(\frac{u}{2}, 0, u)\} \cup ($`MedSeq`$(\frac{u}{2}, v - \frac{u}{2}) + \frac{u}{2})$;
    end if
  end if
else
  if $2|v$ **then**
    Let $k, r \in \mathbb{N}\backslash\{0\}$ such that $v = 2^k r$ and $2 \nmid r$;
    if $v = u - r$ **then**
      $A := \{(\frac{1}{2}v, 0, v), (\frac{3}{4}v, \frac{1}{2}v, v), \ldots, (v, v - r, u)\}$;
    else
      if $v < u - r$ **then**
        $A := \{(\frac{1}{2}v, 0, v), \ldots, (\frac{v-r+u}{2}, v - r, u)\} \cup ($`MedSeq`$(\frac{u+r-v}{2}, r) + v - r)$;
      else
        $A := \{(\frac{1}{2}v, 0, v), \ldots, (\frac{v-r+u}{2}, v - r, u)\} \cup ($`MedSeq`$(\frac{u+r-v}{2}, \frac{v+r-u}{2}) + \frac{v+u-r}{2})$;
      end if
    end if
  else
    $A := u -$ `MedSeq`$(u, u - v)$;
  end if
end if
return wA;

Then $M = \frac{2}{r}M' \cup \{\boldsymbol{\beta}\}$ is an $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\}$-rational mediated set containing $\boldsymbol{\beta}$ such that the denominators of coordinates of points in M are odd numbers and the numerators of coordinates of points in $M\backslash\{\boldsymbol{\beta}\}$ are even numbers as desired. $\square$

The procedure based on Lemma 7 to obtain an $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\}$-rational mediated set containing $\boldsymbol{\beta}$ is stated in Algorithm 2.

Algorithm 2 `LMedSet`$(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\beta})$

Input: $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\beta} \in \mathbb{Q}^n$ such that $\boldsymbol{\beta}$ lies on the line segment between $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$

Output: A sequence of triples $\{(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i)\}_i$ with $\boldsymbol{u}_i = \frac{1}{2}(\boldsymbol{v}_i + \boldsymbol{w}_i)$ such that $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\} \cup \{\boldsymbol{u}_i\}_i$ is a $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\}$-rational mediated set containing $\boldsymbol{\beta}$

1: Let $\boldsymbol{\beta} = (1 - \frac{q}{p})\boldsymbol{\alpha}_1 + \frac{q}{p}\boldsymbol{\alpha}_2$, $p, q \in \mathbb{N}, 0 < q < p, \gcd(p, q) = 1$;
2: $A :=$ `MedSeq`(p, q);
3: $M := \cup_{(u,v,w)\in A}\{((1 - \frac{u}{p})\boldsymbol{\alpha}_1 + \frac{u}{p}\boldsymbol{\alpha}_2, (1 - \frac{v}{p})\boldsymbol{\alpha}_1 + \frac{v}{p}\boldsymbol{\alpha}_2, (1 - \frac{w}{p})\boldsymbol{\alpha}_1 + \frac{w}{p}\boldsymbol{\alpha}_2)\}$;
4: **return** M;

Lemma 8. *For a trellis $\mathscr{T} = \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_m\}$ and a lattice point $\boldsymbol{\beta} \in \mathrm{conv}(\mathscr{T})^\circ$, there exists a $\mathscr{T}$-rational mediated set $M_{\mathscr{T}\boldsymbol{\beta}}$ containing $\boldsymbol{\beta}$ such that the denominators of coordinates of points in $M_{\mathscr{T}\boldsymbol{\beta}}$ are odd numbers and the numerators of coordinates of points in $M_{\mathscr{T}\boldsymbol{\beta}}\backslash\{\boldsymbol{\beta}\}$ are even numbers.*

Proof: Suppose $\boldsymbol{\beta} = \sum_{i=1}^m \frac{q_i}{p}\boldsymbol{\alpha}_i$, where $p = \sum_{i=1}^m q_i$, $p, q_i \in \mathbb{N}\backslash\{0\}$, $\gcd(p, q_1, \ldots, q_m) = 1$. If p is an even number, then because $\gcd(p, q_1, \ldots, q_m) = 1$, there must exist an odd number among the q_i's. Without loss of generality assume q_1 is an odd number. If p is an odd number and there exists an even number among the q_i's, then without loss of generality assume q_1 is an even number. In any of these two cases, we have

$$\boldsymbol{\beta} = \frac{q_1}{p}\boldsymbol{\alpha}_1 + \frac{p - q_1}{p}(\frac{q_2}{p - q_1}\boldsymbol{\alpha}_2 + \cdots + \frac{q_m}{p - q_1}\boldsymbol{\alpha}_m).$$

Let $\boldsymbol{\beta}_1 = \frac{q_2}{p-q_1}\boldsymbol{\alpha}_2 + \cdots + \frac{q_m}{p-q_1}\boldsymbol{\alpha}_m$. Then $\boldsymbol{\beta} = \frac{q_1}{p}\boldsymbol{\alpha}_1 + \frac{p-q_1}{p}\boldsymbol{\beta}_1$.

If p is an odd number and all q_i's are odd numbers, then we have

$$\begin{aligned}\boldsymbol{\beta} = &\frac{q_1}{q_1 + q_2}(\frac{q_1 + q_2}{p}\boldsymbol{\alpha}_1 + \frac{q_3}{p}\boldsymbol{\alpha}_3 + \cdots + \frac{q_m}{p}\boldsymbol{\alpha}_m)\\ &+ \frac{q_2}{q_1 + q_2}(\frac{q_1 + q_2}{p}\boldsymbol{\alpha}_2 + \frac{q_3}{p}\boldsymbol{\alpha}_3 + \cdots + \frac{q_m}{p}\boldsymbol{\alpha}_m).\end{aligned}$$

Let $\boldsymbol{\beta}_1 = \frac{q_1+q_2}{p}\boldsymbol{\alpha}_1 + \frac{q_3}{p}\boldsymbol{\alpha}_3 + \cdots + \frac{q_m}{p}\boldsymbol{\alpha}_m$ and $\boldsymbol{\beta}_2 = \frac{q_1+q_2}{p}\boldsymbol{\alpha}_2 + \frac{q_3}{p}\boldsymbol{\alpha}_3 + \cdots + \frac{q_m}{p}\boldsymbol{\alpha}_m$. Then $\boldsymbol{\beta} = \frac{q_1}{q_1+q_2}\boldsymbol{\beta}_1 + \frac{q_2}{q_1+q_2}\boldsymbol{\beta}_2$.

Apply the same procedure for $\boldsymbol{\beta}_1$ (and $\boldsymbol{\beta}_2$), and continue iteratively. Eventually we obtain a set of points $\{\boldsymbol{\beta}_i\}_{i=1}^l$ such that for each i, $\boldsymbol{\beta}_i = \lambda_i\boldsymbol{\beta}_j + \mu_i\boldsymbol{\beta}_k$ or $\boldsymbol{\beta}_i = \lambda_i\boldsymbol{\beta}_j + \mu_i\boldsymbol{\alpha}_k$ or $\boldsymbol{\beta}_i = \lambda_i\boldsymbol{\alpha}_j + \mu_i\boldsymbol{\alpha}_k$, where $\lambda_i + \mu_i = 1, \lambda_i, \mu_i > 0$. We claim that the denominators of coordinates of $\boldsymbol{\beta}_i$ are odd numbers, and the numerators of coordinates of $\boldsymbol{\beta}_i$ are even numbers. This is because for each $\boldsymbol{\beta}_i$, we have the expression $\boldsymbol{\beta}_i = \sum_j \frac{s_j}{r}\boldsymbol{\alpha}_j$, where r is an odd number and all $\boldsymbol{\alpha}_j$'s are even lattice points. For $\boldsymbol{\beta}_i = \lambda\boldsymbol{\beta}_j + \mu\boldsymbol{\beta}_k$ (or $\boldsymbol{\beta}_i = \lambda\boldsymbol{\beta}_j + \mu\boldsymbol{\alpha}_k$, $\boldsymbol{\beta}_i = \lambda\boldsymbol{\alpha}_j + \mu\boldsymbol{\alpha}_k$ respectively), let M_i be the $\{\boldsymbol{\beta}_j, \boldsymbol{\beta}_k\}$- (or $\{\boldsymbol{\beta}_j, \boldsymbol{\alpha}_k\}$-, $\{\boldsymbol{\alpha}_j, \boldsymbol{\alpha}_k\}$- respectively) rational mediated set containing $\boldsymbol{\beta}_i$ obtained by Lemma 7 such that the denominators of coordinates of points in M_i are odd numbers and the numerators of coordinates of points in $M_i \backslash \{\boldsymbol{\beta}\}$ are even numbers for $i = 0, \ldots, l$ (set $\boldsymbol{\beta}_0 = \boldsymbol{\beta}$). Let $M_{\mathscr{T}\boldsymbol{\beta}} = \cup_{i=0}^{l} M_i$. Then $M_{\mathscr{T}\boldsymbol{\beta}}$ is clearly a $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$ with the desired property. □

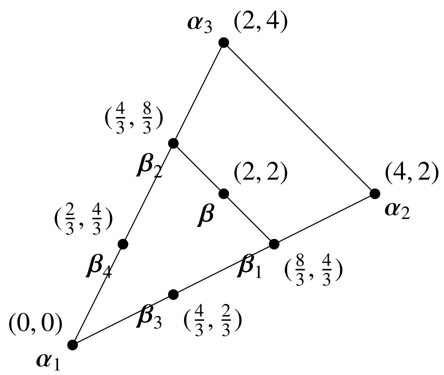
3.2. Decomposing SONC with binomial squares

For $r \in \mathbb{N}$ and $f(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$, let $f(\mathbf{x}^r) := f(x_1^r, \ldots, x_n^r)$. For an odd $r \in \mathbb{N}$, it is clear that $f(\mathbf{x}) = \sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\boldsymbol{\alpha}} \mathbf{x}^{\boldsymbol{\alpha}} - d\mathbf{x}^{\boldsymbol{\beta}}$ is a nonnegative circuit if and only if $f(\mathbf{x}^r) = \sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\boldsymbol{\alpha}} \mathbf{x}^{r\boldsymbol{\alpha}} - d\mathbf{x}^{r\boldsymbol{\beta}}$ is a nonnegative circuit.

Theorem 9. *Let $f = \sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\boldsymbol{\alpha}} \mathbf{x}^{\boldsymbol{\alpha}} - d\mathbf{x}^{\boldsymbol{\beta}} \in \mathbb{R}[\mathbf{x}]$ with $d \neq 0$ be a circuit polynomial. Assume that $M_{\mathscr{T}\boldsymbol{\beta}}$ is a $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$ provided by Lemma 8. For each $\boldsymbol{u} \in M_{\mathscr{T}\boldsymbol{\beta}} \backslash \mathscr{T}$, let $\boldsymbol{u} = \frac{1}{2}(\boldsymbol{v_u} + \boldsymbol{w_u}), \boldsymbol{v_u} \neq \boldsymbol{w_u} \in M_{\mathscr{T}\boldsymbol{\beta}}$. Then f is nonnegative if and only if f can be written as $f = \sum_{\boldsymbol{u}\in M_{\mathscr{T}\boldsymbol{\beta}}\backslash\mathscr{T}} (a_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{v_u}} - b_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{w_u}})^2$, $a_{\boldsymbol{u}}, b_{\boldsymbol{u}} \in \mathbb{R}$.*

Proof: Assume that the least common multiple of denominators appearing in the coordinates of points in $M_{\mathscr{T}\boldsymbol{\beta}}$ is r, which is odd. Then $f(\mathbf{x})$ is nonnegative if and only if $f(\mathbf{x}^r)$ is nonnegative. Multiply all coordinates of points in $M_{\mathscr{T}\boldsymbol{\beta}}$ by r to remove the denominators, and the obtained $rM_{\mathscr{T}\boldsymbol{\beta}} := \{r\boldsymbol{u} \mid \boldsymbol{u} \in M_{\mathscr{T}\boldsymbol{\beta}}\}$ is an $r\mathscr{T}$-mediated set containing $r\boldsymbol{\beta}$. Hence by Theorem 3, $f(\mathbf{x}^r)$ is nonnegative if and only if $f(\mathbf{x}^r)$ can be written as $f(\mathbf{x}^r) = \sum_{\boldsymbol{u}\in M_{\mathscr{T}\boldsymbol{\beta}}\backslash\mathscr{T}} (a_{\boldsymbol{u}} \mathbf{x}^{\frac{r}{2}\boldsymbol{v_u}} - b_{\boldsymbol{u}} \mathbf{x}^{\frac{r}{2}\boldsymbol{w_u}})^2$, $a_{\boldsymbol{u}}, b_{\boldsymbol{u}} \in \mathbb{R}$, which is equivalent to $f(\mathbf{x}) = \sum_{\boldsymbol{u}\in M_{\mathscr{T}\boldsymbol{\beta}}\backslash\mathscr{T}} (a_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{v_u}} - b_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{w_u}})^2$. □

Example 10. *Let $f = x^4y^2 + x^2y^4 + 1 - 3x^2y^2$ be the Motzkin polynomial and $\mathscr{T} = \{\boldsymbol{\alpha}_1 = (0,0), \boldsymbol{\alpha}_2 = (4,2), \boldsymbol{\alpha}_3 = (2,4)\}$, $\boldsymbol{\beta} = (2,2)$. Let $\boldsymbol{\beta}_1 = \frac{1}{3}\boldsymbol{\alpha}_1 + \frac{2}{3}\boldsymbol{\alpha}_2$ and $\boldsymbol{\beta}_2 = \frac{1}{3}\boldsymbol{\alpha}_1 + \frac{2}{3}\boldsymbol{\alpha}_3$ such that $\boldsymbol{\beta} = \frac{1}{2}\boldsymbol{\beta}_1 + \frac{1}{2}\boldsymbol{\beta}_2$. Let $\boldsymbol{\beta}_3 = \frac{2}{3}\boldsymbol{\alpha}_1 + \frac{1}{3}\boldsymbol{\alpha}_2$ and $\boldsymbol{\beta}_4 = \frac{2}{3}\boldsymbol{\alpha}_1 + \frac{1}{3}\boldsymbol{\alpha}_3$. Then $M = \{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha}_3, \boldsymbol{\beta}, \boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \boldsymbol{\beta}_3, \boldsymbol{\beta}_4\}$ is a $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$.*



By Theorem 9, one has $f = x^4y^2 + x^2y^4 + 1 - 3x^2y^2 = (a_1x^{\frac{2}{3}}y^{\frac{4}{3}} - b_1x^{\frac{4}{3}}y^{\frac{2}{3}})^2 + (a_2xy^2 - b_2x^{\frac{1}{3}}y^{\frac{2}{3}})^2 + (a_3x^{\frac{2}{3}}y^{\frac{4}{3}} - b_3)^2 + (a_4x^2y - b_4x^{\frac{2}{3}}y^{\frac{1}{3}})^2 + (a_5x^{\frac{4}{3}}y^{\frac{2}{3}} - b_5)^2$. Comparing coefficients yields $f = \frac{3}{2}(x^{\frac{2}{3}}y^{\frac{4}{3}} - x^{\frac{4}{3}}y^{\frac{2}{3}})^2 + (xy^2 - x^{\frac{1}{3}}y^{\frac{2}{3}})^2 + \frac{1}{2}(x^{\frac{2}{3}}y^{\frac{4}{3}} - 1)^2 + (x^2y - x^{\frac{2}{3}}y^{\frac{1}{3}})^2 + \frac{1}{2}(x^{\frac{4}{3}}y^{\frac{2}{3}} - 1)^2$, a sum of five binomial squares with rational exponents.

Lemma 11. *Let $f(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$. For an odd number r, $f(\mathbf{x}) \in$ SONC if and only if $f(\mathbf{x}^r) \in$ SONC.*

Proof: It comes from the fact that $f(\mathbf{x})$ is a nonnegative circuit if and only if $f(\mathbf{x}^r)$ is a nonnegative circuit for an odd number r. $\Box$

Theorem 12. *Let $f = \sum_{\alpha\in\Lambda(f)} c_\alpha \mathbf{x}^\alpha - \sum_{\beta\in\Gamma(f)} d_\beta \mathbf{x}^\beta \in \mathbb{R}[\mathbf{x}]$. Let $\mathscr{C}(\boldsymbol{\beta})$ be as in (1). For every $\boldsymbol{\beta} \in \Gamma(f)$ and every trellis $\mathscr{T} \in \mathscr{C}(\boldsymbol{\beta})$, let $M_{\mathscr{T}\boldsymbol{\beta}}$ be a $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$ provided by Lemma 8. Let $M = \cup_{\boldsymbol{\beta}\in\Gamma(f)} \cup_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} M_{\mathscr{T}\boldsymbol{\beta}}$. For each $\boldsymbol{u} \in M\backslash\Lambda(f)$, let $\boldsymbol{u} = \frac{1}{2}(\boldsymbol{v_u}+\boldsymbol{w_u}), \boldsymbol{v_u} \neq \boldsymbol{w_u} \in M$. Let $\tilde{\mathscr{A}} = \{\boldsymbol{\alpha} \in \Lambda(f) \mid \boldsymbol{\alpha} \notin \cup_{\boldsymbol{\beta}\in\Gamma(f)} \cup_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} \mathscr{T}\}$. Then $f \in$ SONC if and only if f can be written as $f = \sum_{\boldsymbol{u}\in M\backslash\Lambda(f)} (a_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{v_u}} - b_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{w_u}})^2 + \sum_{\alpha\in\tilde{\mathscr{A}}} c_\alpha \mathbf{x}^\alpha$, $a_{\boldsymbol{u}}, b_{\boldsymbol{u}} \in \mathbb{R}$.*

Proof: Suppose $f \in$ SONC. By Theorem 5.5 in Wang (2018), we can write f as

$$f = \sum_{\beta\in\Gamma(f)} \sum_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} f_{\mathscr{T}\boldsymbol{\beta}} + \sum_{\alpha\in\tilde{\mathscr{A}}} c_\alpha \mathbf{x}^\alpha ,$$

such that every $f_{\mathscr{T}\boldsymbol{\beta}}$ is a nonnegative circuit supported on $\mathscr{T} \cup \{\boldsymbol{\beta}\}$. We have

$$f_{\mathscr{T}\boldsymbol{\beta}} = \sum_{\boldsymbol{u}\in M_{\mathscr{T}\boldsymbol{\beta}}\backslash\mathscr{T}} (a_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{v_u}} - b_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{w_u}})^2 ,$$

for some $a_{\boldsymbol{u}}, b_{\boldsymbol{u}} \in \mathbb{R}$ by Theorem 9. Thus there exist $a_{\boldsymbol{u}}, b_{\boldsymbol{u}} \in \mathbb{R}$ such that

$$f = \sum_{\boldsymbol{u}\in M\backslash\Lambda(f)} (a_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{v_u}} - b_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{w_u}})^2 + \sum_{\alpha\in\tilde{\mathscr{A}}} c_\alpha \mathbf{x}^\alpha .$$

Now suppose that f has the desired form. Assume that the least common multiple of denominators appearing in the coordinates of points in M is an odd positive integer r. Then there exist $a_{\boldsymbol{u}}, b_{\boldsymbol{u}} \in \mathbb{R}$ such that

$$f(\mathbf{x}^r) = \sum_{\boldsymbol{u}\in M\backslash\Lambda(f)} (a_{\boldsymbol{u}} \mathbf{x}^{\frac{r}{2}\boldsymbol{v_u}} - b_{\boldsymbol{u}} \mathbf{x}^{\frac{r}{2}\boldsymbol{w_u}})^2 + \sum_{\alpha\in\tilde{\mathscr{A}}} c_\alpha \mathbf{x}^{r\alpha} .$$

Thus $f(\mathbf{x}^r) \in$ SONC since every binomial or monomial square is a nonnegative circuit. Hence by Lemma 11, $f(\mathbf{x}) \in$ SONC. $\Box$

4. SOC representations of SONC cones

SOCP plays an important role in convex optimization and can be handled via very efficient algorithms. If a SOC representation exists for a given convex cone, then it is possible to design efficient algorithms for optimization problems over the convex cone. In Fawzi (2019), Fawzi proved that PSD cones do not admit any SOC representations in general, which implies that SOS cones do not admit any SOC representations in general. In this section, we prove that dramatically unlike the SOS cones, SONC cones always admit SOC representations. Let us discuss it in more details. Let $\mathcal{Q}^k := \mathcal{Q} \times \cdots \times \mathcal{Q}$ be the Cartesian product of k copies of a SOC $\mathcal{Q}$. A *linear slice* of $\mathcal{Q}^k$ is an intersection of $\mathcal{Q}^k$ with a linear subspace.

Definition 13. *A convex cone $C \subseteq \mathbb{R}^m$ has a* SOC lift of size k *(or simply a $\mathcal{Q}^k$-*lift*) if it can be written as the projection of a slice of $\mathcal{Q}^k$, that is, there is a subspace L of $\mathcal{Q}^k$ and a linear map $\pi\colon \mathcal{Q}^k \to \mathbb{R}^m$ such that $C = \pi(\mathcal{Q}^k \cap L)$.*

We give the following definition of SONC cones supported on given lattice points.

Definition 14. *Given sets of lattice points* $\mathscr{A} \subseteq (2\mathbb{N})^n$, $\mathscr{B}_1 \subseteq \mathrm{conv}(\mathscr{A}) \cap (2\mathbb{N})^n$ *and* $\mathscr{B}_2 \subseteq \mathrm{conv}(\mathscr{A}) \cap (\mathbb{N}^n \backslash (2\mathbb{N})^n)$ *such that* $\mathscr{A} \cap \mathscr{B}_1 = \varnothing$, *define the SONC cone supported on* $\mathscr{A}, \mathscr{B}_1, \mathscr{B}_2$ *as*

$$\begin{aligned}\mathrm{SONC}_{\mathscr{A},\mathscr{B}_1,\mathscr{B}_2} := \{ & (\mathbf{c}_{\mathscr{A}}, \mathbf{d}_{\mathscr{B}_1}, \mathbf{d}_{\mathscr{B}_2}) \in \mathbb{R}_+^{|\mathscr{A}|} \times \mathbb{R}_+^{|\mathscr{B}_1|} \times \mathbb{R}^{|\mathscr{B}_2|} \\ & \mid \sum_{\alpha \in \mathscr{A}} c_\alpha \mathbf{x}^\alpha - \sum_{\beta \in \mathscr{B}_1 \cup \mathscr{B}_2} d_\beta \mathbf{x}^\beta \in \mathrm{SONC}\},\end{aligned}$$

where $\mathbf{c}_{\mathscr{A}} = (c_\alpha)_{\alpha \in \mathscr{A}}$, $\mathbf{d}_{\mathscr{B}_1} = (d_\beta)_{\beta \in \mathscr{B}_1}$ *and* $\mathbf{d}_{\mathscr{B}_2} = (d_\beta)_{\beta \in \mathscr{B}_2}$. *It is easy to check that* $\mathrm{SONC}_{\mathscr{A},\mathscr{B}_1,\mathscr{B}_2}$ *is indeed a convex cone.*

Let $\mathbb{S}_+^2$ be the convex cone of 2×2 positive semidefinite matrices

$$\mathbb{S}_+^2 := \left\{ \begin{bmatrix} a & b \\ b & c \end{bmatrix} \in \mathbb{R}^{2\times 2} \mid \begin{bmatrix} a & b \\ b & c \end{bmatrix} \text{ is positive semidefinite} \right\}.$$

Lemma 15. $\mathbb{S}_+^2$ *is a* 3*-dimensional rotated SOC.*

Proof: The condition for $\begin{bmatrix} a & b \\ b & c \end{bmatrix}$ to be a positive semidefinite matrix is $a \geq 0, c \geq 0, ac \geq b^2$. Thus $\mathbb{S}_+^2$ is a rotated SOC by the definition. □

Now we are ready to prove the main theorem.

Theorem 16. *For* $\mathscr{A} \subseteq (2\mathbb{N})^n$, $\mathscr{B}_1 \subseteq \mathrm{conv}(\mathscr{A}) \cap (2\mathbb{N})^n$ *and* $\mathscr{B}_2 \subseteq \mathrm{conv}(\mathscr{A}) \cap (\mathbb{N}^n \backslash (2\mathbb{N})^n)$ *such that* $\mathscr{A} \cap \mathscr{B}_1 = \varnothing$, *the convex cone* $\mathrm{SONC}_{\mathscr{A},\mathscr{B}_1,\mathscr{B}_2}$ *has an* $(\mathbb{S}_+^2)^k$*-lift for some* $k \in \mathbb{N}$.

Proof: For every $\boldsymbol{\beta} \in \mathscr{B}_1 \cup \mathscr{B}_2$, let $\mathscr{C}(\boldsymbol{\beta})$ be as in (1). Then for every $\boldsymbol{\beta} \in \mathscr{B}_1 \cup \mathscr{B}_2$ and every $\mathscr{T} \in \mathscr{C}(\boldsymbol{\beta})$, let $M_{\mathscr{T}\boldsymbol{\beta}}$ be the $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$ provided by Lemma 8. Let $M = \cup_{\boldsymbol{\beta} \in \mathscr{B}_1 \cup \mathscr{B}_2} \cup_{\mathscr{T} \in \mathscr{C}(\boldsymbol{\beta})} M_{\mathscr{T}\boldsymbol{\beta}}$. For each $\boldsymbol{u}_i \in M \backslash \mathscr{A}$, let us write $\boldsymbol{u}_i = \frac{1}{2}(\boldsymbol{v}_i + \boldsymbol{w}_i)$. Let $\tilde{\mathscr{A}} = \{\boldsymbol{\alpha} \in \mathscr{A} \mid \boldsymbol{\alpha} \notin \cup_{\boldsymbol{\beta} \in \mathscr{B}_1 \cup \mathscr{B}_2} \cup_{\mathscr{T} \in \mathscr{C}(\boldsymbol{\beta})} \mathscr{T}\}$ and $k = \# M \backslash \mathscr{A} + \# \tilde{\mathscr{A}}$.

Then by Theorem 12, a polynomial f is in $\mathrm{SONC}_{\mathscr{A},\mathscr{B}_1,\mathscr{B}_2}$ if and only if f can be written as $f = \sum_{\boldsymbol{u}_i \in M \backslash \mathscr{A}} (a_i \mathbf{x}^{\frac{1}{2}\boldsymbol{v}_i} - b_i \mathbf{x}^{\frac{1}{2}\boldsymbol{w}_i})^2 + \sum_{\alpha \in \tilde{\mathscr{A}}} c_\alpha \mathbf{x}^\alpha$, $a_i, b_i \in \mathbb{R}$, which is equivalent to the existence of matrices $Q_1, \cdots, Q_k \in \mathbb{S}_+^2$ such that

$$f = \sum_{\boldsymbol{u}_i \in M \backslash \mathscr{A}} \left[\frac{1}{2}\boldsymbol{v}_i, \frac{1}{2}\boldsymbol{w}_i\right] Q_i \left[\frac{1}{2}\boldsymbol{v}_i, \frac{1}{2}\boldsymbol{w}_i\right]^T + \sum_{\alpha \in \tilde{\mathscr{A}}, Q_i} \left[\frac{1}{2}\boldsymbol{\alpha}, \mathbf{0}\right] Q_i \left[\frac{1}{2}\boldsymbol{\alpha}, \mathbf{0}\right]^T. \tag{3}$$

Let $\pi : (\mathbb{S}_+^2)^k \to \mathrm{SONC}_{\mathscr{A},\mathscr{B}_1,\mathscr{B}_2}$ be the linear map that maps an element $(Q_1, \cdots, Q_k)$ in $(\mathbb{S}_+^2)^k$ to the coefficient vector of f which is in $\mathrm{SONC}_{\mathscr{A},\mathscr{B}_1,\mathscr{B}_2}$ via the equality (3). Hence π yields an $(\mathbb{S}_+^2)^k$-lift for $\mathrm{SONC}_{\mathscr{A},\mathscr{B}_1,\mathscr{B}_2}$. □

Comparison with Averkov's construction on the size of SOC lifts. Given a SONC cone, the SOC lift due to Averkov (2019) is built on SOC lifts for hypographs of weighted geometric means. In his paper, Averkov appealed to the construction given in Ben-Tal and Nemirovski

(2001) for SOC lifts of weighted geometric means, which is of size $O(p)$ where p is the least common denominator of the weights (i.e., the barycentric coordinates $\{\lambda_i\}$). It is possible to produce such lifts of smaller size. Indeed, Sagnol (2013) presented a construction on matrix geometric means, which yields a SOC lift of size $2\lfloor \log_2(p) \rfloor + 1$ for bivariate weighted geometric means with p being the denominator of the weight; see also Fawzi and Saunderson (2017). Our construction of mediated sequences also equivalently yields a SOC lift for bivariate weighted geometric means. Even though we can only prove a quadratic bound in terms of $\log_2(p)$ for the size of this construction (Lemma 6), we empirically found that it actually depends linearly on $\log_2(p)$, which is approximately $1.29 \log_2(p) - 0.26$. The following table provides a comparison on the sizes of different constructions for (hypographs of) SOC lifts of bivariate weighted geometric means for $p = 10, 10^2, \ldots, 10^8$, where the average size of our construction is taken over all integers q such that the weight $t = q/p$, $0 < q < p$ and $\gcd(p, q) = 1$.

the denominator p	10	10^2	10^3	10^4	10^5	10^6	10^7	10^8
the size predicted by Conjecture 5	4	7	10	14	17	20	24	27
the average size of our construction	4.0	8.4	12.5	16.8	21.2	25.4	29.7	34.0
the size of Sagnol's construction	7	13	19	27	33	39	47	53

5. SONC optimization via SOCP

In this section, we tackle the following unconstrained polynomial optimization problem via SOCP, based on the representation of SONC cones derived in the previous section:

$$(\mathrm{P}): \quad \sup\{\xi : f(\mathbf{x}) - \xi \geq 0, \quad \mathbf{x} \in \mathbb{R}^n\}. \tag{4}$$

Let us denote by ξ^* the optimal value of (4). Replace the nonnegativity constraint in (4) by the following one to obtain a SONC relaxation with optimal value denoted by ξ_{sonc}:

$$(\mathrm{SONC}): \quad \sup\{\xi : f(\mathbf{x}) - \xi \in \mathrm{SONC}\}. \tag{5}$$

5.1. Conversion to PN-polynomials

Suppose $f = \sum_{\alpha \in \Lambda(f)} c_\alpha \mathbf{x}^\alpha - \sum_{\beta \in \Gamma(f)} d_\beta \mathbf{x}^\beta \in \mathbb{R}[\mathbf{x}]$. If $d_\beta > 0$ for all $\boldsymbol{\beta} \in \Gamma(f)$, then we call f a *PN-polynomial*. The prefix "PN" in PN-polynomial is short for "positive part plus negative part". The positive part is given by $\sum_{\alpha \in \Lambda(f)} c_\alpha \mathbf{x}^\alpha$ and the negative part is given by $-\sum_{\beta \in \Gamma(f)} d_\beta \mathbf{x}^\beta$. For a PN-polynomial $f(\mathbf{x})$, it is clear that $f(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in \mathbb{R}^n$ if and only if $f(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in \mathbb{R}^n_+$. Moreover, we have the following lemma.

Lemma 17. *Let $f(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$ be a PN-polynomial. Then for any positive integer k, $f(\mathbf{x}) \in$ SONC if and only if $f(\mathbf{x}^k) \in$ SONC.*

Proof: It comes from the fact that a polynomial $f(\mathbf{x})$ with exactly one negative term is a nonnegative circuit if and only if $f(\mathbf{x}^k)$ is a nonnegative circuit for any positive integer k. □

Theorem 18. *Let $f = \sum_{\alpha \in \Lambda(f)} c_\alpha \mathbf{x}^\alpha - \sum_{\beta \in \Gamma(f)} d_\beta \mathbf{x}^\beta \in \mathbb{R}[\mathbf{x}]$ be a PN-polynomial. Let $\mathscr{C}(\boldsymbol{\beta})$ be as in (1). For every $\boldsymbol{\beta} \in \Gamma(f)$ and every $\mathscr{T} \in \mathscr{C}(\boldsymbol{\beta})$, let $M_{\mathscr{T}\boldsymbol{\beta}}$ be a $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$. Let $M = \cup_{\boldsymbol{\beta} \in \Gamma(f)} \cup_{\mathscr{T} \in \mathscr{C}(\boldsymbol{\beta})} M_{\mathscr{T}\boldsymbol{\beta}}$ and $\tilde{\mathscr{A}} = \{\boldsymbol{\alpha} \in \Lambda(f) \mid \boldsymbol{\alpha} \notin \cup_{\boldsymbol{\beta} \in \Gamma(f)} \cup_{\mathscr{T} \in \mathscr{C}(\boldsymbol{\beta})} \mathscr{T}\}$. For each $\boldsymbol{u} \in M \backslash \Lambda(f)$, let $\boldsymbol{u} = \frac{1}{2}(\boldsymbol{v} + \boldsymbol{w})$. Then $f \in$ SONC if and only if f can be written as $f = \sum_{\boldsymbol{u} \in M \backslash \Lambda(f)} (a_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{v}} - b_{\boldsymbol{u}} \mathbf{x}^{\frac{1}{2}\boldsymbol{w}})^2 + \sum_{\alpha \in \tilde{\mathscr{A}}} c_\alpha \mathbf{x}^\alpha$, $a_{\boldsymbol{u}}, b_{\boldsymbol{u}} \in \mathbb{R}$.*

Proof: It follows easily from Lemma 17 and Theorem 3. □

The significant difference between Theorem 12 and Theorem 18 is that to represent a SONC PN-polynomial as a sum of binomial squares, we do not require the denominators of coordinates of points in $\mathscr{T}$-rational mediated sets to be odd. By virtue of this fact, for given trellis $\mathscr{T} = \{\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_m\}$ and lattice point $\boldsymbol{\beta} \in \text{conv}(\mathscr{T})^\circ$, we can then construct a $\mathscr{T}$-rational mediated set $M_{\mathscr{T}\boldsymbol{\beta}}$ containing $\boldsymbol{\beta}$ which is smaller than that the one from Lemma 8.

Lemma 19. *For a trellis $\mathscr{T}$ and a lattice point $\boldsymbol{\beta} \in \text{conv}(\mathscr{T})^\circ$, there is a $\mathscr{T}$-rational mediated set $M_{\mathscr{T}\boldsymbol{\beta}}$ containing $\boldsymbol{\beta}$.*

Proof: Suppose that $\boldsymbol{\beta} = \sum_{i=1}^m \frac{q_i}{p}\boldsymbol{\alpha}_i$, where $p = \sum_{i=1}^m q_i$, $p, q_i \in \mathbb{N}\backslash\{0\}$, $\gcd(p, q_1, \dots, q_m) = 1$. We can write

$$\boldsymbol{\beta} = \frac{q_1}{p}\boldsymbol{\alpha}_1 + \frac{p-q_1}{p}(\frac{q_2}{p-q_1}\boldsymbol{\alpha}_2 + \cdots + \frac{q_m}{p-q_1}\boldsymbol{\alpha}_m).$$

Let $\boldsymbol{\beta}_1 = \frac{q_2}{p-q_1}\boldsymbol{\alpha}_2 + \cdots + \frac{q_m}{p-q_1}\boldsymbol{\alpha}_m$. Then $\boldsymbol{\beta} = \frac{q_1}{p}\boldsymbol{\alpha}_1 + \frac{p-q_1}{p}\boldsymbol{\beta}_1$. Apply the same procedure for $\boldsymbol{\beta}_1$, and continue like this. Eventually we obtain a set of points $\{\boldsymbol{\beta}_i\}_{i=0}^{m-2}$ (set $\boldsymbol{\beta}_0 = \boldsymbol{\beta}$) such that $\boldsymbol{\beta}_i = \lambda_i\boldsymbol{\alpha}_{i+1} + \mu_i\boldsymbol{\beta}_{i+1}$, $i = 0, \dots, m-3$ and $\boldsymbol{\beta}_{m-2} = \lambda_{m-2}\boldsymbol{\alpha}_{m-1} + \mu_{m-2}\boldsymbol{\alpha}_m$, where $\lambda_i + \mu_i = 1, \lambda_i, \mu_i > 0$, $i = 0, \dots, m-2$. For $\boldsymbol{\beta}_i = \lambda_i\boldsymbol{\alpha}_{i+1} + \mu_i\boldsymbol{\beta}_{i+1}$ (resp. $\boldsymbol{\beta}_{m-2} = \lambda_{m-2}\boldsymbol{\alpha}_{m-1} + \mu_{m-2}\boldsymbol{\alpha}_m$), let M_i be the $\{\boldsymbol{\alpha}_{i+1}, \boldsymbol{\beta}_{i+1}\}$- (resp. $\{\boldsymbol{\alpha}_{m-1}, \boldsymbol{\alpha}_m\}$-) rational mediated set containing $\boldsymbol{\beta}_i$ obtained by Lemma 7, $i = 0, \dots, m-2$. Let $M_{\mathscr{T}\boldsymbol{\beta}} = \cup_{i=0}^{m-2} M_i$. Then clearly $M_{\mathscr{T}\boldsymbol{\beta}}$ is a $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$. □

The procedure based on Lemma 19 to obtain a $\mathscr{T}$-rational mediated sets is stated in Algorithm 3.

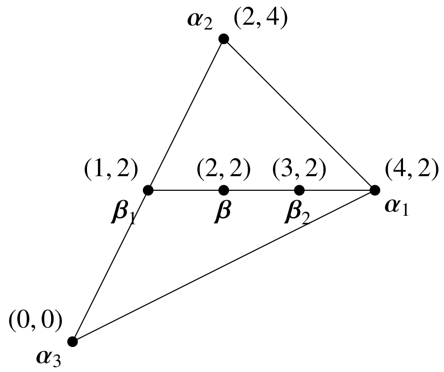
Algorithm 3 `MedSet`($\mathscr{T}$,$\boldsymbol{\beta}$)

Input: A trellis $\mathscr{T} = \{\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_m\}$ and a lattice point $\boldsymbol{\beta} \in \text{conv}(\mathscr{T})^\circ$

Output: A set of triples $\{(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i)\}_i$ with $\boldsymbol{u}_i = \frac{1}{2}(\boldsymbol{v}_i + \boldsymbol{w}_i)$ such that $\mathscr{T} \cup \{\boldsymbol{u}_i\}_i$ is a $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$

1: Let $\boldsymbol{\beta} = \sum_{i=1}^m \frac{q_i}{p}\boldsymbol{\alpha}_i$, where $p = \sum_{i=1}^m q_i$, $p, q_i \in \mathbb{N}\backslash\{0\}$, $\gcd(p, q_1, \dots, q_m) = 1$;
2: $k := 1, \boldsymbol{\beta}_0 := \boldsymbol{\beta}$;
3: **while** $k < m-1$ **do**
4: $\quad\boldsymbol{\beta}_k := \frac{q_{k+1}}{p-(q_1+\cdots+q_k)}\boldsymbol{\alpha}_{k+1} + \cdots + \frac{q_m}{p-(q_1+\cdots+q_k)}\boldsymbol{\alpha}_m$;
5: $\quad M_{k-1} :=$ `LMedSeq`($\boldsymbol{\alpha}_k, \boldsymbol{\beta}_k, \boldsymbol{\beta}_{k-1}$);
6: **end while**
7: $M_{m-2} :=$ `LMedSeq`($\boldsymbol{\alpha}_{m-1}, \boldsymbol{\alpha}_m, \boldsymbol{\beta}_{m-2}$);
8: $M := \cup_{i=0}^{m-2} M_i$;
9: **return** M;

Example 20. *Let $f = x^4y^2 + x^2y^4 + 1 - 3x^2y^2$ be the Motzkin polynomial and $\mathscr{T} = \{\boldsymbol{\alpha}_1 = (4,2), \boldsymbol{\alpha}_2 = (2,4), \boldsymbol{\alpha}_3 = (0,0)\}$, $\boldsymbol{\beta} = (2,2)$. Then $\boldsymbol{\beta} = \frac{1}{3}\boldsymbol{\alpha}_1 + \frac{1}{3}\boldsymbol{\alpha}_2 + \frac{1}{3}\boldsymbol{\alpha}_3 = \frac{1}{3}\boldsymbol{\alpha}_1 + \frac{2}{3}(\frac{1}{2}\boldsymbol{\alpha}_2 + \frac{1}{2}\boldsymbol{\alpha}_3)$. Let $\boldsymbol{\beta}_1 = \frac{1}{2}\boldsymbol{\alpha}_2 + \frac{1}{2}\boldsymbol{\alpha}_3$ such that $\boldsymbol{\beta} = \frac{1}{3}\boldsymbol{\alpha}_1 + \frac{2}{3}\boldsymbol{\beta}_1$. Let $\boldsymbol{\beta}_2 = \frac{2}{3}\boldsymbol{\alpha}_1 + \frac{1}{3}\boldsymbol{\beta}_1$. Then it is easy to check that $M = \{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha}_3, \boldsymbol{\beta}, \boldsymbol{\beta}_1, \boldsymbol{\beta}_2\}$ is a $\mathscr{T}$-rational mediated set containing $\boldsymbol{\beta}$.*



By a simple computation, we have $f = (1-xy^2)^2 + 2(x^{\frac{1}{2}}y - x^{\frac{3}{2}}y)^2 + (xy - x^2y)^2$. *Here we represent* f *as a sum of three binomial squares with rational exponents.*

We associate to a polynomial $f = \sum_{\boldsymbol{\alpha}\in\Lambda(f)} c_{\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} - \sum_{\boldsymbol{\beta}\in\Gamma(f)} d_{\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}}$, the PN-polynomial $\tilde{f} = \sum_{\boldsymbol{\alpha}\in\Lambda(f)} c_{\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} - \sum_{\boldsymbol{\beta}\in\Gamma(f)} |d_{\boldsymbol{\beta}}|\mathbf{x}^{\boldsymbol{\beta}}$.

Lemma 21. *Suppose* $f = \sum_{\boldsymbol{\alpha}\in\Lambda(f)} c_{\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} - \sum_{\boldsymbol{\beta}\in\Gamma(f)} d_{\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}} \in \mathbb{R}[\mathbf{x}]$. *If* $\tilde{f}$ *is nonnegative, then* f *is nonnegative. Moreover,* $\tilde{f} \in$ SONC *if and only if* $f \in$ SONC.

Proof: For any $\mathbf{x} \in \mathbb{R}^n$, we have

$$\begin{aligned} f(\mathbf{x}) &= \sum_{\boldsymbol{\alpha}\in\Lambda(f)} c_{\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} - \sum_{\boldsymbol{\beta}\in\Gamma(f)} d_{\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}} \\ &\geq \sum_{\boldsymbol{\alpha}\in\Lambda(f)} c_{\boldsymbol{\alpha}}|\mathbf{x}|^{\boldsymbol{\alpha}} - \sum_{\boldsymbol{\beta}\in\Gamma(f)} |d_{\boldsymbol{\beta}}||\mathbf{x}|^{\boldsymbol{\beta}} = \tilde{f}(|\mathbf{x}|), \end{aligned}$$

where $|\mathbf{x}| = (|x_1|, \dots, |x_n|)$. It follows that the nonnegativity of $\tilde{f}$ implies the nonnegativity of f.

For every $\boldsymbol{\beta} \in \Gamma(f)$, let $\mathscr{C}(\boldsymbol{\beta})$ be as in (1). Let $\mathscr{B} = \{\boldsymbol{\beta} \in \Gamma(f) \mid \boldsymbol{\beta} \notin (2\mathbb{N})^n \text{ and } d_{\boldsymbol{\beta}} < 0\}$ and $\tilde{\mathscr{A}} = \{\boldsymbol{\alpha} \in \Lambda(f) \mid \boldsymbol{\alpha} \notin \cup_{\boldsymbol{\beta}\in\Gamma(f)} \cup_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} \mathscr{T}\}$. Assume $\tilde{f} \in$ SONC. Then we can write

$$\begin{aligned} \tilde{f} = &\sum_{\boldsymbol{\beta}\in\Gamma(f)\backslash\mathscr{B}} \sum_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} (\sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\mathscr{T}\boldsymbol{\beta}\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} - d_{\mathscr{T}\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}}) \\ &+ \sum_{\boldsymbol{\beta}\in\mathscr{B}} \sum_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} (\sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\mathscr{T}\boldsymbol{\beta}\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} - \tilde{d}_{\mathscr{T}\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}}) + \sum_{\boldsymbol{\alpha}\in\tilde{\mathscr{A}}} c_{\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} \end{aligned}$$

such that each $\sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\mathscr{T}\boldsymbol{\beta}\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} - d_{\mathscr{T}\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}}$ and each $\sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\mathscr{T}\boldsymbol{\beta}\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} - \tilde{d}_{\mathscr{T}\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}}$ are nonnegative circuits. Note that $\sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\mathscr{T}\boldsymbol{\beta}\boldsymbol{\alpha}}\ \mathbf{x}^{\boldsymbol{\alpha}} + \tilde{d}_{\mathscr{T}\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}}$ is also a nonnegative circuit and $\sum_{\mathscr{T}\in\mathscr{T}(\boldsymbol{\beta})} \tilde{d}_{\mathscr{T}\boldsymbol{\beta}} = |d_{\boldsymbol{\beta}}| = -d_{\boldsymbol{\beta}}$ for any $\boldsymbol{\beta} \in \mathscr{B}$. Hence,

$$\begin{aligned} f = &\sum_{\boldsymbol{\beta}\in\Gamma(f)\backslash\mathscr{B}} \sum_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} (\sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\mathscr{T}\boldsymbol{\beta}\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} - d_{\mathscr{T}\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}}) \\ &+ \sum_{\boldsymbol{\beta}\in\mathscr{B}} \sum_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} (\sum_{\boldsymbol{\alpha}\in\mathscr{T}} c_{\mathscr{T}\boldsymbol{\beta}\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} + \tilde{d}_{\mathscr{T}\boldsymbol{\beta}}\mathbf{x}^{\boldsymbol{\beta}}) + \sum_{\boldsymbol{\alpha}\in\tilde{\mathscr{A}}} c_{\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\alpha}} \in \text{SONC}. \end{aligned}$$

The inverse follows similarly. □

Hence by Lemma 21, if we replace the polynomial f in (5) by its associated PN-polynomial $\tilde{f}$, then this does not affect the optimal value of (5):

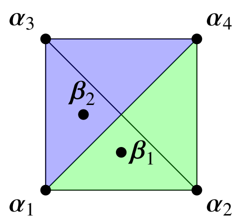
$$\text{(SONC-PN)}: \quad \begin{cases} \sup & \xi \\ \text{s.t.} & \tilde{f}(\mathbf{x}) - \xi \in \text{SONC}\,. \end{cases} \tag{6}$$

Remark 22. *Lemma 21 tells us that the SONC formulation for the polynomial optimization problem (5) always provides the optimal value w.r.t. the corresponding PN-polynomial. See also Remark 24.*

5.2. Compute a simplex cover

Given a polynomial $f = \sum_{\alpha\in\Lambda(f)} c_\alpha \mathbf{x}^\alpha - \sum_{\beta\in\Gamma(f)} d_\beta \mathbf{x}^\beta \in \mathbb{R}[\mathbf{x}]$, in order to obtain a SONC decomposition of f, we use all simplices covering $\boldsymbol{\beta}$ for each $\boldsymbol{\beta} \in \Gamma(f)$ in Theorem 12. In practice, we do not need that many simplices, as illustrated by the following example.

Example 23. *Let $f = 50x^4y^4 + x^4 + 3y^4 + 800 - 100xy^2 - 100x^2y$. Let $\boldsymbol{\alpha}_1 = (0,0), \boldsymbol{\alpha}_2 = (4,0), \boldsymbol{\alpha}_3 = (0,4), \boldsymbol{\alpha}_4 = (4,4)$ and $\boldsymbol{\beta}_1 = (2,1), \boldsymbol{\beta}_2 = (1,2)$. There are two simplices which cover $\boldsymbol{\beta}_1$: one with vertices $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\, \boldsymbol{\alpha}_3\}$, denoted by Δ_1, and one with vertices $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2\, \boldsymbol{\alpha}_4\}$, denoted by Δ_2. There are two simplices which cover $\boldsymbol{\beta}_2$: Δ_1 and one with vertices $\{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_3\, \boldsymbol{\alpha}_4\}$, denoted by Δ_3. One can check that f admits a SONC decomposition $f = g_1 + g_2$, where $g_1 = 20x^4y^4 + x^4 + 400 - 100x^2y$, supported on Δ_2, and $g_2 = 30x^4y^4 + 3y^4 + 400 - 100xy^2$, supported on Δ_3, are both nonnegative circuit polynomials. Hence the simplex Δ_1 is not needed in this SONC decomposition of f.*



A recent study by Papp (2019) proposes a systematic method to compute an optimal simplex cover. It would be worth trying to combine this framework with our SOC characterization for SONC cones to achieve a more accurate algorithm. Here we rely on a heuristics to compute a set of simplices with vertices coming from $\Lambda(f)$ and that covers $\Gamma(f)$. For $\boldsymbol{\beta} \in \Gamma(f)$ and $\boldsymbol{\alpha}_0 \in \Lambda(f)$, define an auxiliary linear program:

$$\begin{aligned} \text{SimSel}(\boldsymbol{\beta}, \Lambda(f), \boldsymbol{\alpha}_0) := \arg\max \quad & \lambda_{\alpha_0} \\ \text{s.t.}\ \{ & \sum_{\alpha\in\Lambda(f)} \lambda_\alpha \cdot \boldsymbol{\alpha} = \boldsymbol{\beta}, \sum_{\alpha\in\Lambda(f)} \lambda_\alpha = 1, \lambda_\alpha \geq 0, \forall \boldsymbol{\alpha} \in \Lambda(f)\}. \end{aligned}$$

Following Seidler and de Wolff (2018), we can ensure that the output of $\text{SimSel}(\boldsymbol{\beta}, \Lambda(f), \boldsymbol{\alpha}_0)$[1] corresponds to a trellis which contains $\boldsymbol{\alpha}_0$ and covers $\boldsymbol{\beta}$.

Suppose $\tilde{f} = \sum_{\alpha\in\Lambda(f)} c_\alpha \mathbf{x}^\alpha - \sum_{\beta\in\Gamma(f)} d_\beta \mathbf{x}^\beta \in \mathbb{R}[\mathbf{x}]$ and assume that $\tilde{\mathscr{A}} = \{\boldsymbol{\alpha} \in \Lambda(f) \mid \boldsymbol{\alpha} \notin \cup_{\beta\in\Gamma(f)} \cup_{\mathscr{T}\in\mathscr{C}(\boldsymbol{\beta})} \mathscr{T}\} = \varnothing$. The procedure `SimplexCover` stated in Algorithm 4 allows one to

[1] If $\boldsymbol{\beta}$ lies on a face of $\text{New}(f)$, then we assume $\boldsymbol{\alpha}_0$ lies on the same face.

Algorithm 4 `SimplexCover(`$\Lambda(f), \Gamma(f)$`)`

Input: $\Lambda(f), \Gamma(f)$
Output: $\{(\mathscr{T}_k, \boldsymbol{\beta}_k)\}_k$: a set of circuits such that $\cup_k \mathscr{T}_k = \Lambda(f)$ and $\{\boldsymbol{\beta}_k\}_k = \Gamma(f)$

$U := \Lambda(f), V := \Gamma(f), k := 0;$
while $U \neq \emptyset$ and $V \neq \emptyset$ **do**
    $k := k + 1;$
    Choose $\boldsymbol{\alpha}_0 \in U$ and $\boldsymbol{\beta}_k \in V;$
    $\boldsymbol{\lambda} := \mathrm{SimSel}(\boldsymbol{\beta}_k, \Lambda(f), \boldsymbol{\alpha}_0);$
    $\mathscr{T}_k := \{\boldsymbol{\alpha} \in \Lambda(f) \mid \lambda_{\boldsymbol{\alpha}} > 0\};$
    $U := U \backslash \mathscr{T}_k, V := V \backslash \{\boldsymbol{\beta}_k\};$
end while
if $V \neq \emptyset$ **then**
    while $V \neq \emptyset$ **do**
        if $U = \emptyset$ **then**
            $U := \Lambda(f);$
        end if
        $k := k + 1;$
        Choose $\mathscr{T}_0 \in U$ and $\boldsymbol{\beta}_k \in V;$
        $\boldsymbol{\lambda} := \mathrm{SimSel}(\boldsymbol{\beta}_k, \Lambda(f), \boldsymbol{\alpha}_0);$
        $\mathscr{T}_k := \{\boldsymbol{\alpha} \in \Lambda(f) \mid \lambda_{\boldsymbol{\alpha}} > 0\};$
        $U := U \backslash \mathscr{T}_k, V := V \backslash \{\boldsymbol{\beta}_k\};$
    end while
else
    while $U \neq \emptyset$ **do**
        if $V = \emptyset$ **then**
            $V := \Gamma(f);$
        end if
        $k := k + 1;$
        Choose $\boldsymbol{\alpha}_0 \in U$ and $\boldsymbol{\beta}_k \in V;$
        $\boldsymbol{\lambda} := \mathrm{SimSel}(\boldsymbol{\beta}_k, \Lambda(f), \boldsymbol{\alpha}_0);$
        $\mathscr{T}_k := \{\boldsymbol{\alpha} \in \Lambda(f) \mid \lambda_{\boldsymbol{\alpha}} > 0\};$
        $U := U \backslash \mathscr{T}_k, V := V \backslash \{\boldsymbol{\beta}_k\};$
    end while
end if
return $\{(\mathscr{T}_k, \boldsymbol{\beta}_k)\}_k$

compute a simplex cover $\{(\mathscr{T}_k, \boldsymbol{\beta}_k)\}_{k=1}^{l}$ for $\tilde{f}$. For each k, let M_k be a $\mathscr{T}_k$-rational mediated set containing $\boldsymbol{\beta}_k$ and $s_k = \# M_k \backslash \mathscr{T}_k$. For each $\boldsymbol{u}_i^k \in M_k \backslash \mathscr{T}_k$, let us write $\boldsymbol{u}_i^k = \frac{1}{2}(\boldsymbol{v}_i^k + \boldsymbol{w}_i^k)$. Let $\mathbf{K}$ be the 3-dimensional rotated SOC, i.e.,

$$\mathbf{K} := \{(a, b, c) \in \mathbb{R}^3 \mid 2ab \geq c^2, a \geq 0, b \geq 0\}. \tag{7}$$

Then we can relax (SONC-PN) to a SOCP problem as follows:

$$\text{(SONC-SOCP)}: \quad \begin{cases} \sup & \xi \\ \text{s.t.} & \tilde{f}(\mathbf{x}) - \xi = \sum_{k=1}^{l} \sum_{i=1}^{s_k} (2a_i^k \mathbf{x}^{\boldsymbol{v}_i^k} + b_i^k \mathbf{x}^{\boldsymbol{w}_i^k} - 2c_i^k \mathbf{x}^{\boldsymbol{u}_i^k}), \\ & (a_i^k, b_i^k, c_i^k) \in \mathbf{K}, \quad \forall i, k. \end{cases} \tag{8}$$

Let us denote by ξ_{socp} the optimal value of (8). Then, we have $\xi_{socp} \leq \xi_{sonc} \leq \xi^*$.

Remark 24. *The quality of obtained SONC lower bounds depends on two successive steps: the relaxation to the corresponding PN-polynomial (from ξ^* to ξ_{sonc}) and the relaxation to a specific simplex cover (from ξ_{sonc} to ξ_{socp}). The loss of bound-quality at the second step can be improved by computing a more optimal simplex cover. Nevertheless, it may happen that the loss of bound-quality at the first step is already big, as shown in Example 25, which indicates that the gap between nonnegative polynomials and SONC PN-polynomials (see figure 1) may greatly affect the quality of SONC lower bounds.*

Example 25. *Let $f = 1 + x_1^4 + x_2^4 - x_1 x_2^2 - x_1^2 x_2 + 5x_1 x_2$. Since $\Lambda(f)$ forms a trellis, the simplex cover for f is unique. One obtains $\xi_{socp} = \xi_{sonc} \approx -6.916501$ while $\xi^* \approx -2.203372$. Hence the relative optimality gap is near 214%.*

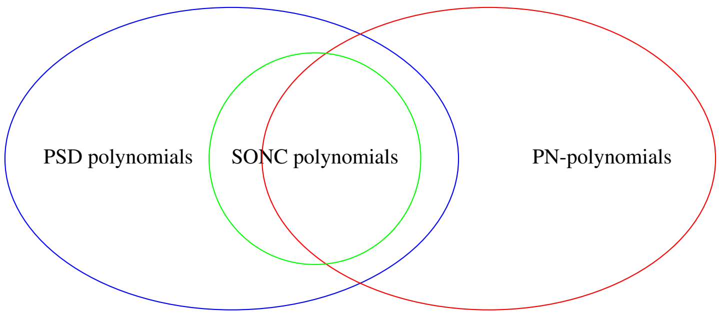


Figure 1: Relationship of different classes of polynomials

6. Exact SOBS decompositions of SONC polynomials

6.1. *The procedure* `ExactSOBS`

Here, we present a procedure, called `ExactSOBS`, to compute exact rational SOBS decompositions (with rational exponents) of PN-polynomials lying in the interior of the SONC cone (if f itself is not a PN-polynomial, then by Lemma 21, an exact rational SOBS decompositions of its associated PN-polynomial $\tilde{f}$ also provides a nonnegativity certificate for f). The procedure, stated in Algorithm 5, is a rounding-projection procedure, in the spirit of the one by Peyrl and Parrilo (2008) to obtain exact SOS decompositions for polynomials in the interior of the SOS

Algorithm 5 $\texttt{ExactSOBS}(f, \hat{\delta}, \tilde{\delta})$

Input: A SONC PN-polynomial $f = \sum_{\alpha \in \mathscr{A}} f_\alpha \mathbf{x}^\alpha \in \mathbb{Q}[\mathbf{x}]$ with $\tilde{\mathscr{A}} = \varnothing$, a precision parameter $\tilde{\delta}$ for the SOCP solver, a rounding precision $\hat{\delta}$

Output: An exact SOBS decomposition of f

Determine the sets $\Lambda(f), \Gamma(f)$ such that $f = \sum_{\alpha \in \Lambda(f)} c_\alpha \mathbf{x}^\alpha - \sum_{\beta \in \Gamma(f)} d_\beta \mathbf{x}^\beta$;

$\{(\mathscr{T}_k, \boldsymbol{\beta}_k)\}_{k=1}^{l} := \texttt{SimplexCover}(\Lambda(f), \Gamma(f))$;

for $k = 1 : l$ **do**

  $\{(\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k)\}_{i=1}^{s_k} := \texttt{MedSet}(\mathscr{T}_k, \boldsymbol{\beta}_k)$;

end for

Obtain $\{(\tilde{a}_i^k, \tilde{b}_i^k, \tilde{c}_i^k)\}_{i,k}$ by solving (SONC-SOCP) with $\xi = 0$ at precision $\tilde{\delta}$;

$\{(\hat{a}_i^k, \hat{b}_i^k, \hat{c}_i^k)\}_{i,k} := \texttt{round}(\{(\tilde{a}_i^k, \tilde{b}_i^k, \tilde{c}_i^k)\}_{i,k}, \hat{\delta})$; {rounding step}

$B := \cup_{k=1}^{l} \cup_{i=1}^{s_k} \{\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k\}$;

for $\boldsymbol{\gamma} \in B$ **do**

  $\eta(\boldsymbol{\gamma}) := \#\{(i,k) \mid \boldsymbol{\gamma} \in \{\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k\}, i = 1, \ldots, s_k, k = 1, \ldots, l\}$;

end for

for $k = 1 : l, i = 1 : s_k$ **do**

  $a_i^k := \hat{a}_i^k - \frac{1}{2\eta(\boldsymbol{v}_i^k)}\Big(\sum_{\boldsymbol{u}_p^q = \boldsymbol{v}_i^k} 2\hat{a}_p^q + \sum_{\boldsymbol{v}_p^q = \boldsymbol{v}_i^k} \hat{b}_p^q - \sum_{\boldsymbol{w}_p^q = \boldsymbol{v}_i^k} 2\hat{c}_p^q - f_{\boldsymbol{v}_i^k}\Big)$; {projection step}

  $b_i^k := \hat{b}_i^k - \frac{1}{\eta(\boldsymbol{w}_i^k)}\Big(\sum_{\boldsymbol{u}_p^q = \boldsymbol{w}_i^k} 2\hat{a}_p^q + \sum_{\boldsymbol{v}_p^q = \boldsymbol{w}_i^k} \hat{b}_p^q - \sum_{\boldsymbol{w}_p^q = \boldsymbol{w}_i^k} 2\hat{c}_p^q - f_{\boldsymbol{w}_i^k}\Big)$; {projection step}

  $c_i^k := \hat{c}_i^k + \frac{1}{2\eta(\boldsymbol{u}_i^k)}\Big(\sum_{\boldsymbol{u}_p^q = \boldsymbol{u}_i^k} 2\hat{a}_p^q + \sum_{\boldsymbol{v}_p^q = \boldsymbol{u}_i^k} \hat{b}_p^q - \sum_{\boldsymbol{w}_p^q = \boldsymbol{u}_i^k} 2\hat{c}_p^q - f_{\boldsymbol{u}_i^k}\Big)$; {projection step}

end for

for $k = 1 : l$ **do**

  $f_k := \sum_{i=1}^{s_k} (2a_i^k \mathbf{x}^{\boldsymbol{v}_i^k} + b_i^k \mathbf{x}^{\boldsymbol{w}_i^k} - 2c_i^k \mathbf{x}^{\boldsymbol{u}_i^k})$;

end for

return $\sum_{k=1}^{l} f_k$;

cone and by Magron et al. (2019b) to obtain exact SAGE decompositions of polynomials in the interior of the SAGE cone.

After computing a simplex cover in Line 2 with Algorithm 4 and the corresponding rational mediated sets in Line 4 with Algorithm 3, the procedure calls a SOCP solver in Line 6 to compute a $\tilde{\delta}$-approximation $(\tilde{a}_i^k, \tilde{b}_i^k, \tilde{c}_i^k)_{i,k}$ of the coefficients involved in the SOBS decomposition of f from (8). This approximation is then rounded to rational points in Line 7 with a prescribed accuracy of $\hat{\delta}$. The three projection steps from Line 13 to Line 15 ensure that the rational points $(a_i^k, b_i^k, c_i^k)_{i,k}$ satisfy exactly the equality constraints from (8), namely that for each $k = 1, \ldots, \ell$ and each $i = 1, \ldots, s_k$, one has

$$f_{\boldsymbol{\gamma}} = \left(\sum_{\substack{p,q \\ \boldsymbol{v}_p^q = \boldsymbol{\gamma}}} 2a_p^q + \sum_{\substack{p,q \\ \boldsymbol{w}_p^q = \boldsymbol{\gamma}}} b_p^q - \sum_{\substack{p,q \\ \boldsymbol{u}_p^q = \boldsymbol{\gamma}}} 2c_p^q \right), \tag{9}$$

for each $\boldsymbol{\gamma} \in \{\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k\}_{i,k}$. As proved later on, if both numerical SOCP solver accuracy $\tilde{\delta}$ and rounding accuracy $\hat{\delta}$ are high enough, then each triple (a_i^k, b_i^k, c_i^k) belongs to the second-order cone $\mathbf{K} = \{(a, b, c) : a, b \geq 0, 2ab \geq c^2\}$.

6.2. Arithmetic complexity

The *bit size* of $i \in \mathbb{Z}$ is denoted by $\tau(i) := \lfloor \log_2(|i|) \rfloor + 1$ with $\tau(0) := 1$. Given $i \in \mathbb{Z}$ and $j \in \mathbb{Z} \backslash \{0\}$ with $\gcd(i, j) = 1$, we define $\tau(i/j) := \max\{\tau(i), \tau(j)\}$. For two mappings $g, h : \mathbb{N}^m \to \mathbb{R}$, we use the notation $g(v) \in O(h(v))$ to state the existence of $i \in \mathbb{N}$ such that $g(v) \leq ih(v)$, for all $v \in \mathbb{N}^m$. The expression "$g(v) \in \widetilde{O}(h(v))$" means that there exists $c \in \mathbb{N}$ such that $g(v) \leq h(v) \log_2(h(v))^c$, for all $v \in \mathbb{N}^m$.

Lemma 26. *Let $f \in \mathbb{R}[\mathbf{x}]$ be a polynomial of degree d and let D be the maximal value of the denominators involved in the triples $\{(\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k)\}_{i,k}$ in (8). Then $D \leq (1 + nd^2)^{n+1}$.*

Proof: Consider any circuit $(\mathscr{T}, \boldsymbol{\beta})$ such that $\mathscr{T} \subseteq \Lambda(f)$ and $\boldsymbol{\beta} \in \Gamma(f)$. Without loss of generality, assume that $\mathscr{T} = \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_{n+1}\}$. We write $\boldsymbol{\beta} = \sum_{i=1}^{n+1} \frac{q_i}{p} \boldsymbol{\alpha}_i$, where $p = \sum_{i=1}^{n+1} q_i$, $p, q_i \in \mathbb{N} \backslash \{0\}$, $\gcd(p, q_1, \ldots, q_{n+1}) = 1$.

First, one can easily see that the denominators involved in the output $\{(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i)\}_i$ of the algorithm $\texttt{LMedSet}(\boldsymbol{\alpha}', \boldsymbol{\alpha}'', \boldsymbol{\beta}')$ do not exceed the common denominator of $\boldsymbol{\alpha}', \boldsymbol{\alpha}'', \boldsymbol{\beta}'$. Thus it suffices to consider the denominators of $\boldsymbol{\beta}_i$'s appearing in the algorithm $\texttt{MedSet}$, which are clearly bounded from above by p. Note that $\boldsymbol{\lambda} = (\frac{q_1}{p}, \ldots, \frac{q_{n+1}}{p})$ is the unique solution to the system of linear equations:

$$\begin{bmatrix} 1 & \cdots & 1 \\ \boldsymbol{\alpha}_1 & \cdots & \boldsymbol{\alpha}_{n+1} \end{bmatrix} \boldsymbol{\lambda} = \begin{bmatrix} 1 \\ \boldsymbol{\beta} \end{bmatrix}.$$

Hence $p \leq \det(\begin{bmatrix} 1 & \cdots & 1 \\ \boldsymbol{\alpha}_1 & \cdots & \boldsymbol{\alpha}_{n+1} \end{bmatrix}) \leq (1 + nd^2)^{\frac{n+1}{2}}$. The latter inequality follows from Hadamard's inequality. As a result, $D \leq p^2 \leq (1 + nd^2)^{n+1}$. $\square$

For later purpose, we recall the result which bounds the roots of univariate polynomials with integer coefficients:

Lemma 27. *(Mignotte, 1992, Theorem 4.2 (ii)) Let $f \in \mathbb{Z}[x]$ of degree d, with coefficient bit size bounded from above by τ. If $f(\tilde{x}) = 0$ and $\tilde{x} \neq 0$, then $\frac{1}{2^\tau + 1} \leq |\tilde{x}| \leq 2^\tau + 1$.*

Lemma 28. *Let $f = \sum_{\alpha \in \mathscr{A}} f_\alpha \mathbf{x}^\alpha \in \mathbb{Q}[\mathbf{x}]$ be a SONC PN-polynomial of degree $d \geq 2$ with $\tau = \tau(f)$. Assume that $\mathscr{A} = \varnothing$ and f lies in the interior of the SONC cone. Let D be the maximal value of the denominators involved in the triples $\{(\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k)\}_{i,k}$ in (8), bounded as in Lemma 26. Then, there exists $N \in \mathbb{N}$ such that for $\varepsilon := 2^{-N}$, $f - \varepsilon \sum_{k=1}^{l} \sum_{i=1}^{s_k} (\mathbf{x}^{\boldsymbol{u}_i^k} + \mathbf{x}^{\boldsymbol{v}_i^k} + \mathbf{x}^{\boldsymbol{w}_i^k})$ has an SOBS decomposition (with rational exponents), with $N \leq \tau(\varepsilon) \in O(\tau \cdot (4dD + 6)^{3n+3})$, thus $N \leq \tau(\varepsilon) \in O(\tau \cdot (5d)^{3n+3}(1 + nd^2)^{3(n+1)^2})$.*

Proof: Let us define $g(\mathbf{x}) := f(\mathbf{x}^D)$. Note that the degree of g is at most Dd. In addition, for each each $\boldsymbol{\gamma} \in \{\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k\}_{i,k}$ involved in the SOBS decomposition of f, one has $\mathbf{x}^{D\boldsymbol{\gamma}} \in \mathbb{R}[\mathbf{x}]$. Thus g is a sum of binomial squares and one has $\|D\boldsymbol{\gamma}\|_1 \leq Dd$. Since the polynomial f is in the interior of the SONC cone, then the polynomial g is also in the interior of the SONC cone. Then by definition, there exists $N \in \mathbb{N}$ such that for $\varepsilon := 2^{-N}$, one has $g - \varepsilon \sum_{k=1}^{l} \sum_{i=1}^{s_k} (\mathbf{x}^{D\boldsymbol{u}_i^k} + \mathbf{x}^{D\boldsymbol{v}_i^k} + \mathbf{x}^{D\boldsymbol{w}_i^k}) \in \text{SONC}$. Let us define the polynomial $h(\mathbf{x}, z) := f(\mathbf{x}^D) - z \sum_{k=1}^{l} \sum_{i=1}^{s_k} (\mathbf{x}^{D\boldsymbol{u}_i^k} + \mathbf{x}^{D\boldsymbol{v}_i^k} + \mathbf{x}^{D\boldsymbol{w}_i^k})$, and consider the algebraic set V defined by:

$$V := \left\{ (\mathbf{x}, z) \in \mathbb{R}^{n+1} : h(\mathbf{x}, z) = \frac{\partial h}{\partial x_1} = \cdots = \frac{\partial h}{\partial x_n} = 0 \right\} .$$

The degree of h is also Dd. Using Proposition A.1 from Magron and Safey El Din (2018b), there exists a polynomial in $\mathbb{Z}[z]$ of degree less than $(Dd + 1)^{n+1}$ with coefficients of bit size less than $\tau \cdot (4Dd + 6)^{3n+3}$ such that its set of real roots contains the projection of V on the z-axis. By Lemma 27, it is enough to take $N \leq \tau \cdot (4Dd + 6)^{3n+3}$. By Lemma 26, one has $D \leq (1 + nd^2)^{n+1}$. The desired complexity bound follows from the fact that $d \geq 2$ implies that $6 \leq d(1 + nd^2)^{n+1}$. By Theorem 18, the polynomial $h(\mathbf{x}, \varepsilon)$ admits an SOBS decomposition. Therefore $h(\mathbf{x}^{1/D}, \varepsilon)$ also admits an SOBS decomposition, the desired result. □

Lemma 29. *Let $f \in \mathbb{R}[\mathbf{x}]$ be as in Lemma 28 with t terms. Let L be the total number of triples $(\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k)$ appearing in (8), i.e., $L = \sum_{k=1}^{l} s_k$. Then $L < \frac{1}{8} tn((n+1)\log_2(1+nd^2)+3)^2 \in \tilde{O}(d^n n^3)$.*

Proof: First note that the number of circuits produced by the procedure $\texttt{SimplexCover}(\Lambda(f), \Gamma(f))$ is less than t. Consider any circuit $(\mathscr{T}, \boldsymbol{\beta})$ such that $\mathscr{T} \subseteq \Lambda(f)$ and $\boldsymbol{\beta} \in \Gamma(f)$. Without loss of generality, assume that $\mathscr{T} = \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_{n+1}\}$. We write $\boldsymbol{\beta} = \sum_{i=1}^{n+1} \frac{q_i}{p} \boldsymbol{\alpha}_i$, where $p = \sum_{i=1}^{n+1} q_i$, $p, q_i \in \mathbb{N} \backslash \{0\}$, $\gcd(p, q_1, \ldots, q_{n+1}) = 1$. Then by the proof of Lemma 26, $p \leq (1 + nd^2)^{\frac{n+1}{2}}$. For each triple $(\boldsymbol{\alpha}_k, \boldsymbol{\beta}_k, \boldsymbol{\beta}_{k-1})$, $1 \leq k \leq n-1$ or $(\boldsymbol{\alpha}_n, \boldsymbol{\alpha}_{n+1}, \boldsymbol{\beta}_{n-1})$ appearing in the algorithm $\texttt{MedSet}$, by Lemma 6 and the proof of Lemma 7, the number of triples $(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i)$ produced by the algorithm $\texttt{LMedSet}$ is less than $\frac{1}{2}(\log_2(p) + \frac{3}{2})^2$. Putting all above together, we have $L < tn \cdot \frac{1}{2}(\log_2((1 + nd^2)^{\frac{n+1}{2}}) + \frac{3}{2})^2 = \frac{1}{8} tn((n+1)\log_2(1 + nd^2) + 3)^2$. Note that the number of terms t is less than $2d^n$:

$$t = \binom{n+d}{n} = \frac{(n+d)\ldots(n+1)}{n!} = \left(1 + \frac{d}{n}\right)\left(1 + \frac{d}{n-1}\right)\ldots(1+d) \leq d^{n-1}(1+d) \leq 2d^n ,$$

yielding the final complexity result. □

Remark 30. *By Lemma 29, the number of second-order cones used in (SONC-SOCP) (8) scales linearly in the number of terms of the polynomial.*

Theorem 31. *Let f, D be as in Lemma 28. Let L be the total number of triples $(\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k)$ appearing in (8), i.e., $L = \sum_{k=1}^{l} s_k$. There exist $\hat{\delta}$, $\tilde{\delta}$ of bit size less than $O(\tau \cdot (4Dd+6)^{3n+3})$, such that* `ExactSOBS`$(f, \hat{\delta}, \tilde{\delta})$ *terminates and outputs a rational SOBS decomposition of f within $O((\tau(L) + \tau \cdot (4Dd+6)^{3n+3})\, L^{3.5})$ arithmetic operations, yielding a total number of*

$$\widetilde{O}\left(\tau \cdot 5^{3n+3} d^{6.5n+3} n^{10.5} (1+nd^2)^{3(n+1)^2}\right)$$

arithmetic operations.

Proof: Let ε be as in Lemma 28. Then, there exist sequences of triples $\{(\overline{a}_i^k, \overline{b}_i^k, \overline{c}_i^k)\}_{i,k}$ as in (8) such that

$$f - \varepsilon \sum_{k=1}^{l} \sum_{i=1}^{s_k} (\mathbf{x}^{\boldsymbol{u}_i^k} + \mathbf{x}^{\boldsymbol{v}_i^k} + \mathbf{x}^{\boldsymbol{w}_i^k}) = \sum_{k=1}^{l} \sum_{i=1}^{s_k} (2\overline{a}_i^k \mathbf{x}^{\boldsymbol{v}_i^k} + \overline{b}_i^k \mathbf{x}^{\boldsymbol{w}_i^k} - 2\overline{c}_i^k \mathbf{x}^{\boldsymbol{u}_i^k}), \quad (\overline{a}_i^k, \overline{b}_i^k, \overline{c}_i^k) \in \mathbf{K}, \forall i, k.$$

(1) Let us define $\delta := \min\{\frac{3\varepsilon^2}{4}, \frac{\varepsilon}{2}\}$, $\tilde{a}_i^k := \overline{a}_i^k + \frac{\varepsilon}{2}$, $\tilde{b}_i^k := \overline{b}_i^k + \varepsilon$ and $\tilde{c}_i^k := \overline{c}_i^k - \frac{\varepsilon}{2}$. Let us show that $\{(\tilde{a}_i^k, \tilde{b}_i^k, \tilde{c}_i^k)\}_{i,k}$ is a strictly feasible solution of (8) (with $\xi = 0$) such that $\tilde{a}_i^k \geq \delta$, $b_i^k \geq \delta$ and $2\tilde{a}_i^k \tilde{b}_i^k - (\tilde{c}_i^k)^2 \geq \delta$.
According to these definitions, one has

$$f = \sum_{k=1}^{l} \sum_{i=1}^{s_k} (2\tilde{a}_i^k \mathbf{x}^{\boldsymbol{v}_i^k} + \tilde{b}_i^k \mathbf{x}^{\boldsymbol{w}_i^k} - 2\tilde{c}_i^k \mathbf{x}^{\boldsymbol{u}_i^k}),$$

and

$$\sum_{\substack{p,q \\ \boldsymbol{v}_p^q = \boldsymbol{\gamma}}} 2\tilde{a}_p^q + \sum_{\substack{p,q \\ \boldsymbol{w}_p^q = \boldsymbol{\gamma}}} \tilde{b}_p^q - \sum_{\substack{p,q \\ \boldsymbol{u}_p^q = \boldsymbol{\gamma}}} 2\tilde{c}_p^q - f_{\boldsymbol{\gamma}} = 0, \tag{10}$$

for each $\boldsymbol{\gamma} \in \{\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k\}_{i,k}$. Since $\overline{a}_i^k, \overline{b}_i^k \geq 0$, one has $\tilde{a} \geq \frac{\varepsilon}{2} \geq \delta$, $\tilde{b} \geq \varepsilon \geq \delta$. In addition, for all $(a, b, c) \in \mathbf{K}$, one has $2ab - c^2 \geq 0$, which implies that $-\sqrt{2ab} \leq c \leq \sqrt{2ab}$ and $2a + b + c \geq 2a + b - \sqrt{2ab} = (\sqrt{2a} - \sqrt{b})^2 + \sqrt{2ab} \geq 0$. Hence, one has

$$2\tilde{a}_i^k \tilde{b}_i^k - (\tilde{c}_i^k)^2 = (2\overline{a}_i^k + \varepsilon)(\overline{b}_i^k + \varepsilon) - (\overline{c}_i^k - \frac{\varepsilon}{2})^2 = 2\overline{a}_i^k \overline{b}_i^k - (\overline{c}_i^k)^2 + (2\overline{a}_i^k + \overline{b}_i^k + \overline{c}_i^k)\varepsilon + \frac{3\varepsilon^2}{4} \geq \frac{3\varepsilon^2}{4} \geq \delta.$$

With this choice, one has $\tau(\delta) \leq 2\tau(\varepsilon) + \tau(\frac{3}{4})$. So δ has bit size less than $O(\tau \cdot (4Dd+6)^{3n+3})$.
(2) Let us assume that one relies in Algorithm 5 on a rounding procedure with precision $\hat{\delta}$ so that $|\tilde{a}_i^k - \hat{a}_i^k| \leq \hat{\delta}$ (and similarly for $\tilde{b}_i^k$ and $\tilde{c}_i^k$). By using the triangular inequality, one obtains

$$\left| \sum_{\substack{p,q \\ \boldsymbol{v}_p^q = \boldsymbol{\gamma}}} 2\hat{a}_p^q + \sum_{\substack{p,q \\ \boldsymbol{w}_p^q = \boldsymbol{\gamma}}} \hat{b}_p^q - \sum_{\substack{p,q \\ \boldsymbol{u}_p^q = \boldsymbol{\gamma}}} 2\hat{c}_p^q - f_{\boldsymbol{\gamma}} \right| \leq \sum_{\substack{p,q \\ \boldsymbol{v}_p^q = \boldsymbol{\gamma}}} 2|\hat{a}_p^q - \tilde{\mathscr{A}}_p^q| + \sum_{\substack{p,q \\ \boldsymbol{w}_p^q = \boldsymbol{\gamma}}} |\hat{b}_p^q - \tilde{b}_p^q| + \sum_{\substack{p,q \\ \boldsymbol{u}_p^q = \boldsymbol{\gamma}}} 2|\hat{c}_p^q - \tilde{c}_p^q| \leq 5\eta(\boldsymbol{\gamma})\hat{\delta}, \tag{11}$$

for each $\boldsymbol{\gamma} \in \{\boldsymbol{u}_i^k, \boldsymbol{v}_i^k, \boldsymbol{w}_i^k\}_{i,k}$. Let us consider the projection steps of Algorithm 5. For a_i^k, b_i^k and c_i^k defined from Line 13 to Line 15, one has

$$a_i^k \geq \hat{a}_i^k - \frac{5}{2}\hat{\delta}\,, \tag{12}$$

$$b_i^k \geq \hat{b}_i^k - 5\hat{\delta}\,, \tag{13}$$

$$\tilde{c}_i^k + \frac{5}{2}\hat{\delta} \geq c_i^k\,. \tag{14}$$

Let us choose $\hat{\delta} = \frac{\varepsilon}{5}$. Therefore in the worse case scenario, one has $a_i^k = \tilde{a}_i^k - \frac{\varepsilon}{2} = \overline{a}_i^k$, $b_i^k = \tilde{b}_i^k - \varepsilon = \overline{b}_i^k$ and $c_i^k = \tilde{c}_i^k + \frac{\varepsilon}{2} = \overline{c}_i^k$.

To conclude, if one solves the SOCP problem in Algorithm 5 at precision $\tilde{\delta}$ at least δ and rounding accuracy $\hat{\delta}$, then the output will always provide a valid SOBS decomposition. Both accuracy parameters have bit size less than $O(\tau \cdot (4Dd+6)^{3n+3})$.

The arithmetic complexity of the procedure to solve the SOCP problem at accuracy $\tilde{\delta}$ is derived from § 4.6.2 in Ben-Tal and Nemirovski (2001). There are $3L$ inequality constraints and $3L$ equality constraints (which can themselves be cast as $6L$ inequality constraints), involving $3L$ variables, yielding an overall complexity of $O\left(\sqrt{9L+1} \cdot 3L \cdot ((3L)^2 + 9L + 9L) \cdot \log \frac{L+\delta^2}{\delta}\right)$, which leads to a number of $O\left((\tau(L) + \tau \cdot (4Dd+6)^{3n+3})\, L^{3.5}\right)$ arithmetic operations. Using the bound on L stated in Lemma 29, we obtain a total number of $\widetilde{O}\left(\tau \cdot 5^{3n+3} d^{6.5n+3} n^{10.5} (1+nd^2)^{3(n+1)^2}\right)$ arithmetic operations. The simplex cover algorithm `SimplexCover` solves $O(t)$ linear programs involving $O(t)$ variables. The complexity of a linear program is $O(t^{2.5})$. So the total complexity is $O(t^{3.5})$. Let p be as in Lemma 26 which is bounded by $(1+nd^2)^{\frac{n+1}{2}}$. The procedure `MedSeq`(p,q) requires $O(\log_2 p)$ iterations and each iteration needs $O(\log_2 p)$ arithmetic operations. Since we have $O(t)$ circuits and each circuit corresponds to (at most) n mediated sequences, the total complexity is $O(tn \log_2(p)^2) = O(tn^3 \log_2(1+nd^2)^2)$. Overall, this shows that the total number of arithmetic operations required by these two steps of `ExactSOBS` have a negligible cost by comparison with the numerical SOCP procedure, yielding the desired result. □

7. Numerical experiments

In this section we present numerical results of the proposed algorithms for unconstrained POPs. Our tool, called `SONCSOCP`, implements the simplex cover procedure (see Algorithm 4) as well as the procedure `MedSet` (see Algorithm 3) computing rational mediated sets and computes the optimal value ξ_{socp} of the SOCP (8) with Mosek; see Andersen and Andersen (2000). All experiments were performed on an Intel Core i5-8265U@1.60GHz CPU with 8GB RAM memory and WINDOWS 10 system. `SONCSOCP` is available at github:`SONCSOCP`.

Our benchmarks are issued from the database of randomly generated polynomials provided by Seidler and de Wolff in Seidler and de Wolff (2018). Depending on the Newton polytope, these benchmarks are divided into three classes: the ones with standard simplices, the ones with general simplices and the ones with arbitrary Newton polytopes. See Seidler and de Wolff (2018) for the details on the construction of these polynomials.

7.1. Computing a lower bound via SONC optimization

In this subsection, we perform SONC optimization (8) and compare the performance of `SONCSOCP` with that of `POEM`, which relies on the ECOS solver to solve geometric programs

Table 1: The notation

n	number of variables
d	degree
t	number of terms
l	lower bound on the number of inner terms
opt	optimal value
time	running time in seconds
bit	bit size

(see Seidler and de Wolff (2018) for more details). To measure the quality of a given lower bound ξ_{lb}, we rely on the "`local_min`" function available in `POEM` which computes an upper bound ξ_{min} on the minimum of a polynomial. The relative optimality gap is defined by $\frac{|\xi_{min}-\xi_{lb}|}{|\xi_{min}|}$.

Standard simplex. For the standard simplex case, we take 10 polynomials of different types (labeled by N). Running time and lower bounds obtained with `SONCSOCP` and `POEM` are displayed in Table 2. Note that for polynomials with $\Lambda(\cdot)$ forming a trellis, the simplex cover is unique, thus the bounds obtained by `SONCSOCP` and `POEM` are the same theoretically, which is also reflected in Table 2. For each polynomial, the relative optimality gap is less than 1% and for 8 out of 10 polynomials, it is less than 0.1% (see Figure 3).

	N	1	2	3	4	5	6	7	8	9	10
	n	10	10	10	20	20	20	30	30	40	40
	d	40	50	60	40	50	60	50	60	50	60
	t	20	20	20	30	30	30	50	50	100	100
time	`SONCSOCP`	0.04	0.04	0.04	0.14	0.14	0.13	0.43	0.40	2.23	2.21
	`POEM`	0.26	0.27	0.26	0.43	0.44	0.42	1.78	1.79	2.20	2.25
opt	`SONCSOCP`	3.52	3.52	3.52	2.64	2.64	2.64	2.94	2.94	4.41	4.41
	`POEM`	3.52	3.52	3.52	2.64	2.64	2.64	2.94	2.94	4.41	4.41

Table 2: Results for the standard simplex case

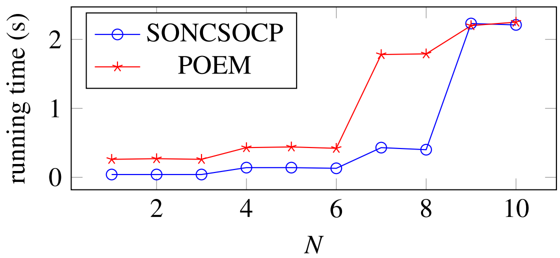


Figure 2: Running time for the standard simplex case

General simplex. Here, we take 10 polynomials of different types (labeled by N). Running time and lower bounds obtained with `SONCSOCP` and `POEM` are displayed in Table 3. As before, the SONC lower bounds obtained by `SONCSOCP` and `POEM` are the same. For each polynomial except for the one corresponding to $N = 7$, the relative optimality gap is within 30%, and for 6 out of 10 polynomials, the gap is below 1% (see Figure 5). `POEM` fails to obtain a lower bound for the

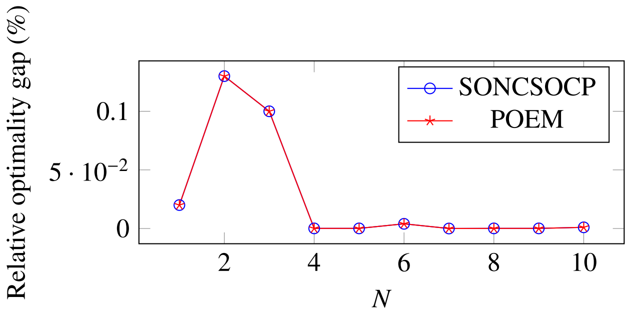


Figure 3: Relative optimality gap for the standard simplex case

instance $N = 10$ by returning $-$Inf. Figure 4 shows that, overall, the running times of `SONCSOCP` and `POEM` are close. `SONCSOCP` is faster than `POEM` for the instance $N = 6$, possibly because better performance is obtained when the degree is relatively low.

	N	1	2	3	4	5	6	7	8	9	10
	n	10	10	10	10	10	10	10	10	10	10
	d	20	30	40	50	60	20	30	40	50	60
	t	20	20	20	20	20	30	30	30	30	30
time	`SONCSOCP`	0.32	0.29	0.36	0.48	0.54	0.56	0.73	0.88	1.04	1.04
	`POEM`	0.28	0.31	0.31	0.31	0.43	0.74	0.75	0.74	0.72	0.76
opt	`SONCSOCP`	1.18	0.22	0.38	0.90	0.06	4.00	-4.64	1.62	2.95	5.40
	`POEM`	1.18	0.22	0.38	0.90	0.06	4.00	-4.64	1.62	2.95	$-$Inf

Table 3: Results for the general simplex case

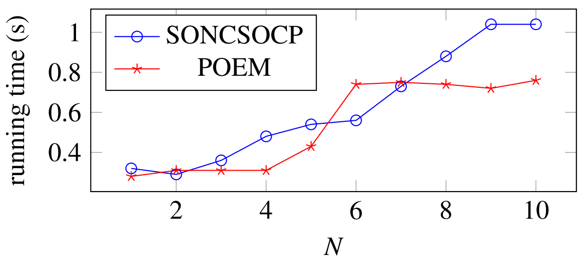


Figure 4: Running time for the general simplex case

Arbitrary polytope. For the arbitrary polytope case, we take 20 polynomials of different types (labeled by N). With regard to these instances, `POEM` always throws an error "expected square matrix". Running time and lower bounds obtained with `SONCSOCP` are displayed in Table 4. The relative optimality gap is always within 25% and within 1% for 17 out of 20 polynomials (see Figure 6).

7.2. *Exact nonnegativity certificates*

In this subsection, we certify nonnegativity of polynomials by computing an exact SOBS decomposition via the procedure `ExactSOBS`. The benchmarks are selected to be in the interior of the SONC cone. We take the number of variables $n = 4, 8, 10$, the degree $d = 10, 20, 30$ and

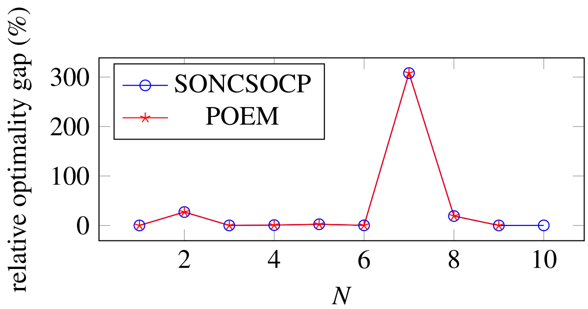


Figure 5: Relative optimality gap for the general simplex case

N		1	2	3	4	5	6	7	8	9	10
n		10	10	10	10	10	10	10	10	10	10
d		20	20	20	30	30	30	40	40	40	50
t		30	100	300	30	100	300	30	100	300	30
l		15	71	231	15	71	231	15	71	231	15
SONCSOCP	time	0.38	1.75	6.86	0.64	3.13	11.3	0.72	4.01	14.6	0.76
	opt	0.70	3.32	31.7	3.31	15.3	3.31	0.47	5.42	38.7	1.56
N		11	12	13	14	15	16	17	18	19	20
n		10	10	10	10	10	20	20	20	20	20
d		50	50	60	60	60	30	30	40	40	40
t		100	300	30	100	300	50	100	50	100	200
l		71	231	15	71	231	5	15	5	15	35
SONCSOCP	time	4.41	16.8	1.84	11.2	42.4	3.20	8.84	2.60	10.5	38.7
	opt	0.20	7.00	3.31	2.52	23.4	0.70	4.91	4.13	2.81	9.97

Table 4: Results for the arbitrary polytope case

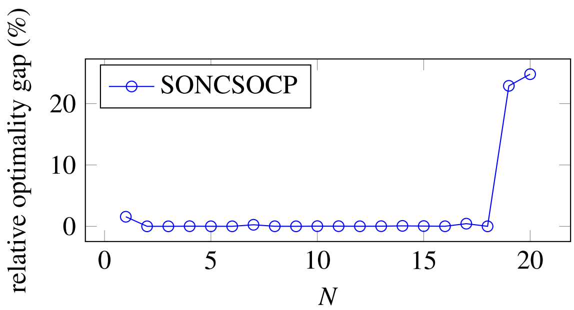


Figure 6: Relative optimality gap for the arbitrary polytope case

the number of terms $t = 20, 30, 50, 100$. We set the precision parameter $\tilde{\delta}$ for the SOCP solver (Mosek) to be 10^{-8} and the rounding precision $\hat{\delta}$ to be 10^{-5}. The results are displayed in Table 5, 6 and 7 respectively. For polynomials in each table, their Newton polytope is either a standard simplex or an arbitrary polytope. The notation "-" indicates that the corresponding polynomial does not exist in the database.

As we can see from the tables, the procedure `ExactSOBS` is very efficient to compute an exact SOBS decomposition. Even for the largest instance ($n = 10, d = 30, t = 100$), it takes around 6 seconds to get the result. We also observe that (somewhat surprisingly) the time spent for symbolic computation and the time spent for numerical computation are nearly equal. Note that we were not able to compare our certification procedure with the current version of the `POEM` software library, since the rounding-projection procedure from Magron et al. (2019b) seems to be not available anymore.

	$n = 4$, standard simplex						$n = 4$, arbitrary polytope					
t	$d = 10$		$d = 20$		$d = 30$		$d = 10$		$d = 20$		$d = 30$	
	time	bit	time	bit	time	bit	time	bit	time	bit	time	bit
20	0.06	123	0.06	154	0.07	162	0.07	109	0.07	129	0.13	118
30	0.09	189	0.09	193	0.09	199	0.09	135	0.12	106	0.16	121
50	0.09	248	0.12	282	0.14	332	0.22	195	0.13	124	0.35	226
100	0.16	362	0.18	364	-	-	-	-	-	-	0.48	210

Table 5: Exact SOBS decompositions for polynomials with four variables

	$n = 8$, standard simplex						$n = 8$, arbitrary polytope					
t	$d = 10$		$d = 20$		$d = 30$		$d = 10$		$d = 20$		$d = 30$	
	time	bit	time	bit	time	bit	time	bit	time	bit	time	bit
20	-	-	0.07	135	0.07	144	0.13	101	0.27	138	0.19	140
30	-	-	0.11	177	0.13	184	0.17	141	0.31	143	0.56	172
50	-	-	0.18	273	0.18	279	0.29	144	0.75	212	1.28	206
100	-	-	0.45	356	0.46	419	0.81	277	2.61	333	3.45	264

Table 6: Exact SOBS decompositions for polynomials with eight variables

	$n = 10$, standard simplex						$n = 10$, arbitrary polytope					
t	$d = 10$		$d = 20$		$d = 30$		$d = 10$		$d = 20$		$d = 30$	
	time	bit	time	bit	time	bit	time	bit	time	bit	time	bit
20	-	-	0.06	119	0.07	124	-	-	0.29	158	0.58	186
30	-	-	0.16	173	0.13	185	0.11	140	0.84	169	1.46	175
50	-	-	0.24	253	0.25	256	0.31	178	1.43	179	2.77	224
100	-	-	0.53	431	0.44	401	-	-	3.98	307	6.38	302

Table 7: Exact SOBS decompositions for polynomials with ten variables

8. Conclusions

In this paper, we provide a constructive proof that each SONC cone admits a SOC representation. Based on this, we propose an algorithm to compute a lower bound for unconstrained POPs via SOCP. Numerical experiments demonstrate the efficiency of our algorithm even when the number of variables and the degree are fairly large. Even though the complexity of our algorithm depends on the degree in theory, it turns out that this dependency is rather mild. For all numerical examples tested in this paper, the running time is below one minute even for polynomials of degree up to 60.

Since the running time is satisfactory, the main concern of SONC-based algorithms for sparse polynomial optimization might be the quality of obtained lower bounds. For many examples tested in this paper, the relative optimality gap is within 1%. However, it can happen that the SONC lower bound is not accurate and this cannot be avoided by computing an optimal simplex cover. To improve the quality of such bounds, it is mandatory to find more complex representations of nonnegative polynomials, which involve SONC polynomials. We will investigate it in the future.

Another line of research is to extend our SONC-SOCP framework to constrained polynomial optimization. Note that constrained polynomial optimization based on the SAGE certificate has been studied in Murray et al. (2020). It is also worth investigating how the SOCP methodology works in the SAGE context.

Acknowledgements. Both authors were supported by the Tremplin ERC Stg Grant ANR-18-ERC2-0004-01 (T-COPS project). The second author was supported by the FMJH Program PGMO (EPICS project) and EDF, Thales, Orange et Criteo. This work has benefited from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie Actions, grant agreement 813211 (POEMA) as well as from the AI Interdisciplinary Institute ANITI funding, through the French "Investing for the Future PIA3" program under the Grant agreement n°ANR-19-PI3A-0004.

References

A. A. Ahmadi and A. Majumdar. DSOS and SDSOS optimization: more tractable alternatives to sum of squares and semidefinite optimization. *SIAM Journal on Applied Algebra and Geometry*, 3(2):193–230, 2019. 3

F. Alizadeh and D. Goldfarb. Second-order cone programming. *Mathematical programming*, 95(1):3–51, 2003. 3

E. D. Andersen and K. D. Andersen. The Mosek Interior Point Optimizer for Linear Programming: An Implementation of the Homogeneous Algorithm. In Hans Frenk, Kees Roos, Tamás Terlaky, and Shuzhong Zhang, editors, *High Performance Optimization*, volume 33 of *Applied Optimization*, pages 197–232. Springer US, 2000. ISBN 978-1-4419-4819-9. doi: 10.1007/978-1-4757-3216-0_8. URL http://dx.doi.org/10.1007/978-1-4757-3216-0_8. 23

G. Averkov. Optimal size of linear matrix inequalities in semidefinite approaches to polynomial optimization. *SIAM Journal on Applied Algebra and Geometry*, 3(1):128–151, 2019. 3, 12

A. Ben-Tal and A. Nemirovski. *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*. SIAM, 2001. 3, 12, 23

V. Chandrasekaran and P. Shah. Relative Entropy Relaxations for Signomial Optimization. *SIAM J. Optim.*, 26(2): 1147–1173, 2016. 1

M. Dressler, S. Iliman, and T. De Wolff. An approach to constrained polynomial optimization via nonnegative circuit polynomials and geometric programming. *Journal of Symbolic Computation*, 91:149–172, 2019. 2

Mareike Dressler, Janin Heuer, Helen Naumann, and Timo de Wolff. Global optimization via the dual sonc cone and linear programming. In *Proceedings of the 45th International Symposium on Symbolic and Algebraic Computation*, pages 138–145, 2020. 2

H. Fawzi. On representing the positive semidefinite cone using the second-order cone. *Mathematical Programming*, 175 (1-2):109–118, 2019. 11

Hamza Fawzi and James Saunderson. Lieb's concavity theorem, matrix geometric means, and semidefinite optimization. *Linear Algebra and its Applications*, 513:240–263, 2017. 13

Jacob Hartzer, Olivia Röhrig, Timo de Wolff, and Oğuzhan Yürük. Initial steps in the classification of maximal mediated sets. *Journal of Symbolic Computation*, 2020. 5

S. Iliman and T. De Wolff. Amoebas, nonnegative polynomials and sums of squares supported on circuits. *Research in the Mathematical Sciences*, 3(1):9, 2016a. 1, 4, 5

S. Iliman and T. De Wolff. Lower bounds for polynomials with simplex newton polytopes based on geometric programming. *SIAM Journal on Optimization*, 26(2):1128–1146, 2016b. 2

C. Josz. *Application of polynomial optimization to electricity transmission networks*. Theses, Université Pierre et Marie Curie - Paris VI, July 2016. URL `https://tel.archives-ouvertes.fr/tel-01478431`. 2

Lukas Katthän, Helen Naumann, and Thorsten Theobald. A unified framework of sage and sonc polynomials and its duality theory. *Mathematics of Computation*, 90(329):1297–1322, 2021. 1

Igor Klep, Victor Magron, and Janez Povh. Sparse noncommutative polynomial optimization. *Mathematical Programming*, pages 1–41, 2021. 2

V. Magron and M. Safey El Din. RealCertify: a Maple package for certifying non-negativity. In *ISSAC'18: Proceedings of the 2018 ACM International Symposium on Symbolic and Algebraic Computation*. ACM, New York, NY, USA, 2018a. 2

V. Magron and M. Safey El Din. On Exact Reznick, Hilbert-Artin and Putinar's Representations. *arXiv preprint arXiv:1811.10062*, 2018b. 21

V. Magron and M. Safey El Din. On Exact Polya and Putinar's Representations. In *ISSAC'18: Proceedings of the 2018 ACM International Symposium on Symbolic and Algebraic Computation*. ACM, New York, NY, USA, 2018c. 2

V. Magron, G. Constantinides, and A. Donaldson. Certified Roundoff Error Bounds Using Semidefinite Programming. *ACM Trans. Math. Softw.*, 43(4):1–34, 2017. 2

V. Magron, M. Safey El Din, and M. Schweighofer. Algorithms for weighted sum of squares decomposition of non-negative univariate polynomials. *Journal of Symbolic Computation*, 93:200–220, 2019a. 2

V. Magron, H. Seidler, and T. de Wolff. Exact optimization via sums of nonnegative circuits and arithmetic-geometric-mean-exponentials. In *Proceedings of the 2019 on International Symposium on Symbolic and Algebraic Computation*, ISSAC '19, page 291–298, New York, NY, USA, 2019b. 2, 20, 27

M. Mignotte. *Mathematics for Computer Algebra*. Springer-Verlag New York, Inc., New York, NY, USA, 1992. 20

Riley Murray, Venkat Chandrasekaran, and Adam Wierman. Signomial and polynomial optimization via relative entropy and partial dualization. *Mathematical Programming Computation*, pages 1–39, 2020. 28

Riley Murray, Venkat Chandrasekaran, and Adam Wierman. Newton polytopes and relative entropy optimization. *Foundations of Computational Mathematics*, pages 1–35, 2021. 1, 2

D. Papp. Duality of sum of nonnegative circuit polynomials and optimal SONC bounds. *arXiv preprint arXiv:1912.04718*, 2019. 16

H. Peyrl and P.A. Parrilo. Computing sum of squares decompositions with rational coefficients. *Theoretical Computer Science*, 409(2):269–281, 2008. 2, 18

Victoria Powers and Bruce Reznick. A note on mediated simplices. *Journal of Pure and Applied Algebra*, 225(7): 106608, 2021. 5

B. Reznick. Forms derived from the arithmetic-geometric inequality. *Mathematische Annalen*, 283(3):431–464, 1989. 5

Guillaume Sagnol. On the semidefinite representation of real functions applied to symmetric matrices. *Linear Algebra and its Applications*, 439(10):2829–2843, 2013. 13

H. Seidler and T. de Wolff. An experimental comparison of sonc and sos certificates for unconstrained optimization. *arXiv preprint arXiv:1808.08431*, 2018. 2, 16, 23, 24

H. Waki, S. Kim, M. Kojima, and M. Muramatsu. Sums of Squares and Semidefinite Programming Relaxations for Polynomial Optimization Problems with Structured Sparsity. *SIAM Journal on Optimization*, 17(1):218–242, 2006. 2

J. Wang. Nonnegative polynomials and circuit polynomials. *arXiv preprint arXiv:1804.09455*, 2018. 1, 2, 4, 11

J. Wang and V. Magron. Exploiting term sparsity in Noncommutative Polynomial Optimization. *arXiv preprint arXiv:2010.06956*, 2020a. 2

J. Wang and V. Magron. A second Order Cone Characterization for Sums of Nonnegative Circuits. In *ISSAC'20: Proceedings of the 2020 ACM International Symposium on Symbolic and Algebraic Computation*, 2020b. 3

J. Wang, V. Magron, J.-B. Lasserre, and N. H. A. Mai. CS-TSSOS: Correlative and term sparsity for large-scale polynomial optimization. *arXiv:2005.02828*, 2020. 2

J. Wang, M. Maggio, and V. Magron. SparseJSR: A Fast Algorithm to Compute Joint Spectral Radius via Sparse SOS Decompositions. *Accepted by ACC'21*, 2021a. 2

Jie Wang, Victor Magron, and Jean-Bernard Lasserre. Tssos: A moment-sos hierarchy that exploits term sparsity. *SIAM Journal on Optimization*, 31(1):30–58, 2021b. 2

Jie Wang, Victor Magron, and Jean-Bernard Lasserre. Chordal-tssos: a moment-sos hierarchy that exploits term sparsity with chordal extension. *SIAM Journal on Optimization*, 31(1):114–141, 2021c. 2