ALGORITHMIC ASPECTS OF SUMS OF HERMITIAN SQUARES

SABINE BURGDORF[1,3], KRISTIJAN CAFUTA, IGOR KLEP[2,3], AND JANEZ POVH[4]

Abstract. This paper presents an algorithm and its implementation in the software package NCSOStools for finding sums of hermitian squares and commutators decompositions for polynomials in noncommuting variables. The algorithm is based on noncommutative analogs of the classical Gram matrix method and the Newton polytope method, which allows us to use semidefinite programming. For rational polynomials numerical evidence can be tweaked to obtain an exact certificate using rational numbers. In the presence of Slater points, the Peyrl-Parrilo rounding and projecting method applies. On the other hand, in the absence of strict feasibility, a variant of the facial reduction is proposed to reduce the size of the semidefinite program and to enforce the existence of Slater points.

1. Introduction

The main question studied in this paper is whether a given real polynomial in noncommuting variables (nc polynomial) can be decomposed as a sum of hermitian squares and commutators. Using semidefinite programming we obtain numerical evidence and, if the input polynomial is rational, we can employ facial reduction to extract an exact rational certificate.

1.1. **Motivation.** The interest in finding decompositions of an noncommutative (nc) polynomial as a sum of hermitian squares and commutators is based on the following simple fact. If such a decomposition exists, the given nc polynomial is necessarily trace-positive, i.e., all of its evaluations at tuples of matrices have nonnegative trace. Following Helton's seminal paper [Hel02], this belongs to *free real algebraic geometry* (including *free positivity*) where one is interested in positivity of nc polynomials. Much of today's interest in (free) real algebraic geometry is due to its powerful applications. For instance, the use of sums of squares and the truncated moment problem for polynomial optimization on $\mathbb{R}^n$ established by Lasserre and Parrilo [Las01, Las09, Par03, PS03, Sch05] is nowadays a common fact in real algebraic geometry with applications to control theory, mathematical finance and operations research. In the free context there are many facets of applications as well. A nice survey on connections to control theory, systems engineering and optimization is given by de Oliveira, Helton, McCullough, Putinar [dOHMP08]. Applications of the free case to quantum physics are explained e.g. by Pironio, Navascués, Acín [PNA10] who also consider computational aspects related to sums of hermitian squares (without commutators). Trace-positive nc polynomials fill a gap

Date: 19/09/2011 at 15:27:45.

2010 *Mathematics Subject Classification.* Primary 90C22, 13J30; Secondary 47A57, 08B20.

Key words and phrases. sum of squares, semidefinite programming, noncommutative polynomial, Matlab toolbox, Newton polytope, free positivity.

[1] Partially supported by the Zukunftskolleg Konstanz.

[2] Supported by the Slovenian Research Agency (project no. J1-3608 and program no. P1-0222).

[3] Partially supported by the French-Slovene partnership project Proteus 20208ZM.

[4] Supported by the Slovenian Research Agency (program no. P1-0297).

between these two cases, so we expect a considerable development of their applications in the future.

On the theoretical level, trace-positive nc polynomials arise e.g. in the Lieb-Seiringer reformulation of the famous Bessis-Moussa-Villani (BMV) conjecture [BMV75] from statistical quantum mechanics.[1] Many modern results on this problem have been obtained with the aid of computer programs – using sums of hermitian squares and commutators decompositions – written in an ad-hoc manner. This connection will be explained in detail later to demonstrate the usage of our proposed algorithm. In addition, trace-positive nc polynomials occur naturally in von Neumann algebras and functional analysis. For instance, Connes' embedding problem [Con76] on finite II_1-factors is a question about the existence of a certain type of sum of hermitian squares (sohs) certificates for trace-positive nc polynomials [KS08a]. It is widely believed that Connes' conjecture is false and our results will enable us to look for a counterexample using a computer algebra system.

As a consequence of this surge of interest in free real algebraic geometry and sums of (hermitian) squares of nc polynomials we developed `NCSOStools` [CKP11] – an open source Matlab toolbox for solving such problems using *semidefinite programming*. As a side product our toolbox implements symbolic computation with noncommuting variables in Matlab.

1.2. Related work and contribution.

We will denote the convex cone of sums of hermitian squares and commutators by Θ^2.

Sum of hermitian squares decompositions were intensively studied by several authors. An outstanding result is due to Helton [Hel02], who has proved that for an nc polynomial $f \in \mathbb{R}\langle \underline{X} \rangle$, we have $f(A_1, \ldots, A_n) \succeq 0$ for *all* symmetric matrices A_i of the same size if and only if f is a sum of hermitian squares. We also refer the reader to [McC01, MP05] for nice alternative proofs. In [KP10] the third and the fourth author presented an algorithm for finding sums of hermitian squares decompositions (without commutators) using a variant of the Gram matrix method. The key ingredient of the method was semidefinite programming together with the Newton chip method to reduce the size of the semidefinite programming problems, which eventually turned out to be linear in the length and in the degree of the nc polynomial. Extending this method we proposed in [BCKP] another variant of the Gram matrix method to answer the question whether $f \in \Theta^2$ holds. Similarly to [KP10], semidefinite programming was the main tool. However, an important topic that remained open in [BCKP] was how to provide *efficiently* numerical or exact certificates for either $f \in \Theta^2$ or $f \notin \Theta^2$.

Therefore the main contribution of this paper is the following:

(a) We present the *tracial Gram matrix method*, tailored for sums of hermitian squares and commutators, to resolve the separability question for Θ^2. We also present an improvement of this method using a *cyclic* extension of the *Newton chip method* from [KP10] which reduces the dimensions of the underlying semidefinite programs to a more manageable level. This method can be understood as a noncommutative generalization of the classical Newton polytope method [Rez78].

(b) Once we know whether a given *rational* nc polynomial f belongs to Θ^2 we want to obtain an *exact* (rational) certificate. Following ideas from [PP08] we propose an algorithm which under strict feasibility assumption theoretically and practically always yields a rational certificate. On the other hand, in the absence of strict feasibility, a variant of the facial reduction [BW81] (in our case projecting onto the orthogonal

[1] Recently, Stahl announced a proof of the original formulation of the BMV conjecture [Sta].

complement of the null space of the analytic center) is used to reduce the size of the semidefinite program and enforce the existence of Slater points.

(c) We provide new rational certificates for three instances of nc polynomials related to the Bessis-Moussa-Villani conjecture to demonstrate how to use the proposed algorithm as implemented in `NCSOStools`.

2. Preliminaries

2.1. **Words, nc polynomials and involution.**

Fix $n \in \mathbb{N}$ and let $\langle \underline{X} \rangle$ be the set of *words* in the n noncommuting letters $X_1, \dots, X_n$ (including the empty word denoted by 1), i.e., $\langle \underline{X} \rangle$ is the monoid freely generated by $\underline{X} := (X_1, \dots, X_n)$. We consider linear combinations $\sum_w a_w w$ with $a_w \in \mathbb{R}$, $w \in \langle \underline{X} \rangle$ of words in the n letters $\underline{X}$ which we call *nc polynomials.* The set of all nc polynomials is actually a *free algebra*, which we denote by $\mathbb{R}\langle \underline{X} \rangle$. An element of the form aw where $a \in \mathbb{R} \setminus \{0\}$ and $w \in \langle \underline{X} \rangle$ is called a *monomial* and a its *coefficient.* The length of the longest word in an nc polynomial $f \in \mathbb{R}\langle \underline{X} \rangle$ is the *degree* of f and is denoted by $\deg f$. The set of all nc polynomials of degree $\leq d$ will be denoted by $\mathbb{R}\langle \underline{X} \rangle_{\leq d}$. The length of the shortest word appearing in $f \in \mathbb{R}\langle \underline{X} \rangle$ is called the *min-degree* of f and denoted by $\operatorname{mindeg} f$. Also of interest is the degree of f in X_i, $\deg_i f$ and the minimum degree of f in X_i, $\operatorname{mindeg}_i f$. If an nc polynomial f involves only two variables, we use $\mathbb{R}\langle X, Y \rangle$ instead of $\mathbb{R}\langle X_1, X_2 \rangle$.

We equip $\mathbb{R}\langle \underline{X} \rangle$ with the *involution* $*$ that fixes $\mathbb{R} \cup \{\underline{X}\}$ pointwise and thus reverses words, e.g. $(X_1 X_2^2 X_3 - 2X_3^3)^* = X_3 X_2^2 X_1 - 2X_3^3$. Hence $\mathbb{R}\langle \underline{X} \rangle$ is the $*$-algebra freely generated by n symmetric letters. The involution extends naturally to matrices (in particular, to vectors) over $\mathbb{R}\langle \underline{X} \rangle$. For instance, if $V = (v_i)$ is a (column) vector of nc polynomials $v_i \in \mathbb{R}\langle \underline{X} \rangle$, then V^* is the row vector with components v_i^*. We use V^t to denote the row vector with components v_i.

2.2. **Sum of hermitian squares and commutators.**

Let $\operatorname{Sym} \mathbb{R}\langle \underline{X} \rangle$ denote the set of all *symmetric elements*, that is,

$$\operatorname{Sym} \mathbb{R}\langle \underline{X} \rangle := \{ f \in \mathbb{R}\langle \underline{X} \rangle \mid f = f^* \}.$$

An nc polynomial of the form $g^* g$ is called a *hermitian square* and the set of all sums of hermitian squares will be denoted by Σ^2. Clearly, $\Sigma^2 \subsetneq \operatorname{Sym} \mathbb{R}\langle \underline{X} \rangle$.

Example 2.1. The nc polynomial $f = X^2 - X^2 Y - Y X^2 + Y X^2 Y + X Y^2 X$ is a sum of hermitian squares, in fact, $f = (X - XY)^*(X - XY) + (YX)^*(YX)$. In particular, $f(A, B)$ is positive semidefinite for all symmetric matrices A, B. For a concrete example, with $A = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & -2 \\ 0 & -2 & 1 \end{bmatrix}$ and $B = \begin{bmatrix} 1 & 0 & 1 \\ 0 & -2 & -1 \\ 1 & -1 & 1 \end{bmatrix}$, we have

$$f(A, B) = A^2 - A^2 B - B A^2 + B A^2 B + A B^2 A = \begin{bmatrix} 7 & 12 & 0 \\ 12 & 39 & 0 \\ 0 & 0 & 25 \end{bmatrix} \succeq 0.$$

The next notation we need is cyclic equivalence [KS08a] whose definition is motivated by the fact that we are interested in the *trace* of a given nc polynomial under matrix evaluations.

Definition 2.2. An element of the form $[p, q] := pq - qp$, where p, q are polynomials from $\mathbb{R}\langle \underline{X} \rangle$, is a *commutator*. Polynomials $f, g \in \mathbb{R}\langle \underline{X} \rangle$ are called *cyclically equivalent* ($f \overset{\text{cyc}}{\sim} g$) if

$f-g$ is a sum of commutators:

$$f-g=\sum_{i=1}^{k}[p_i,q_i]=\sum_{i=1}^{k}(p_iq_i-q_ip_i) \text{ for some } k\in\mathbb{N} \text{ and } p_i,q_i\in\mathbb{R}\langle\underline{X}\rangle.$$

It is clear that $\overset{\text{cyc}}{\sim}$ is an equivalence relation. The following remark shows how to test if given nc polynomials are cyclically equivalent.

Remark 2.3.

(a) For $v,w\in\langle\underline{X}\rangle$, we have $v\overset{\text{cyc}}{\sim}w$ if and only if there are $v_1,v_2\in\langle\underline{X}\rangle$ such that $v=v_1v_2$ and $w=v_2v_1$. That is, $v\overset{\text{cyc}}{\sim}w$ if and only if w is a cyclic permutation of v.

(b) Polynomials $f=\sum_{w\in\langle\underline{X}\rangle}a_ww$ and $g=\sum_{w\in\langle\underline{X}\rangle}b_ww$ ($a_w,b_w\in\mathbb{R}$) are cyclically equivalent if and only if for each $v\in\langle\underline{X}\rangle$,

$$\sum_{\substack{w\in\langle\underline{X}\rangle\\ w\overset{\text{cyc}}{\sim}v}}a_w=\sum_{\substack{w\in\langle\underline{X}\rangle\\ w\overset{\text{cyc}}{\sim}v}}b_w. \tag{1}$$

Example 2.4. We have $2X^2Y^2X^3+XY^2X^2+XY^2X^4\overset{\text{cyc}}{\sim}3YX^5Y+YX^3Y$ as

$$\begin{aligned}2X^2Y^2X^3+XY^2X^2+XY^2X^4-(3YX^5Y+YX^3Y)&=\\ &=[2X^2Y,YX^3]+[XY,YX^4]+[XY,YX^2].\end{aligned}$$

Definition 2.5. Let

$$\Theta^2:=\{f\in\mathbb{R}\langle\underline{X}\rangle\mid\exists g\in\Sigma^2:\ f\overset{\text{cyc}}{\sim}g\}$$

denote the convex cone of all nc polynomials cyclically equivalent to a sum of hermitian squares. By definition, the elements in Θ^2 are exactly the nc polynomials which can be written as sums of hermitian squares and commutators.

Example 2.6. Consider $f=X^2Y^2+XY^2X+XYXY+YX^2Y+YXYX+Y^2X^2\in\mathbb{R}\langle X,Y\rangle$. This nc polynomial is of the form

$$\begin{aligned}f&=(XYXY+YXYX+XY^2X+YX^2Y)+2XY^2X+(\text{sum of commutators})\\ &=(XY+YX)^*(XY+YX)+2(YX)^*(YX)+(\text{sum of commutators}),\end{aligned}$$

hence we have $f\in\Theta^2$ taking the nc polynomials $g_1=(XY+YX)$ and $g_2=\sqrt{2}YX$ in the Θ^2-certificate. In particular, $\operatorname{tr}(f(A,B))\geq 0$ for all symmetric matrices A,B but in general $f(A,B)$ is not positive semidefinite. For a concrete example, with $A=\begin{bmatrix}1&0\\0&-2\end{bmatrix}$ and $B=\begin{bmatrix}0&1\\1&2\end{bmatrix}$, we have

$$f(A,B)=\begin{bmatrix}3&18\\18&105\end{bmatrix}\not\succeq 0$$

and $\operatorname{tr}(f(A,B))=108>0$.

Definition 2.7. An nc polynomial $f\in\mathbb{R}\langle\underline{X}\rangle$ is called *trace-positive* if

$$\operatorname{tr}(f(\underline{A}))\geq 0 \text{ for all tuples of symmetric matrices } \underline{A} \text{ of the same size.} \tag{2}$$

Clearly, every nc polynomial cyclically equivalent to a sum of hermitian squares is trace-positive. But there are trace-positive nc polynomials which are *not* members of Θ^2. The easiest example is the noncommutative Motzkin polynomial, $f=XY^4X+YX^4Y-3XY^2X+1$

[KS08a, Example 4.4]. We also refer the reader to [KS08b, Example 3.5] for more sophisticated examples obtained by considering the BMV conjecture. Nevertheless, the obvious Θ^2-certificate for trace-positivity turns out to be very useful in optimization.

3. Implementation and computational algorithms

In this section we discuss an algorithm based on the Gram matrix method for testing the membership in Θ^2 and present an improvement using the tracial version of the Newton polytope which we call the Newton cyclic chip method (Section 3.2). The implementation with the aid of semidefinite programming is presented in Sections 3.3 and 3.4.

3.1. **The tracial Gram matrix method.** Testing whether a given $f \in \mathbb{R}\langle \underline{X}\rangle$ is an element of Σ^2 or Θ^2 can be done efficiently by using semidefinite programming as first observed in [KS08b, Section 3], see also [KP10, BCKP]. The method behind it is a variant of the *Gram matrix method* and is based on the following proposition, which is a natural extension of the results for sums of hermitian squares (cf. [Hel02, Section 2.2] or [KP10, Theorem 3.1 and Algorithm 1]), which are in turn variants of the classical result for polynomials in commuting variables due to Choi, Lam and Reznick ([CLR95, Section 2]; see also [Par03]).

Proposition 3.1. *Let W be the vector of all words $w \in \langle \underline{X}\rangle$ satisfying $2\deg(w) \leq \deg(f)$, where $f \in \mathbb{R}\langle \underline{X}\rangle$. Then*

(a) *$f \in \Sigma^2$ if and only if there exists a positive semidefinite matrix G such that*

$$f = W^*GW; \tag{3}$$

(b) *$f \in \Theta^2$ if and only if there exists a positive semidefinite matrix G such that*

$$f \overset{\text{cyc}}{\sim} W^*GW; \tag{4}$$

Moreover, given a positive semidefinite matrix G of rank r satisfying (3) or (4), respectively, one can construct nc polynomials $g_1, \dots, g_r \in \mathbb{R}\langle \underline{X}\rangle$ such that

$$f = \sum_{i=1}^{r} g_i^* g_i \tag{5}$$

or

$$f \overset{\text{cyc}}{\sim} \sum_{i=1}^{r} g_i^* g_i, \tag{6}$$

respectively.

Definition 3.2. A (not necessarily positive semidefinite) matrix G satisfying (3) is called *a Gram matrix* for f, while a matrix G satisfying (4) is called *a tracial Gram matrix* for f.

The proof of Proposition 3.1 is straightforward as in the commutative case. We will present a modification of this proposition including improvements using a noncommutative analog of the Newton polytope in Proposition 3.7, so we omit the proof here.

For an nc polynomial $f \in \mathbb{R}\langle \underline{X}\rangle$ the Gram matrix and the tracial Gram matrix are in general *not* unique, hence determining whether $f \in \Sigma^2$ (or $f \in \Theta^2$) amounts to finding *a* positive semidefinite (tracial) Gram matrix from the affine set of all (tracial) Gram matrices for f. Problems like this can in theory be solved *exactly* using quantifier elimination. However, this only works for problems of small size, so a *numerical* approach is needed in practice. Thus we turn to semidefinite programming, which has become a standard tool in the mathematical optimization area in the last two decades. The readers not familiar with this topic are referred to [WSV00, Tod01, VB96].

3.2. **The Newton cyclic chip method.** In this subsection we present a tracial version of the classical Newton polytope used to reduce the size of the Gram matrix needed for a sum of hermitian squares decomposition.

We will need to consider the free monoid $[\underline{x}]$ in *commuting* variables $\underline{x} := (x_1, \dots, x_n)$ and its semigroup algebra $\mathbb{R}[\underline{x}]$ of polynomials in commuting variables. There is a natural mapping $\langle \underline{X} \rangle \to [\underline{x}]$. For a given word $w \in \langle \underline{X} \rangle$ its image under this mapping is called the *commutative collapse* of w and we use $\operatorname{cc}(w)$ to denote it. If needed, we write $\operatorname{cc}(w) = \underline{x}^{\underline{d}_w}$ where $\underline{x}^{\underline{d}} = x_1^{d_1} \cdots x_n^{d_n}$ for $d_i = \deg_i(w) \in \mathbb{N}_0^n$. Similarly, we introduce the commutative collapse of a set of words $V \subseteq \mathbb{R}\langle \underline{X} \rangle$. For $f = \sum_w a_w w \in \mathbb{R}\langle \underline{X} \rangle$ we define

$$\operatorname{cc}(f) := \{\operatorname{cc}(w) \in [\underline{x}] \mid a_w \neq 0\}.$$

Note that the commutative collapse of an nc polynomial is a *set* of words in commuting variables. As an example, $\operatorname{cc}(XY - YX) = \{xy\}$.

We generalize the degree of an nc polynomial as follows: given $\underline{\alpha} = (\alpha_1, \dots, \alpha_n) \in \mathbb{R}^n$ we define the $\underline{\alpha}$*-degree* $\deg_{\underline{\alpha}}$ of a word $w \in \langle \underline{X} \rangle$ as the standard scalar product between $\underline{\alpha}$ and the exponent of the commutative collapse of w, i.e., if $\operatorname{cc}(w) = \underline{x}^{\underline{d}} = x_1^{d_1} \cdots x_n^{d_n}$, then the $\underline{\alpha}$-degree of w is

$$\deg_{\underline{\alpha}} w := \sum_{i=1}^{n} \alpha_i d_i = \langle \underline{\alpha}, \underline{d} \rangle. \tag{7}$$

We also set $\deg_{\underline{\alpha}} 0 := -\infty$. Note that for all $\underline{\alpha} \in \mathbb{R}^n$, we have

$$u \overset{\text{cyc}}{\sim} v \Rightarrow \deg_{\underline{\alpha}} u = \deg_{\underline{\alpha}} v, \tag{8}$$

$$\deg_{\underline{\alpha}}(uv) = \deg_{\underline{\alpha}} u + \deg_{\underline{\alpha}} v. \tag{9}$$

This notion extends naturally to the $\underline{\alpha}$-degree and $\underline{\alpha}$-min-degree of an nc polynomial $f = \sum_w a_w w \in \mathbb{R}\langle \underline{X} \rangle$:

$$\deg_{\underline{\alpha}} f := \max_{a_w \neq 0} \deg_{\underline{\alpha}} w, \quad \operatorname{mindeg}_{\underline{\alpha}} f := \min_{a_w \neq 0} \deg_{\underline{\alpha}} w. \tag{10}$$

As special cases, note that the (total) degree corresponds to the $\underline{\alpha}$ with all ones and the individual i-degrees $\deg_i$ correspond to the standard unit vectors e_i.

Two cyclically equivalent nc polynomials in general *do not* have the same $\underline{\alpha}$-degree. We therefore modify the definition to obtain the more robust *cyclic-*$\underline{\alpha}$*-degree* $\operatorname{cdeg}_{\underline{\alpha}}$ and *cyclic-*$\underline{\alpha}$*-min-degree* $\operatorname{mincdeg}_{\underline{\alpha}}$:

$$\operatorname{cdeg}_{\underline{\alpha}} f := \min_{g \overset{\text{cyc}}{\sim} f} \deg_{\underline{\alpha}} g, \quad \operatorname{mincdeg}_{\underline{\alpha}} f := \max_{g \overset{\text{cyc}}{\sim} f} \operatorname{mindeg}_{\underline{\alpha}} g. \tag{11}$$

For instance, for $f = X_1^2 X_2^2 X_1^2 + X_2^4 X_3^4 - X_3^4 X_2^4 + X_1 X_2 - X_2 X_1 \overset{\text{cyc}}{\sim} X_1^4 X_2^2$ we have

$$\deg_{(1,1,3)} f = 16, \ \operatorname{mindeg}_{(1,1,3)} f = 2, \ \operatorname{cdeg}_{(1,1,3)} f = 6, \ \operatorname{mincdeg}_{(1,1,3)} f = 6.$$

Definition 3.3. Let $w \in \mathbb{R}\langle \underline{X} \rangle$. The canonical representative $[w]$ of w is the first with respect to the lexicographic order among words cyclically equivalent to w. For $f = \sum_w a_w w \in \operatorname{Sym} \mathbb{R}\langle \underline{X} \rangle$ we define the *canonical representative* $[f]$ of f as follows:

$$[f] := \sum_{[w]} a_{[w]} [w] \in \mathbb{R}\langle \underline{X} \rangle.$$

That is, $[f]$ contains only canonical representatives of words from f with coefficients

$$a_{[w]} := \sum_{u \overset{\mathrm{cyc}}{\sim} w} a_u.$$

For example, if $f = 2Y^2X^2 - XY^2X + XY - YX$, then $[f] = X^2Y^2$.

Proposition 3.4.

(1) *If* $f = \sum_w a_w w \overset{\mathrm{cyc}}{\sim} g = \sum_w b_w w$, *then* $a_{[w]} = b_{[w]}$ *for all* $w \in \langle \underline{X} \rangle$.
(2) *For all* $\underline{\alpha} \in \mathbb{R}^n$ *and* $f \in \mathbb{R}\langle \underline{X} \rangle$ *we have* $\operatorname{cdeg}_{\underline{\alpha}} f = \deg_{\underline{\alpha}}[f]$ *and* $\operatorname{mincdeg}_{\underline{\alpha}} f = \operatorname{mindeg}_{\underline{\alpha}}[f]$.

Proof. Property (1) is obvious. Let us consider (2). Since $f \overset{\mathrm{cyc}}{\sim} [f]$, $\operatorname{cdeg}_{\underline{\alpha}} f \le \deg_{\underline{\alpha}}[f]$. Suppose there exists $g \overset{\mathrm{cyc}}{\sim} f$ with $\deg_{\underline{\alpha}_0} g < \deg_{\underline{\alpha}_0}[f]$ for some $\underline{\alpha}_0 \in \mathbb{R}^n$. There is a word $[w]$ with $\deg_{\underline{\alpha}_0}[w] = \deg_{\underline{\alpha}_0}[f]$, and the coefficient of $[w]$ in $[f]$ is non-zero. But by the first part of the proposition the same is true for g, hence $\deg_{\underline{\alpha}_0} g \ge \deg_{\underline{\alpha}_0}[f]$, which is a contradiction. The second part of property (2) follows using the same line of reasoning. ■

Lemma 3.5. *If* $f \overset{\mathrm{cyc}}{\sim} g = \sum_i g_i^* g_i$, *then* $\operatorname{cdeg}_{\underline{\alpha}} f = \deg_{\underline{\alpha}} g$ *and* $\operatorname{mincdeg}_{\underline{\alpha}} f = \operatorname{mindeg}_{\underline{\alpha}} g$ *for all* $\underline{\alpha} \in \mathbb{R}^n$.

Proof. If $g = 0$ then lemma is true for trivial reasons. Otherwise, by definition, $\operatorname{cdeg}_{\underline{\alpha}} f \le \deg_{\underline{\alpha}} g$ for all $\underline{\alpha} \in \mathbb{R}^n$. Suppose there exists $\underline{\alpha}_0 \in \mathbb{R}^n$ with $\operatorname{cdeg}_{\underline{\alpha}_0} f < \deg_{\underline{\alpha}_0} g$. For $[f] \overset{\mathrm{cyc}}{\sim} f$ we have $\operatorname{cdeg}_{\underline{\alpha}_0} f = \deg_{\underline{\alpha}_0}[f] < \deg_{\underline{\alpha}_0} g =: 2\Delta \neq 0$. Let p_i be the homogeneous part of g_i with $\underline{\alpha}_0$-degree equal to Δ and $r_i = g_i - p_i$. Then $\deg_{\underline{\alpha}_0}(r_i) < \Delta$ and

$$[f] \overset{\mathrm{cyc}}{\sim} \sum g_i^* g_i = \sum (p_i + r_i)^*(p_i + r_i) = \sum p_i^* p_i + \sum p_i^* r_i + \sum r_i^* p_i + \sum r_i^* r_i. \tag{12}$$

Since each word w in $p_i^* r_i$, $r_i^* p_i$ and $r_i^* r_i$ has $\deg_{\underline{\alpha}_0} w < 2\Delta$ (by (9)), none of these can be cyclically equivalent to a nontrivial word in $p_i^* p_i$, because each nontrivial word in $p_i^* p_i$ has $\underline{\alpha}_0$-degree equal to $2\Delta \neq 0$ (note that for each i, $p_i^* p_i \overset{\mathrm{cyc}}{\not\sim} 0$ or $p_i = 0$ due to [KS08b, Lemma 3.2]). Similarly, by assumption there is no word in $[f]$ with $\underline{\alpha}_0$-degree equal to 2Δ. Thus

$$0 \overset{\mathrm{cyc}}{\sim} \sum p_i^* p_i, \quad [f] \overset{\mathrm{cyc}}{\sim} \sum p_i^* r_i + \sum r_i^* p_i + \sum r_i^* r_i.$$

However, [KS08b, Lemma 3.2] implies $p_i = 0$ for all i contradicting $\deg_{\underline{\alpha}_0} g = 2\Delta$. Likewise we prove the second statement $\operatorname{mincdeg}_{\underline{\alpha}} f = \operatorname{mindeg}_{\underline{\alpha}} g$. ■

Lemma 3.6. *Let* $f \in \mathbb{R}\langle \underline{X} \rangle$ *and* $w \in \langle \underline{X} \rangle$. *Then*

$$\operatorname{mincdeg}_{\underline{\alpha}}(f) \le 2\deg_{\underline{\alpha}}(w) \text{ for all } \underline{\alpha} \in \mathbb{R}^n \;\Leftrightarrow\; 2\deg_{\underline{\alpha}}(w) \le \operatorname{cdeg}_{\underline{\alpha}}(f) \text{ for all } \underline{\alpha} \in \mathbb{R}^n. \tag{13}$$

Proof. This is a straightforward consequence of the fact that for all $\underline{\alpha} \in \mathbb{R}^n$ and for all $g \in \mathbb{R}\langle \underline{X} \rangle$ we have $\operatorname{mindeg}_{\underline{\alpha}} g \le 2\deg_{\underline{\alpha}} w$ if and only if $\deg_{-\underline{\alpha}} g \ge 2\deg_{-\underline{\alpha}} w$. ■

The next proposition is the desired improvement of Proposition 3.1 and is the basis for our Newton cyclic chip method.

Proposition 3.7. *Suppose* $f \in \mathbb{R}\langle \underline{X} \rangle$. *Then* $f \in \Theta^2$ *if and only if there exists a positive semidefinite matrix* G *such that*

$$f \overset{\mathrm{cyc}}{\sim} W^* G W, \tag{14}$$

where W *is a vector consisting of all words* $w \in \langle \underline{X} \rangle$ *satisfying*

$$\operatorname{mincdeg}_{\underline{\alpha}}(f) \le 2\deg_{\underline{\alpha}}(w) \le \operatorname{cdeg}_{\underline{\alpha}}(f) \quad \textit{for all } \underline{\alpha} \in \mathbb{R}^n. \tag{15}$$

Conversely, given such a positive semidefinite matrix G of rank r, one can construct nc polynomials $g_1, \dots, g_r \in \mathbb{R}\langle \underline{X} \rangle$ with $f \overset{\text{cyc}}{\sim} \sum_{i=1}^r g_i^* g_i$.

Proof. If $f \overset{\text{cyc}}{\sim} g = \sum_i g_i^* g_i \in \Sigma^2$, then $\deg_{\underline{\alpha}} g = \operatorname{cdeg}_{\underline{\alpha}} f$ for all $\underline{\alpha} \in \mathbb{R}^n$, as follows from Lemma 3.5. Therefore, $2 \deg_{\underline{\alpha}} g_i \leq \deg_{\underline{\alpha}} g = \operatorname{cdeg}_{\underline{\alpha}} f$ for all i and for all $\underline{\alpha} \in \mathbb{R}^n$, hence g_i contains only words satisfying (15). We only verified the right hand side of (15), which suffices by Lemma 3.6. Write $g_i = G_i^t W$, where G_i^t is the (row) vector consisting of the coefficients of g_i. Then $g_i^* g_i = W^t G_i G_i^t W$ and, by setting $G := \sum_i G_i G_i^t$, property (14) clearly holds. The inverse of this claim is obvious.

Given a positive semidefinite $G \in \mathbb{R}^{N \times N}$ of rank r satisfying (14), write $G = \sum_{i=1}^r G_i G_i^t$ for $G_i \in \mathbb{R}^{N \times 1}$. Defining $g_i := G_i^t W$ yields $f \overset{\text{cyc}}{\sim} \sum_{i=1}^r g_i^* g_i$. ■

Given a polynomial $f \in \mathbb{R}[\underline{x}]$ (in commuting variables) the *Newton polytope* $N(f)$ consists of all integer lattice points in the convex hull of the degrees $\underline{d} = (d_1, \dots, d_n)$ of words appearing in f, considered as vectors in $\mathbb{R}^n$ (see e.g. [Rez78] for details). That is, for $f = \sum_{\underline{d}} a_{\underline{d}} \underline{x}^{\underline{d}} \in \mathbb{R}[\underline{x}]$,

$$N(f) := \mathbb{Z}^n \cap \operatorname{conv}\big(\{\underline{d} \in \mathbb{Z}^n \mid a_{\underline{d}} \neq 0\}\big).$$

We will also refer to the set

$$\frac{1}{2} N(f) := \{\underline{d} \in \mathbb{Z}^n \mid 2\underline{d} \in N(f)\}.$$

Similarly, $N(S)$ and $\frac{1}{2}N(S)$ are defined, where S is a set of words in commuting variables.

Lemma 3.8. *Let $f \in \mathbb{R}\langle \underline{X} \rangle$ be an nc polynomial and W be the vector constructed in Proposition 3.7. Then*

$$\operatorname{cc}(W) = \big\{\underline{x}^{\underline{d}} \mid \underline{d} \in \frac{1}{2} N(\operatorname{cc}([f]))\big\}.$$

Proof. Suppose first that $\underline{d} \in \frac{1}{2}N(\operatorname{cc}([f]))$. We have to prove that $\underline{x}^{\underline{d}} \in \operatorname{cc}(W)$. Recall that $\operatorname{cc}(W) = \{\operatorname{cc}(w) \in [\underline{x}] \mid w \text{ satisfies } (15)\}$. By Lemma 3.6 and since $\deg_{\underline{\alpha}}(w) = \deg_{\underline{\alpha}}(\operatorname{cc}(w))$, we need to show that $2\deg_{\underline{\alpha}}(\underline{x}^{\underline{d}}) \leq \operatorname{cdeg}_{\underline{\alpha}}(f)$ for all $\underline{\alpha} \in \mathbb{R}^n$. Since $2\underline{d} = \sum_{w \in \operatorname{cc}([f])} \lambda_w \underline{d}_w$ for $\underline{d}_w \in N(\operatorname{cc}([f]))$, where $\lambda_w \geq 0$ and $\sum_{w \in \operatorname{cc}([f])} \lambda_w = 1$, it follows that

$$\begin{aligned} 2\deg_{\underline{\alpha}}(\underline{x}^{\underline{d}}) &= \langle \underline{\alpha},\, 2\underline{d} \rangle = \sum_{w \in \operatorname{cc}([f])} \lambda_w \langle \underline{\alpha},\, \underline{d}_w \rangle = \sum_{w \in \operatorname{cc}([f])} \lambda_w \deg_{\underline{\alpha}} w \\ &\leq \sum_{w \in \operatorname{cc}([f])} \lambda_w \deg_{\underline{\alpha}}([f]) = \deg_{\underline{\alpha}}([f]) = \operatorname{cdeg}_{\underline{\alpha}}(f). \end{aligned}$$

To prove the converse implication suppose that $2\underline{d}_0 \in \mathbb{N}_0^n$ and $2\underline{d}_0 \notin N(\operatorname{cc}([f]))$. By the Hahn-Banach separation theorem there exists a separation vector $\underline{\alpha}_0 \in \mathbb{R}^n$ such that $\langle \underline{\alpha}_0,\, 2\underline{d}_0 \rangle > \langle \underline{\alpha}_0,\, \underline{d} \rangle$ for all $\underline{d} \in N(\operatorname{cc}([f]))$. This implies in particular that $\langle \underline{\alpha}_0,\, 2\underline{d}_0 \rangle > \langle \underline{\alpha}_0,\, \underline{d}_w \rangle$ for all $w \in \operatorname{cc}([f])$, hence $2\deg_{\underline{\alpha}_0}(x^{\underline{d}_0}) > \operatorname{cdeg}_{\underline{\alpha}_0}(f)$ and $x^{\underline{d}_0} \notin \operatorname{cc}(W)$. ■

Example 3.9. Let $f = 1 + XY - YX + 2X^2 - 4Y^5 \in \mathbb{R}\langle X, Y \rangle$. Then $[f] = 1 + 2X^2 - 4Y^5$,

$$\operatorname{cc}(f) = \{1, x^2, xy, y^5\} \subseteq [x, y], \quad \operatorname{cc}([f]) = \{1, x^2, y^5\} \subseteq [x, y],$$

$$\begin{aligned} N(\operatorname{cc}([f])) &= \mathbb{Z}^2 \cap \operatorname{conv}\big(\{(0,0),\, (2,0),\, (0,5)\}\big) \\ &= \big\{(0,0),\, (0,1),\, (0,2),\, (0,3),\, (0,4),\, (0,5),\, (1,0),\, (1,1),\, (1,2),\, (2,0)\big\}. \end{aligned}$$

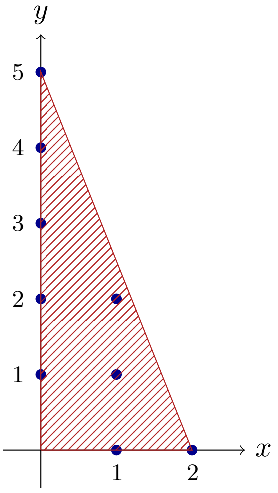

FIGURE 1. The Newton polytope of $f = 1 + XY - YX + 2X^2 - 4Y^5$

We note

$$\frac{1}{2}N(\mathrm{cc}([f])) = \big\{(0,0),\ (0,1),\ (0,2),\ (1,0)\big\}.$$

The reader will easily verify that $W = \begin{bmatrix}1 & Y & Y^2 & X\end{bmatrix}^t$ and hence

$$\mathrm{cc}(W) = \{\underline{x}^{\underline{d}} \mid \underline{d} \in \frac{1}{2}N(\mathrm{cc}([f]))\}.$$

3.3. **Sums of hermitian squares and commutators and semidefinite programming.** In this subsection we present a *conceptual algorithm* based on semidefinite programming for checking whether an nc polynomial of degree $\leq 2d$ is cyclically equivalent to a sum of hermitian squares. Following Proposition 3.7 we must determine whether there exists a positive semidefinite matrix G such that $f \overset{\mathrm{cyc}}{\sim} W^*GW$. This is a semidefinite feasibility problem in the matrix variable G, where the constraints $\langle A_i, G\rangle = b_i$ are essentially equations (1). Note that since $w^* \overset{\mathrm{cyc}}{\not\sim} w$ in general, these constraints (i.e., the matrices A_i) need not be symmetric, as we can see from the following example.

Example 3.10. Let

$$\begin{array}{rcl} f & = & 2XY^2XYX + 4XYX^2YX + XY^4X + 2YXY^2X^2 \\ & = & (Y^2X + 2XYX)^*(Y^2X + 2XYX) - 2XYXY^2X + 2YXY^2X^2 \\ & \overset{\mathrm{cyc}}{\sim} & (Y^2X + 2XYX)^*(Y^2X + 2XYX). \end{array}$$

If we take $W = \begin{bmatrix}XYX & Y^2X\end{bmatrix}^t$, then a tracial Gram matrix G for f is, e.g., obtained as a solution of the following semidefinite program (SDP):

$$\begin{array}{rrcl} \inf & \langle C, G\rangle & & \\ \text{s. t.} & & & \\ XYX^2YX: & G_{1,1} & = & 4 \\ XYXY^2X: & G_{1,2} & = & 2 \\ XY^2XYX: & G_{2,1} & = & 2 \\ XY^4X: & G_{2,2} & = & 1 \\ & G & \succeq & 0. \end{array}$$

Remark 3.11. The matrix C in Example 3.10 is arbitrary. One can use $C = I$, a commonly used heuristic for matrix rank minimization [RFP10]. Often, however, a solution of *high-rank* is desired (cf. Section 4). Then $C = 0$ is used, since under a strict feasibility assumption the interior point methods yield solutions in the relative interior of the optimal face, which is in our case the whole feasibility set. If strict complementarity is additionally provided, the interior point methods lead to the analytic center of the feasibility set [HdKR02]. Even though these assumptions do not always hold for the instances of SDPs we construct, in our computational experiments the choice $C = 0$ in the objective function almost always gave a solution of higher rank than the choice $C = I$.

Remark 3.12. As we restrict our attention to nc polynomials which are cyclically equivalent to symmetric nc polynomials (the others are clearly not in Θ^2), we may always merge the equations corresponding to a particular word and its involution, e.g. in Example 3.10 we can replace the second and the third equation with a single constraint $G_{1,2} + G_{2,1} = 4$.

We formalize the lesson from Remark 3.12 as follows:

Lemma 3.13. *If $f = \sum_w a_w w \in \Theta^2$, then for every $v \in \langle \underline{X} \rangle$*

$$\sum_{w \overset{\text{cyc}}{\sim} v} a_w = \sum_{w \overset{\text{cyc}}{\sim} v^*} a_w. \tag{16}$$

Proof. Using Proposition 3.7 we have

$$\sum_{w \overset{\text{cyc}}{\sim} v} a_w = \sum_{\substack{p,q \in W \\ p^*q \overset{\text{cyc}}{\sim} v}} G_{p,q} = \sum_{\substack{p,q \in W \\ p^*q \overset{\text{cyc}}{\sim} v}} G_{q,p} = \sum_{\substack{p,q \in W \\ q^*p \overset{\text{cyc}}{\sim} v}} G_{p,q} = \sum_{w \overset{\text{cyc}}{\sim} v^*} a_w. \qquad \blacksquare$$

Corollary 3.14. *Given $f \in \mathbb{R}\langle \underline{X} \rangle$ we have:*

(1) *If f does not satisfy (16), then $f \notin \Theta^2$.*
(2) *If f satisfies (16), then we can determine whether $f \in \Theta^2$ by solving the following SDP with only symmetric constraints:*

$$\begin{array}{lrcl} \inf & \langle C, G \rangle & & \\ \text{s.t.} & \displaystyle\sum_{\substack{p,q,\ p^*q \overset{\text{cyc}}{\sim} v \\ \vee\ p^*q \overset{\text{cyc}}{\sim} v^*}} G_{p,q} & = & \displaystyle\sum_{w \overset{\text{cyc}}{\sim} v} (a_w + a_{w^*}), \ \forall v \in W \\ & G & \succeq & 0. \end{array} \tag{CSOHS$_{\text{SDP}}$}$$

The constraints in (CSOHS$_{\text{SDP}}$) are $\langle A_v, G \rangle = b_v$, where $b_v = \sum_{w \overset{\text{cyc}}{\sim} v} (a_w + a_{w^*})$ and $A_v = A_{v^*}$ is the symmetric matrix defined by

$$(A_v)_{p,q} = \begin{cases} 2; & \text{if } p^*q \overset{\text{cyc}}{\sim} v \ \&\ p^*q \overset{\text{cyc}}{\sim} v^*, \\ 1; & \text{if } p^*q \overset{\text{cyc}}{\sim} v \ \&\ p^*q \overset{\text{cyc}}{\nsim} v^*, \\ 0; & \text{otherwise.} \end{cases}$$

The conceptual algorithm to determine whether a given nc polynomial is cyclically equivalent to a sum of hermitian squares (the *tracial Gram matrix method*) is now as described in Algorithm 1:

Note that in Step 5 we can take different decompositions. For example we can compute a Cholesky decomposition (which is not unique if G is not positive definite), the eigenvalue decomposition etc.

```
INPUT: $f \in \mathbb{R}\langle\underline{X}\rangle$ with $f = \sum_{w\in\langle\underline{X}\rangle} a_w w$, where $a_w \in \mathbb{R}$.
STEP 1: If $f$ does not satisfy (16), then $f \notin \Theta^2$. Stop.
STEP 2: Construct $W$.
STEP 3: Construct data $A_v, b_v, C$ corresponding to (CSOHS$_{\mathrm{SDP}}$).
STEP 4: Solve (CSOHS$_{\mathrm{SDP}}$) to obtain $G$. If it is not feasible, then $f \notin \Theta^2$.
        Stop.
STEP 5: Compute a decomposition $G = R^t R$.
OUTPUT: Sum of hermitian squares cyclically equivalent to $f$: $f \overset{\mathrm{cyc}}{\sim} \sum_i g_i^* g_i$,
where $g_i$ denotes the $i$-th component of $RW$.
```

Algorithm 1: The tracial Gram matrix method for finding Θ^2-certificates.

We next focus on the implementation of Step 2 of the Gram matrix method. That is, we construct the vector W containing all words from $\langle\underline{X}\rangle$ satisfying (15). This is a (noncommutative) analogue of the Newton polytope method for the commutative case [Rez78]. Indeed, let $f = \sum a_w w \in \mathbb{R}\langle\underline{X}\rangle$ of degree $\leq 2d$ be given and $u \in \langle\underline{X}\rangle$ be a word which is a candidate for inclusion in W. Then the following is true:

$$\begin{array}{rrcl}
 & 2\deg_{\underline{\alpha}} u & \leq & \operatorname{cdeg}_{\underline{\alpha}} f \quad \text{for all } \underline{\alpha} \in \mathbb{R}^n \\
\Leftrightarrow & 2\deg_{\underline{\alpha}} u & \leq & \deg_{\underline{\alpha}}[f] \quad \text{for all } \underline{\alpha} \in \mathbb{R}^n \\
\Leftrightarrow & 2\langle\underline{\alpha}, \underline{d}_u\rangle & \leq & \max_{w\in \operatorname{cc}([f])}\{\langle\underline{\alpha}, \underline{d}_w\rangle\} \quad \text{for all } \underline{\alpha} \in \mathbb{R}^n \\
\Leftrightarrow & 0 & \leq & \inf_{\underline{\alpha}\in\mathbb{R}^n} \max_{w\in \operatorname{cc}([f])}\{\langle\underline{\alpha}, \underline{d}_w - 2\underline{d}_u\rangle\} \\
\Leftrightarrow & 0 & \leq & \inf\{t \mid \langle\underline{\alpha}, \underline{d}_w - 2\underline{d}_u\rangle \ \leq\ t,\ w \in \operatorname{cc}([f]),\ \underline{\alpha} \in \mathbb{R}^n\}.
\end{array}$$

Verifying the last inequality amounts to solving a linear program in $n+1$ variables with $\operatorname{card}(\operatorname{cc}([f]))$ linear inequalities. Solving such linear programs can be done easily for the problems we are interested in (note that due to other limitations we are considering only nc polynomials f with $n + d \leq 50$). If f is an nc polynomial in 2 variables and has 10000 monomials, then we obtain a linear program (LP) in 3 variables with at most 10000 constraints. Nowadays LP solvers solve such problems easily (within a second); see [Mit03] for a comparison of the state-of-the-art LP solvers.

Algorithm 2 below (the *Newton cyclic chip method*) is the implementation of Step 2 of Algorithm 1.

```
INPUT: $f \in \mathbb{R}\langle\underline{X}\rangle$ with $\deg f \leq 2d$, $f = \sum_{w\in\langle\underline{X}\rangle} a_w w$, where $a_w \in \mathbb{R}$.
STEP 1: Let $V_d$ be the vector of all words in $[\underline{x}]$ with degree $\leq d$.
STEP 2: $W := \varnothing$.
STEP 3: For every $v \in V_d$: if $v$ satisfies (15), then
            $W = W \cup \{$all (noncommutative) permutations of $v\}$.
OUTPUT: $W$.
```

Algorithm 2: The Newton cyclic chip method

Remark 3.15. The vector V_d from Step 1 in Algorithm 2 has length $\dim \mathbb{R}[\underline{x}]_d = \binom{n+d}{d}$, hence we need to solve this number of linear programs in Step 3. For each word v feasible for (15) we add at most $d!$ words to W in Step 3. The length of the constructed W is usually much smaller than the number of all words $w \in \langle\underline{X}\rangle$ of degree $\leq d$. On the other hand, it is often much larger than the vector of words obtained by the Newton chip method [KP10] developed for the sum of hermitian squares decomposition.

3.4. **Software implementation.** Implementing the tracial Gram matrix method together with the Newton cyclic chip method should be done carefully due to several potential bottlenecks. Obviously the most expensive part of the Gram matrix method is Step 4 (solving $\mathrm{CSOHS}_{\mathrm{SDP}}$). Its complexity is determined by the order of the matrix variable G and the number of linear equations. Both parameters are strongly related to the vector W from Step 2. Indeed, the order of G is exactly the length $|W|$ and the number of linear equations is at least $\frac{|W|^2}{(d+1)(2d-1)!}$. This follows from the fact that for each product $u^*v,\ u,v \in W$ there are at most $d+1$ pairs u_i, v_i such that $u_i^*v_i = u^*v$ and at most $(2d-1)!$ cyclically equivalent products.

The vector W constructed by the Newton cyclic chip method is in general the best possible and is the default procedure used by `NCcycSos` in our package `NCSOStools` [CKP11]. `NCcycSos` takes an nc polynomial as input and returns the answer if it is a member of Θ^2. It might be time consuming, as we have already pointed out in Remark 3.15. However, if we know in advance that it is enough to consider products u^*v for some V and $u, v \in V (\subseteq W)$, then we can add this V as an input to `NCcycSos` and skip Step 2 in the Gram matrix method.

Remark 3.16. In a special case we can construct a shorter vector W. Namely, if we know that for a representation $f \overset{\mathrm{cyc}}{\sim} g \in \Sigma^2$ we have that $\sum_{w \overset{\mathrm{cyc}}{\sim} v^*v} g_w \neq 0$ for all hermitian squares v^*v appearing in g, then we can construct W by a slight generalization of the Newton chip method from [KP10]. In this case we take the right chips satisfying (15) of all hermitian squares which are cyclically equivalent to words from f instead of all words $w \in \langle \underline{X} \rangle$ satisfying (15). This works e.g. for the BMV polynomials (see Subsection 4.2) but does not work for the following nc polynomial

$$f = 1 - 4XYX + 2X^2 + X^2Y^4X^2 \overset{\mathrm{cyc}}{\sim} 2(XY - X)(YX - X) + (X^2Y^2 - 1)(Y^2X^2 - 1).$$

In fact, the hermitian square $2XY^2X$ cancels with $-X^2Y^2$ and $-Y^2X^2$ and we don't get the necessary words XY and YX in W by applying the Newton chip method.

We point out that in general the semidefinite program ($\mathrm{CSOHS}_{\mathrm{SDP}}$) might have no strictly feasible points. Absence of (primal) strictly feasible points might cause numerical difficulties while solving ($\mathrm{CSOHS}_{\mathrm{SDP}}$). However, as in [KP10], we can enforce strong duality which is crucial for all SDP solvers by setting the matrix C in ($\mathrm{CSOHS}_{\mathrm{SDP}}$) equal to I (actually any full rank matrix will do); see [KP10, Section 4.1] for details. Another source of numerical problems is the infeasibility of ($\mathrm{CSOHS}_{\mathrm{SDP}}$), which is the case when $f \notin \Theta^2$. We point out that SDP solvers which are supported by `NCSOStools` have easily overcome these difficulties on all tested instances.

Our implementation of the Newton cyclic chip method is augmented by an additional test used to further reduce the length of W. Indeed, if $w \in W$ satisfies the following properties:

(a) if $u^*v \overset{\mathrm{cyc}}{\sim} w^*w$ for some $u, v \in W$, then $u = v$ (i.e., any product cyclically equivalent to w^*w is a hermitian square);
(b) neither w^*w nor any other product cyclically equivalent to w^*w appears in f,

then we can delete w from W, and also all u with $u^*u \overset{\mathrm{cyc}}{\sim} w^*w$. This test is implemented in `NCcycSos` and is run before solving ($\mathrm{CSOHS}_{\mathrm{SDP}}$). It amounts to finding (iteratively) all equations of the type $\langle A_w, G \rangle = 0$ with A_w diagonal.

4. Rational sums of hermitian squares and the BMV conjecture

In this section particular emphasis is given to the extraction of *rational* certificates if the input data is rational. We present several examples illustrating our results, e.g. concerning the BMV conjecture from statistical physics (Subsection 4.2).

4.1. Rational sums of hermitian squares.

Consider a feasibility SDP in primal form

$$\begin{array}{rrcl} & G & \succeq & 0 \\ \text{s. t.} & \langle A_i, G\rangle & = & b_i, \quad i=1,\dots,m \end{array} \tag{FSDP}$$

and assume the input data A_i, b_i is rational for $i = 1, \dots, m$. If the problem is feasible, does there exist a *rational* solution? If so, can one use a combination of numerical and symbolic computation to produce one?

Example 4.1. Some caution is necessary, as a feasible SDP of the form (FSDP) need not admit a rational solution. For a concrete example, note that

$$\begin{bmatrix} 2 & x \\ x & 1 \end{bmatrix} \oplus \begin{bmatrix} x & 1 & 0 \\ 1 & x & 1 \\ 0 & 1 & x \end{bmatrix} \succeq 0 \quad \Leftrightarrow \quad x = \sqrt{2}.$$

On the other hand, if (FSDP) admits a feasible *positive definite* solution, then it admits a (positive definite) *rational* solution. More exactly, we have the following:

Theorem 4.2 (Peyrl & Parrilo [PP08])**.** *If an approximate feasible point G_0 for (FSDP) satisfies*

$$\delta := \min(\operatorname{eig}(G_0)) > \|(\langle A_i, G_0\rangle - b_i)_i\| =: \varepsilon, \tag{17}$$

then a (positive definite) rational feasible point G exists. It can be obtained from G_0 in the following two steps (cf. Figure 2):

(1) *compute a rational approximation $\tilde{G}$ of G_0 with $\tau := \|\tilde{G} - G_0\|$ satisfying $\tau^2 + \varepsilon^2 < \delta^2$;*
(2) *project $\tilde{G}$ onto the affine subspace $\mathcal{L}$ given by the equations $\langle A_i, G\rangle = b_i$ to obtain G.*

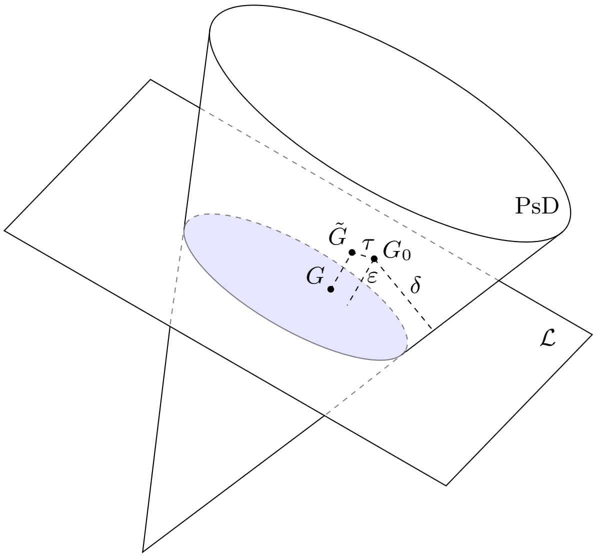

Figure 2. Rounding and projecting to obtain a rational solution

Note that the results in [PP08] are stated for SDPs arising from sum of squares problems, but their results carry over verbatim to the setting of (the seemingly more) general SDPs. The rationalization scheme based on this Peyrl-Parrilo technique has been implemented in `NCSOStools`; see Example 4.4 for a demonstration.

4.2. **BMV conjecture.** In their 2004 paper [LS04], Lieb and Seiringer gave the following purely algebraic reformulation of the Bessis-Moussa-Villani (BMV) conjecture [BMV75] from quantum statistical physics:

Conjecture 4.3. *For all positive semidefinite matrices A and B and all $m \in \mathbb{N}$, the polynomial $p(t) := \operatorname{tr}((A+tB)^m) \in \mathbb{R}[t]$ has only nonnegative coefficients.*

The coefficient of t^k in $p(t)$ for a given m is the trace of $S_{m,k}(A,B)$, where $S_{m,k}(A,B)$ is the sum of all words of length m in the letters A and B in which B appears exactly k times. For example, $S_{4,2}(A,B) = A^2B^2 + ABAB + AB^2A + BABA + B^2A^2 + BA^2B$. $S_{m,k}(X,Y)$ is thus an nc polynomial; it is the sum of all words in two variables X, Y of degree m with $\operatorname{cdeg}_{(0,1)} f = \deg_{(0,1)} f = k$ (and therefore $\operatorname{cdeg}_{(1,0)} f = \deg_{(1,0)} f = m-k$).

In the last few years there has been much activity around the following question: which pairs (m,k) does $S_{m,k}(X^2,Y^2) \in \Theta^2$ or $S_{m,k}(X,Y) \in \Theta^2$ hold for? An affirmative answer (for all m,k) to the former would suffice for the BMV conjecture to hold; this question has been resolved completely (see e.g. [KS08b, CDTA10, CKP10]), however only finitely many nontrivial $S_{m,k}(X^2,Y^2)$ admit a Θ^2-certificate. Adding to the current state of knowledge (nicely summarized in [CDTA10]), we shall use our computer algebra system `NCSOStools` to establish $S_{8,2}(X,Y) \in \Theta^2$, $S_{12,4}(X,Y) \in \Theta^2$, and $S_{14,6}(X,Y) \notin \Theta^2$.

Example 4.4. Consider the nc polynomial $f = S_{8,2}(X,Y)$. To prove that $f \in \Theta^2$ with the aid of `NCSOStools`, proceed as follows:

(1) Define two noncommuting variables:

```
>> NCvars x y
```

(2) Our nc polynomial f is constructed using `BMV(8,2)`. For a numerical test whether $f \in \Theta^2$, run

```
>> params.obj = 0;
>> [IsCycEq,G0,W,sohs,g,SDP_data] = NCcycSos(BMV(8,2), params);
```

This yields a *floating point* Gram matrix G_0

$$G_0 = \begin{bmatrix} 3.9135 & 2.0912 & -0.1590 & 0.9430 \\ 2.0912 & 4.4341 & 1.0570 & -0.1298 \\ -0.1590 & 1.0570 & 4.1435 & 1.9088 \\ 0.9430 & -0.1298 & 1.9088 & 4.0865 \end{bmatrix}$$

for the word vector

$$W = \begin{bmatrix} X^3Y & X^2YX & XYX^2 & YX^3 \end{bmatrix}^t.$$

The rest of the output: `IsCycEq` $= 1$ since f is (numerically) an element of Θ^2; `sohs` is a vector of nc polynomials g_i with $f \overset{\text{cyc}}{\sim} \sum_i g_i^* g_i =$ `g`; `SDP_data` is the SDP data for ($\text{CSOHS}_{\text{SDP}}$) constructed from f.

(3) To round and project the obtained floating point solution `G0`, feed `G0` and `SDP_data` into `RprojRldlt`:

```
>> [G,L,D,P,err]=RprojRldlt(G0,SDP_data,true)
```

This produces a rational Gram matrix `G` for f with respect to W and its LDU decomposition $PLDL^tP^t$, where P is a permutation matrix, L lower unitriangular, and D a diagonal matrix with positive entries. We caution the reader that `L,D`, and `G` are cells, each containing numerators and denominators separately as a matrix. Finally, the obtained rational sum of hermitian squares certificate for $f = S_{8,2}(X,Y)$ is

$$f \overset{\text{cyc}}{\sim} \sum_{i=1}^{4} \lambda_i g_i^* g_i$$

for

$$\begin{aligned} g_1 &= X^3Y + \frac{1}{2}X^2YX + \frac{1}{4}YX^3 \\ g_2 &= X^2YX + \frac{1}{3}XYX^2 - \frac{1}{6}YX^3 \\ g_3 &= XYX^2 + \frac{13}{22}YX^3 \\ g_4 &= YX^3 \end{aligned}$$

and

$$\lambda_1 = 4, \quad \lambda_2 = 3, \quad \lambda_3 = \frac{11}{3}, \quad \lambda_4 = \frac{105}{44}.$$

Not all is lost, however, if the SDP solver gives a *singular* feasible point G_0 for (FSDP). Suppose that z is a *rational* null vector for G_0. Let P be a change of basis matrix containing z as a first column and a (rational) orthogonal basis for the orthogonal complement $\{z\}^\perp$ as its remaining columns. Then

$$P^tG_0P = \begin{bmatrix} 0 & 0 \\ 0 & \hat{G}_0 \end{bmatrix},$$

i.e.,

$$G_0 = P^{-t} \begin{bmatrix} 0 & 0 \\ 0 & \hat{G}_0 \end{bmatrix} P^{-1}$$

for some symmetric $\hat{G}_0$. Hence

$$b_i = \langle A_i, G_0 \rangle = \operatorname{tr}(A_iG_0) = \operatorname{tr}\left(A_iP^{-t} \begin{bmatrix} 0 & 0 \\ 0 & \hat{G}_0 \end{bmatrix} P^{-1}\right) = \operatorname{tr}\left(P^{-1}A_iP^{-t} \begin{bmatrix} 0 & 0 \\ 0 & \hat{G}_0 \end{bmatrix}\right).$$

So if

$$P^{-1}A_iP^{-t} = \begin{bmatrix} a_i & c_i^t \\ c_i & \hat{A}_i \end{bmatrix}$$

then $\hat{A}_i$ is a symmetric matrix with rational entries and

$$b_i = \operatorname{tr}\left(\begin{bmatrix} a_i & c_i^t \\ c_i & \hat{A}_i \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & \hat{G}_0 \end{bmatrix}\right) = \operatorname{tr}(\hat{A}_i\hat{G}_0) = \langle \hat{A}_i, \hat{G}_0 \rangle.$$

We have established a variant of the facial reduction [BW81] which applies whenever the original SDP is given by rational data and has a singular feasible point with a rational null vector:

Theorem 4.5. *Let* (FSDP)*,* G_0 *and* $\hat{A}_i$ *be as above. Consider the feasibility SDP*

$$\begin{array}{rcl} \hat{G} & \succeq & 0 \\ \text{s. t.} \quad \langle \hat{A}_i, \hat{G} \rangle & = & b_i, \quad i = 1, \ldots, m \end{array} \tag{FSDP'}$$

(1) (FSDP') *is feasible if and only if* (FSDP) *is feasible.*

(2) (FSDP') *admits a rational solution if and only if* (FSDP) *does.*

Let us demonstrate this procedure:

Example 4.6. Consider $f = S_{12,4}(X,Y)$. To prove that $f \in \Theta^2$ with the aid of NCSOStools, proceed as follows:

(1) Define two noncommuting variables:

```
>> NCvars x y
```

(2) Our nc polynomial f is constructed using `BMV(12,4)`. For a numerical test whether $f \in \Theta^2$, run

```
>> [IsCycEq,G0,W,sohs,g,SDP_data] = NCcycSos(BMV(12,4));
```

This yields a floating point Gram matrix G_0 that is *singular*.

(3) Try to round and project the obtained floating point solution `G0`, feed `G0` and `SDP_data` into `RprojRldlt`:

```
>> [G,L,D,P,err]=RprojRldlt(G0,SDP_data)
```

This exits with an error, since unlike in Example 4.4, the rounding and projecting alone does not yield a rational feasible point.

(4) Instead, let us reexamine G_0. A quick view at the matrix reveals its first and second column coincide. Likewise the last two columns are the same. We thus run our interactive procedure which aids the computer in reducing the size of the SDP as in Theorem 4.5.

```
>> [G,SDP_data]=fac_reduct(BMV(12,4))
```

This leads the computer to return a floating point feasible point $G_0 \in \mathbb{R}^{15\times 15}$ and the data for this SDP, `SDP_data`. It also stays in interactive mode and the user can inspect the matrix and enter the null vector z to be used in the dimension reduction. In fact, as the first two and the last two columns of G_0 are the same, we feed in two null vectors (as a matrix of two columns):

```
K>> z=[1 0;-1 0;0 0;0 0;0 0;0 0;0 0;0 0;0 0;0 0;0 0;0 0;0 0;0 1;0 -1];
    return
```

Inside the interactive routine this enables the computer to produce a positive definite feasible $\hat{G}_0 \in \mathbb{R}^{13\times 13}$. Hence we exit the interactive routine.

```
K>> stop=1;return
```

Now, NCSOStools uses $\hat{G}_0$ to produce a rational positive semidefinite Gram matrix G for f, which proves $f = S_{12,4}(X,Y) \in \Theta^2$. Like in the previous example, the solution G is a cell containing two matrices with numerators and denominators of the rational entries of G. The reader can verify that $f = W^*GW$ exactly by doing rational arithmetic or approximately by computing floating point approximation for G and using floating point arithmetic.

Example 4.7. We conclude this presentation by showing $S_{14,6}(X,Y) \notin \Theta^2$. We define two noncommuting variables and run `NCcycSos` as in the previous examples:

```
>> NCvars x y
>> [IsCycEq,G0,V,sohs,g,SDP_data] = NCcycSos(BMV(14,6));
```

However, this seems to be an infeasible problem. In fact, we shall use the generated data `SDP_data` to prove it is strongly infeasible by computing a rational hyperplane separating Θ^2 and $S_{14,6}(X,Y)$. Let $\mathcal{P}$ be the set of all nc polynomials p with $\deg_{(1,0)} p = \operatorname{mindeg}_{(1,0)} p = 8$

and $\deg_{(0,1)} p = \operatorname{mindeg}_{(0,1)} p = 6$. Obviously, $S_{14,6}(X,Y) \in \mathcal{P}$. Each $p \in \mathcal{P}$ can be represented by a 35×35 Gram matrix using the basis V from our SDP. An important observation is that $p \in \Theta^2$ if and only if there is a positive semidefinite G satisfying $p \overset{\text{cyc}}{\sim} V^*GV$, cf. [KS08b, Section 3] or [CKP10, Section 2.2].

Let $L : \mathcal{P} \to \mathbb{R}$ be a linear $*$-map nonnegative on $\Theta^2 \cap \mathcal{P}$. It can be represented as $p \mapsto \langle M, G_p \rangle$ for a symmetric 35×35 matrix M, where G_p is a Gram matrix for p. Since $L(\Sigma^2) \subseteq [0,\infty)$, the matrix M is positive semidefinite. The fact that $L(f) = 0$ for all $f \overset{\text{cyc}}{\sim} 0$, can be modeled with constraints $\langle M, H \rangle = 0$ for all $H \in A^\perp$, cf. [CKP10, Section 2.2]. Here, $A^\perp$ is the orthogonal complement of the span of the A_v from Section 3.3 in the set of symmetric matrices. Clearly, it suffices to consider H from a linearly independent generating subset $\mathcal{C}$ of $A^\perp$.

To express $L(S_{14,6}(X,Y)) < 0$, we first compute a Gram matrix for $S_{14,6}(X,Y)$. The matrix $A =$ `SDP_data.A` and vector $b =$ `SDP_data.b` model the linear constraints $\langle A_v, G \rangle = b_v$ for $v \in \langle X, Y \rangle$ with $\deg_{(1,0)} v = 8, \deg_{(0,1)} v = 6$. Hence a symmetrized solution of the linear system

```
>> SDP_data.A\SDP_data.b
```

will be a Gram matrix G for $S_{14,6}(X,Y)$. Now consider the feasibility SDP

$$\begin{array}{rcl} M & \succeq & 0 \\ \text{s.t. } \langle M, G \rangle & = & -35, \quad \forall H \in \mathcal{C} : \langle M, H \rangle \ = \ 0. \end{array}$$

(Here, -35 is just a convenient scaling factor.) Every feasible point induces a hyperplane separating Θ^2 and $S_{14,6}(X,Y)$. Solving this SDP with SeDuMi (using the trivial objective function $C = 0$) yields a floating point solution `M0` in the relative interior of the optimal face, see Remark 3.11, with minimal eigenvalue $\delta = 0.3426$ and residual norm $\varepsilon = 6.8 \cdot 10^{-9}$. Thus we can find a rational feasible solution `M` as explained in Theorem 4.2, using `RprojRldlt`. This proves $S_{14,6}(X,Y) \notin \Theta^2$.

5. Conclusions

In this paper we considered polynomials in noncommuting variables which can be decomposed as a sum of hermitian squares and commutators. We presented a systematic way of finding such a decomposition using our open source computer algebra system `NCSOStools`, freely available at `http://ncsostools.fis.unm.si/`.

The main part of the method – a variant of the classical Gram matrix method – is given by the construction of a semidefinite program. Its solution (if it exists) yields a numerical certificate for the decomposition. The presented Newton cyclic chip method is used to reduce the size of the underlying semidefinite program. Moreover, we also apply an algorithm which under a strict feasibility assumption theoretically and practically yields an exact rational certificate if the input is rational. Finally, in the absence of strict feasibility, a variant of the facial reduction is proposed to reduce the size of the semidefinite program and enforce the existence of Slater points. These results are illustrated by numerous examples also providing demonstrations of how to use the proposed algorithm with our computer algebra system `NCSOStools`.

References

[BCKP] S. Burgdorf, K. Cafuta, I. Klep, and J. Povh. The tracial moment problem and trace-optimization of polynomials. http://www.optimization-online.org/DB_HTML/2010/04/2595.html 2, 5

[BMV75] D. Bessis, P. Moussa, and M. Villani. Monotonic converging variational approximations to the functional integrals in quantum statistical mechanics. *J. Mathematical Phys.*, 16(11):2318–2325, 1975. 2, 14

[BW81] J.M. Borwein and H. Wolkowicz. Facial reduction for a cone-convex programming problem. *J. Austral. Math. Soc. Ser. A*, 30(3):369–380, 1980/81. 2, 15

[CDTA10] B. Collins, K.J. Dykema, and F. Torres-Ayala. Sum-of-squares results for polynomials related to the Bessis-Moussa-Villani conjecture. *J. Stat. Phys.*, 139(5):779–799, 2010. 14

[CKP10] K. Cafuta, I. Klep, and J. Povh. A note on the nonexistence of sum of squares certificates for the Bessis-Moussa-Villani conjecture. *J. Math. Phys.*, 51(8):083521, 10, 2010. 14, 17

[CKP11] K. Cafuta, I. Klep, and J. Povh. NCSOStools: a computer algebra system for symbolic and numerical computation with noncommutative polynomials. *Optim. Methods and Softw.*, 26(3):363–380, 2011. http://ncsostools.fis.unm.si/ 2, 12

[CLR95] M.D. Choi, T.Y. Lam, and B. Reznick. Sums of squares of real polynomials. In B. Jacob and A. Rosenberg, editors, *K-theory and algebraic geometry: connections with quadratic forms and division algebras (Santa Barbara, CA, 1992)*, volume 58 of *Proc. Sympos. Pure Math.*, pages 103–126. Amer. Math. Soc., Providence, RI, 1995. 5

[Con76] A. Connes. Classification of injective factors. Cases II_1, II_∞, III_λ, $\lambda \neq 1$. *Ann. of Math. (2)*, 104:73–115, 1976. 2

[dOHMP08] M.C. de Oliveira, J.W. Helton, S. McCullough, and M. Putinar. Engineering systems and free semi-algebraic geometry. In M. Putinar and S. Sullivant, editors, *Emerging applications of algebraic geometry*, volume 149 of *IMA Vol. Math. Appl.*, pages 17–61. Springer, New York, 2008. 1

[HdKR02] M. Halická, E. de Klerk, and C. Roos. On the convergence of the central path in semidefinite optimization. *SIAM J. Optim.*, 12(4):1090–1099, 2002. 10

[Hel02] J.W. Helton. “Positive” noncommutative polynomials are sums of squares. *Ann. of Math. (2)*, 156(2):675–694, 2002. 1, 2, 5

[KP10] I. Klep and J. Povh. Semidefinite programming and sums of hermitian squares of noncommutative polynomials. *J. Pure Appl. Algebra*, 214:740–749, 2010. 2, 5, 12

[KS08a] I. Klep and M. Schweighofer. Connes’ embedding conjecture and sums of Hermitian squares. *Adv. Math.*, 217(4):1816–1837, 2008. 2, 3, 5

[KS08b] I. Klep and M. Schweighofer. Sums of Hermitian squares and the BMV conjecture. *J. Stat. Phys*, 133(4):739–760, 2008. 5, 7, 14, 17

[Las01] J. B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM J. Optim.*, 11(3):796–817, 2000/01. 1

[Las09] J. B. Lassere. *Moments, positive polynomials and their applications*, volume 1 of *Imperial College Press Optimization Series*. Imperial College Press, London, 2009. 1

[LS04] E.H. Lieb and R. Seiringer. Equivalent forms of the Bessis-Moussa-Villani conjecture. *J. Stat. Phys.*, 115(1-2):185–190, 2004. 14

[McC01] S. McCullough. Factorization of operator-valued polynomials in several non-commuting variables. *Linear Algebra Appl.*, 326(1-3):193–203, 2001. 2

[Mit03] D. Mittelmann. An independent benchmarking of SDP and SOCP solvers. *Math. Program. B*, 95:407–430, 2003. http://plato.asu.edu/bench.html 11

[MP05] S. McCullough and M. Putinar. Noncommutative sums of squares. *Pacific J. Math.*, 218(1):167–171, 2005. 2

[Par03] P.A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Math. Program.*, 96(2, Ser. B):293–320, 2003. 1, 5

[PNA10] S. Pironio, M. Navascués, and A. Acín. Convergent relaxations of polynomial optimization problems with noncommuting variables. *SIAM J. Optim.*, 20(5):2157–2180, 2010. 1

[PP08] H. Peyrl and P.A. Parrilo. Computing sum of squares decompositions with rational coefficients. *Theoret. Comput. Sci.*, 409(2):269–281, 2008. 2, 13, 14

[PS03] P.A. Parrilo and B. Sturmfels. Minimizing polynomial functions. In *Algorithmic and quantitative real algebraic geometry (Piscataway, NJ, 2001)*, volume 60 of *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.*, pages 83–99. Amer. Math. Soc., Providence, RI, 2003. 1

[Rez78] B. Reznick. Extremal PSD forms with few terms. *Duke Math. J.*, 45(2):363–374, 1978. 2, 8, 11

[RFP10] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Rev.*, 52(3):471–501, 2010. 10

[Sch05] M. Schweighofer. Optimization of polynomials on compact semialgebraic sets. *SIAM J. Optim.*, 15(3):805–825, 2005. 1

[Sta] H.R. Stahl. Proof of the BMV conjecture. http://arxiv.org/abs/1107.4875 2

[Tod01] M. J. Todd. Semidefinite optimization. *Acta Numer.*, 10:515–560, 2001. 6

[VB96] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Rev.*, 38(1):49–95, 1996. 6

[WSV00] H. Wolkowicz, R. Saigal, and L. Vandenberghe. *Handbook of semidefinite programming.* International Series in Operations Research & Management Science, 27. Kluwer Academic Publishers, Boston, MA, 2000. Theory, algorithms, and applications. 6

Sabine Burgdorf, Universität Konstanz, Fachbereich Mathematik und Statistik, 78457 Konstanz, Germany

E-mail address: sabine.burgdorf@uni-konstanz.de

Kristijan Cafuta, Univerza v Ljubljani, Fakulteta za elektrotehniko, Laboratorij za uporabno matematiko, Tržaška 25, 1000 Ljubljana, Slovenia

E-mail address: kristijan.cafuta@fe.uni-lj.si

Igor Klep, Univerza v Mariboru, Fakulteta za naravoslovje in matematiko, Koroška 160, 2000 Maribor, and Univerza v Ljubljani, Fakulteta za matematiko in fiziko, Jadranska 19, 1111 Ljubljana, Slovenia

E-mail address: igor.klep@fmf.uni-lj.si

Janez Povh, Fakulteta za informacijske študije v Novem mestu, Novi trg 5, 8000 Novo mesto, Slovenia

E-mail address: janez.povh@fis.unm.si

NOT FOR PUBLICATION

Contents

1. Introduction 1
1.1. Motivation 1
1.2. Related work and contribution 2
2. Preliminaries 3
2.1. Words, nc polynomials and involution 3
2.2. Sum of hermitian squares and commutators 3
3. Implementation and computational algorithms 5
3.1. The tracial Gram matrix method 5
3.2. The Newton cyclic chip method 6
3.3. Sums of hermitian squares and commutators and semidefinite programming 9
3.4. Software implementation 12
4. Rational sums of hermitian squares and the BMV conjecture 13
4.1. Rational sums of hermitian squares 13
4.2. BMV conjecture 14
5. Conclusions 17
References 18
Index 20