Selecting a Monomial Basis for Sums of Squares Programming over a Quotient Ring

Frank Permenter[1] and Pablo A. Parrilo[2]

***Abstract*— In this paper we describe a method for choosing a "good" monomial basis for a sums of squares (SOS) program formulated over a quotient ring. It is known that the monomial basis need only include standard monomials with respect to a Groebner basis. We show that in many cases it is possible to use a reduced subset of standard monomials by combining Groebner basis techniques with the well-known Newton polytope reduction. This reduced subset of standard monomials yields a smaller semidefinite program for obtaining a certificate of non-negativity of a polynomial on an algebraic variety.**

I. INTRODUCTION

Many practical engineering problems require demonstrating non-negativity of a multivariate polynomial over an algebraic variety, i.e. over the solution set of polynomial equations. This problem arises, for example, in local Lyapunov analysis of a polynomial dynamical system.

Unfortunately, certifying non-negativity over a variety is in general a hard computational problem. An alternative is to demonstrate a polynomial is equal to a sum of squares over the variety by solving a *sums of squares program*. A sums of squares program optimizes a linear function of polynomial coefficients subject to constraints that polynomials are sums of squares. If the polynomials in the program are of bounded degree, a sums of squares program is equivalent to a semidefinite program (SDP) and hence efficiently solved [7]. Consider the following sums of squares program, which demonstrates non-negativity of the polynomial $f(\mathbf{x})$ on the set

$$V = \{\mathbf{x} : h_i(\mathbf{x}) = 0, \quad i = 1, \ldots, m\}, \tag{1}$$

where $\mathbf{x}$ is a vector of indeterminates and each h_i is a polynomial in $\mathbb{R}[\mathbf{x}]$:

$$\begin{aligned} &\text{Find } s(\mathbf{x}) \text{ and } \lambda_i(\mathbf{x}) \in \mathbb{R}[\mathbf{x}] \\ &\text{subject to} \\ &s(\mathbf{x}) \text{ is a sum of squares} \\ &s(\mathbf{x}) - f(\mathbf{x}) = \sum_i^m \lambda_i(\mathbf{x}) h_i(\mathbf{x}). \end{aligned} \tag{2}$$

Feasibility of (2) is sufficient to conclude non-negativity of $f(\mathbf{x})$ on V. To see this, note that feasibility implies $f(\mathbf{x})$ and a sum of squares polynomial differ by an expression that vanishes everywhere on V. If one specifies a monomial basis for $\lambda_i(\mathbf{x})$ and $s(\mathbf{x})$, this feasibility problem is equivalent to an SDP. This follows because the sum of squares constraint is equivalent to a semidefinite constraint on the coefficients of $s(\mathbf{x})$, and the equality constraint is equivalent to linear equations on the coefficients of $s(\mathbf{x})$ and $\lambda_i(\mathbf{x})$.

Naturally, the complexity of this sums of squares program grows with the complexity of the underlying variety, as does the size of the corresponding SDP. It is therefore natural to explore how algebraic structure can be exploited to simplify this sums of squares program. Consider the following reformulation [10], which is feasible if and only if (2) is feasible:

$$\begin{aligned} &\text{Find } s(\mathbf{x}) \\ &\text{subject to} \\ &s(\mathbf{x}) \text{ is a sum of squares} \\ &\overline{s(\mathbf{x})} = \overline{f(\mathbf{x})}. \end{aligned} \tag{3}$$

Here, $\overline{s(\mathbf{x})}$ denotes the *normal form* of $s(\mathbf{x})$ with respect to a Groebner basis for the ideal $I = \langle h_1, h_2, \ldots, h_m \rangle$. Two polynomials have the same normal form if and only if they share an equivalence class in the *quotient ring* $\mathbb{R}[\mathbf{x}]/I$. In other words, two polynomials have the same normal form if and only if they differ by a polynomial of the form $\sum_i^m \lambda_i(\mathbf{x}) h_i(\mathbf{x})$. Note if one specifies a vector of monomials $\vec{m}(\mathbf{x})$ one can solve (3) with the SDP:

$$\begin{aligned} &\text{Find } Q \in \mathbb{S}^n \\ &\text{subject to} \\ &\overline{\vec{m}(\mathbf{x})^T Q \vec{m}(\mathbf{x})} = \overline{f(\mathbf{x})} \\ &Q \succeq 0. \end{aligned} \tag{4}$$

This is an SDP since the constraint matching normal forms is *linear* in the entries of Q.

Formulation (3) has two practical advantages over (2). First, the polynomials $\lambda_i(\mathbf{x})$ have been eliminated from the search. Second, one can build $\vec{m}(\mathbf{x})$ in the SDP formulation (4) using only *standard monomials*, which are defined with respect to a Groebner basis as all monomials not divisible by an *initial term* of a polynomial in the Groebner basis. Here, initial term means the unique maximal term of a polynomial with respect to a *term ordering*. Constructing $\vec{m}(\mathbf{x})$ from standard monomials is justified by the fact that any sum of squares polynomial is congruent modulo I to a sum of squares of polynomials supported by standard monomials (see, for example, Lemma 2 of Section III).

Systematic procedures for selecting $\vec{m}(\mathbf{x})$, however, have not been thoroughly addressed. If we consider a total degree

[1]F. Permenter is with the Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology, Cambridge, MA 02139. fperment@mit.edu

[2]P.A. Parrilo is with the Laboratory For Information and Decision Systems (LIDS), Massachusetts Institute of Technology, Cambridge, MA 02139. parrilo@mit.edu

ordering $\geq_\alpha$, a term bound $\mathbf{x}^\gamma$, and define $in_\alpha(s(\mathbf{x}))$ to be the maximal term of $s(\mathbf{x})$ with respect to $\geq_\alpha$, an "optimal" monomial vector is one of minimum dimension solving the following problem:

Problem 1: Given a total degree ordering $\geq_\alpha$ and term bound $\mathbf{x}^\gamma$, find a vector of monomials $\vec{m}(\mathbf{x})$ such that the semidefinite program (4) is feasible whenever the sums of squares program (3) is feasible for some $s(\mathbf{x})$ satisfying $in_\alpha(s(\mathbf{x})) \leq_\alpha \mathbf{x}^\gamma$.

Note the total degree ordering ensures the set of all polynomials satisfying the term bound $\mathbf{x}^\gamma$ can be expressed with a finite set of monomials, which in turn implies an $\vec{m}(\mathbf{x})$ of finite dimension exists that solves Problem 1. In particular, one can construct an $\vec{m}(\mathbf{x})$ solving Problem 1 from the set of *Term Bounded Standard Monomials*, which we denote M_γ and define below:

Definition 1: Term Bounded Standard Monomials. For a given $\mathbf{x}^\gamma$ and total degree ordering $\leq_\alpha$, let M_γ be the set of all exponents β such that $\mathbf{x}^\beta$ is a standard monomial and $\mathbf{x}^{2\beta} \leq_\alpha \mathbf{x}^\gamma$.

A monomial vector constructed from all elements of M_γ solves Problem 1 for a general polynomial $f(\mathbf{x})$ and a general ideal I. In this paper, we develop an algorithm for constructing $\vec{m}(\mathbf{x})$ using just a *subset* of M_γ by exploiting problem specific structure. Given a Groebner basis for I with respect to a total degree ordering $\leq_\alpha$, the algorithm we present constructs a subset of M_γ using the structure induced by multi-divisor polynomial division and well known Newton polytope arguments. As we illustrate with examples, this subset is often a strict subset of M_γ which enables one to build smaller SDP formulations for the sums of squares program (3).

A. Prior Work

Many authors have investigated ways of exploiting algebraic in sums of squares programs. General methods are introduced in [10] and [8] that exploit sparsity and symmetry. Other techniques for exploiting sparsity are further discussed in [12], [6], [5]. Symmetry methods are discussed in [3]. Practical implementations of these types of techniques are discussed in [4].

Quotient ring formulations, aside from their introduction in [10] and discussion in [8], have received less attention. This is perhaps due to their reliance on Groebner bases methods, and their lack of support by sums of squares modeling tools. The former concern is not always relevant since Groebner bases are easily calculated (or are immediately available) in many instances. The latter concern, of course, does not take away from the inherent power of quotient ring formulations, which we hope to extend with this paper.

II. BACKGROUND

We begin by reviewing polynomial ideals and Groebner bases. We then discuss important properties of multivariate, multi-divisor polynomial long division and its connections to quotient rings.

A. Polynomial Ideals, Term Orderings, and Groebner Bases

The ideal I generated by a set of polynomials

$$H = \{h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_m(\mathbf{x})\}$$

is denoted

$$I = \langle h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_m(\mathbf{x})\rangle$$

and is equal to the set

$$I = \left\{ \sum_{i=1}^{m} \lambda_i(\mathbf{x}) h_i(\mathbf{x}) : \lambda_i(\mathbf{x}) \in \mathbb{R}[\mathbf{x}] \right\}. \tag{5}$$

In other words, the ideal generated by a set of polynomials H is the set of all polynomials that can be obtained by summing scaled versions of polynomials in H, where the scale factors can be any polynomial in $\mathbb{R}[\mathbf{x}]$.

A term ordering $\geq_\alpha$ is a relation on $\mathbb{Z}_+^n$ (i.e. monomial exponents) satisfying

- The relation $\geq_\alpha$ is a total ordering.
- If $\gamma >_\alpha \beta$ then for any $\zeta \in \mathbb{Z}_+^n$, $\gamma + \zeta >_\alpha \beta + \zeta$.
- The relation $\geq_\alpha$ is a well-ordering, which means every nonempty subset of $\mathbb{Z}_+^n$ has a smallest element.

We say a term ordering is a *total degree ordering* whenever

$$\sum_{i=1}^{n} \gamma_i \geq \sum_{i=1}^{n} \beta_i \Rightarrow \gamma \geq_\alpha \beta.$$

We will abuse notation and write for a monomial that $\mathbf{x}^\gamma \geq_\alpha \mathbf{x}^\beta$ whenever $\gamma \geq_\alpha \beta$.

The initial term $in_\alpha(f(\mathbf{x}))$ of a polynomial is the unique maximal term of $f(\mathbf{x})$ with respect to $\geq_\alpha$. Given a term ordering $\geq_\alpha$, one can consider the set of initial terms of polynomials in I. These initial terms generate an ideal, called the *initial ideal*, which we denote $in_\alpha(I)$.

For a particular term ordering, we say a set of polynomials

$$G = \{g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_n(\mathbf{x})\}$$

is a Groebner basis for an ideal I if and only if it generates I and

$$in_\alpha(I) = \langle in_\alpha(g_1), in_\alpha(g_2), \ldots, in_\alpha(g_n)\rangle. \tag{6}$$

Property (6) implies the initial term of any polynomial in I is divisible by an element of the Groebner basis. This enables a division algorithm equipped with a Groebner basis to decide if an arbitrary polynomial $f(\mathbf{x})$ is a member of the ideal I. We discuss the important properties of this division algorithm in the next section.

B. Division Algorithm and Normal Forms

Given a polynomial $f(\mathbf{x})$, a list of divisors $H = \{h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_m(\mathbf{x})\}$ and a monomial ordering $\geq_\alpha$, a division algorithm can be defined as in [2] that finds $\lambda_i(\mathbf{x})$ and a remainder term $\overline{f(\mathbf{x})}^H$ such that

$$f(\mathbf{x}) = \sum_{i=1}^{m} \lambda_i(\mathbf{x}) h_i(\mathbf{x}) + \overline{f(\mathbf{x})}^H$$

with properties

$$in_\alpha(\overline{f(\mathbf{x})}^H) \text{ is not divisible by any } in_\alpha(h_i(\mathbf{x})) \quad (7)$$

$$in_\alpha(\overline{f(\mathbf{x})}^H) \leq_\alpha in_\alpha(f(\mathbf{x})) \quad (8)$$

$$in_\alpha(\lambda_i(\mathbf{x})h_i(\mathbf{x})) \leq_\alpha in_\alpha(f(\mathbf{x}))$$

If we use a list of divisors G that form a Groebner basis for $I = \langle h_1, h_2, \ldots, h_m \rangle$, the remainder term $\overline{f(\mathbf{x})}^G$ equals $\overline{f(\mathbf{x})}$, the normal form of $f(\mathbf{x})$ with respect to the Groebner basis. The normal form of a polynomial is unique, meaning the list of divisors G can be permuted without changing the remainder term $\overline{f(\mathbf{x})}^G$. Normal forms also obey the following properties:

- Ideal membership

$$f(\mathbf{x}) \in I \Leftrightarrow \overline{f(\mathbf{x})} = 0$$

- Arithmetic identities

$$\overline{g(\mathbf{x}) \cdot f(\mathbf{x})} = \overline{\overline{g(\mathbf{x})} \cdot \overline{f(\mathbf{x})}} \quad (9)$$

$$\overline{g(\mathbf{x}) + f(\mathbf{x})} = \overline{g(\mathbf{x})} + \overline{f(\mathbf{x})} \quad (10)$$

- Congruence modulo I

$$s(\mathbf{x}) \equiv f(\mathbf{x}) \mod I \Leftrightarrow \overline{f(\mathbf{x})} = \overline{s(\mathbf{x})} \quad (11)$$

Note property (7) implies $\overline{f(\mathbf{x})}$ contains only standard monomials, i.e. monomials not divisible by an initial term of a polynomial in the Groebner basis. Property (11) gives the correspondence between normal forms and equivalence classes in $\mathbb{R}[\mathbf{x}]/I$.

III. Approach

We now present an algorithm for computing a vector of standard monomials satisfying Problem 1 given a term bound $\mathbf{x}^\gamma$, a total degree term ordering $\geq_\alpha$, and a corresponding Groebner basis. First, we calculate a set of monomials S_γ consistent with the structure induced by the Groebner basis when it is used as the divisor set in the division algorithm. We then use S_γ to compute a suitable set of monomials appearing in a sum of squares decomposition of $s(\mathbf{x})$ using Newton polytope arguments. Finally, we take normal forms of these monomials to arrive at a reduced set N_γ of standard monomials. We prove N_γ is a subset of the term bounded standard monomials M_γ and that this subset can be used to construct a monomial vector solving Problem 1.

The proposed procedure is explicitly given in Algorithm 1. Here, we define the support $\text{supp}(w(\mathbf{x}))$ of a polynomial $w(\mathbf{x})$ to be the unique set giving

$$w(\mathbf{x}) = \sum_{\beta \in \text{supp}(w(\mathbf{x}))} c_\beta \mathbf{x}^\beta$$

for nonzero real valued coefficients c_β. We claim Algorithm 1 has the following properties:

Property 1: Algorithm 1 returns a set N_γ such that the monomial vector $\vec{m}(\mathbf{x})_i = \mathbf{x}^{\beta_i}, \beta_i \in N_\gamma, i = 1, \ldots, |N_\gamma|$ solves Problem 1.

Property 2: The set N_γ returned by Algorithm 1 is a subset of the Term Bounded Standard Monomials M_γ.

Input: A polynomial $f(\mathbf{x})$, a total degree term ordering $\geq_\alpha$ and corresponding Groebner basis G, a maximal monomial $\mathbf{x}^\gamma$
Output: A set of standard monomial exponents N_γ
Compute monomials in normal form of $f(\mathbf{x})$
$S_\gamma = \text{supp}(\overline{f(\mathbf{x})})$
Monomials that respect term bound
foreach $g(\mathbf{x}) \in G$ **do**
  # Total degree ordering ensures this loop terminates
  forall $\mathbf{x}^\beta \leq_\alpha \mathbf{x}^\gamma$ **do**
    if $in_\alpha(\mathbf{x}^\beta g(\mathbf{x})) \leq_\alpha \mathbf{x}^\gamma$ **then**
      $S_\gamma = S_\gamma \cup \text{supp}(\mathbf{x}^\beta g(\mathbf{x}))$
    end
  end
end
Normal form of monomials in Newton polytope
$N_\gamma = \{\}$
foreach $\beta \in conv(\frac{1}{2}S_\gamma) \cap \mathbb{Z}^n$ **do**
  $N_\gamma = N_\gamma \cup \text{supp}(\overline{\mathbf{x}^\beta})$
end

Algorithm 1: Computes a reduced subset of term bounded standard monomials.

To understand the mechanics of Algorithm 1 and how Properties 1 and 2 can be demonstrated, first note that *any* polynomial $s(\mathbf{x})$ satisfying the conditions

$$in_\alpha(s(\mathbf{x})) \leq_\alpha \mathbf{x}^\gamma \quad (12)$$

$$s(\mathbf{x}) \equiv f(\mathbf{x}) \mod I \quad (13)$$

can be decomposed using a Groebner basis G for I with respect to the term ordering $\leq_\alpha$ as

$$s(\mathbf{x}) = \sum_{i=1}^{|G|} \lambda_i(\mathbf{x}) g_i(\mathbf{x}) + \overline{f(\mathbf{x})}$$

where

$$in_\alpha(\lambda_i(\mathbf{x})g_i(\mathbf{x})) \leq_\alpha in_\alpha(s(\mathbf{x})) \leq_\alpha \mathbf{x}^\gamma,$$

and $g_i \in G$ and $\lambda_i(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$. Existence of this decomposition is implied by properties of division by a Groebner basis and motivates the construction in Algorithm 1 of the set S_γ, which has the obvious property:

$$s(\mathbf{x}) \text{ satisfies (12) and (13)} \Rightarrow \text{supp}(s(\mathbf{x})) \subseteq S_\gamma.$$

Remark 1: The total degree ordering ensures that the set S_γ constructed by Algorithm 1 is *finite*.

The next phase of the algorithm exploits the structure of sums of squares polynomials. If a polynomial with support in S_γ is equal to a sum of squares $\sum_i p_i(\mathbf{x})^2$, then Newton polytope arguments given in [1] describe the support of $p_i(\mathbf{x})$. Specifically,

$$s(\mathbf{x}) = \sum_i p_i(\mathbf{x})^2 \quad \Rightarrow \text{supp}(p_i) \subseteq \text{conv}\left(\frac{1}{2}S_\gamma\right) \cap \mathbb{Z}^n,$$

where $conv$ denotes convex hull. This gives the following immediate result:

Lemma 1: If a sums of squares polynomial $\sum_i p_i(\mathbf{x})^2$ satisfies (12) and (13), then supp$(p_i(\mathbf{x}))$ is contained in conv$(\frac{1}{2}S_\gamma) \cap \mathbb{Z}^n$.

Lemma 1 implies a monomial vector constructed from conv$(\frac{1}{2}S_\gamma) \cap \mathbb{Z}^n$ solves Problem 1. We are interested, however, not in conv$(\frac{1}{2}S_\gamma) \cap \mathbb{Z}^n$ but in a set of standard monomials. The algorithm therefore builds the set N_γ by taking normal forms of monomials constructed from conv$(\frac{1}{2}S_\gamma) \cap \mathbb{Z}^n$. This is motivated by the following lemma:

Lemma 2: If a polynomial $s(\mathbf{x}) = \sum_i p_i(\mathbf{x})^2$ satisfies $s(\mathbf{x}) \equiv f(\mathbf{x}) \mod I$, then there exists a (possibly different) sum of squares polynomial $\hat{s}(\mathbf{x}) = \sum_i \hat{p}_i(\mathbf{x})^2$ satisfying $\hat{s}(\mathbf{x}) \equiv f(\mathbf{x}) \mod I$ where supp$(\hat{p}_i) =$ supp$(\overline{p_i})$.

Proof: See Appendix. ■

Combining Lemma 1, Lemma 2, and (10) demonstrates a monomial vector constructed from N_γ solves Problem 1 and hence demonstrates Property 1.

To demonstrate Property 2, we need the following basic fact relating term orderings and polytopes:

Lemma 3: If $\geq_\alpha$ is a valid term ordering and S_γ is a finite set of exponents, then there is a unique maximal element $\mathbf{x}^\zeta$ of $\{\mathbf{x}^\beta : \beta \in \text{conv}(S_\gamma)\}$ with respect to $\geq_\alpha$, and ζ is an element of S_γ.

Proof: See Appendix. ■

To show that $N_\gamma \subseteq M_\gamma$, we first note that for any $\sigma \in S_\gamma$, $\mathbf{x}^\sigma \leq_\alpha \mathbf{x}^\gamma$ by construction of S_γ. We then use Lemma 3 to conclude for any point β in conv$(\frac{1}{2}S_\gamma) \cap \mathbb{Z}^n$, $\mathbf{x}^{2\beta} \leq_\alpha \mathbf{x}^\gamma$. Finally, we use (8) to conclude that for any ζ in supp$(\overline{\mathbf{x}^\beta})$, $\mathbf{x}^{2\zeta} \leq_\alpha \mathbf{x}^\gamma$. That is, all ζ in N_γ are also in M_γ and the desired result follows.

IV. Examples

We illustrate the results with a few examples.

A. Example 1

Consider proving non-negativity of

$$f(x, y) = -x^3 - xy^2$$

on the variety $V = \{(x, y) : x^3 + y^2 = 0\}$. Take I to be the ideal $\langle x^3 + y^2 \rangle$ and $\geq_\alpha$ to be the graded lex ordering with $x > y$. Since the ideal I has one generator, a Groebner basis for I in any term ordering is just the generator $x^3 + y^2$.

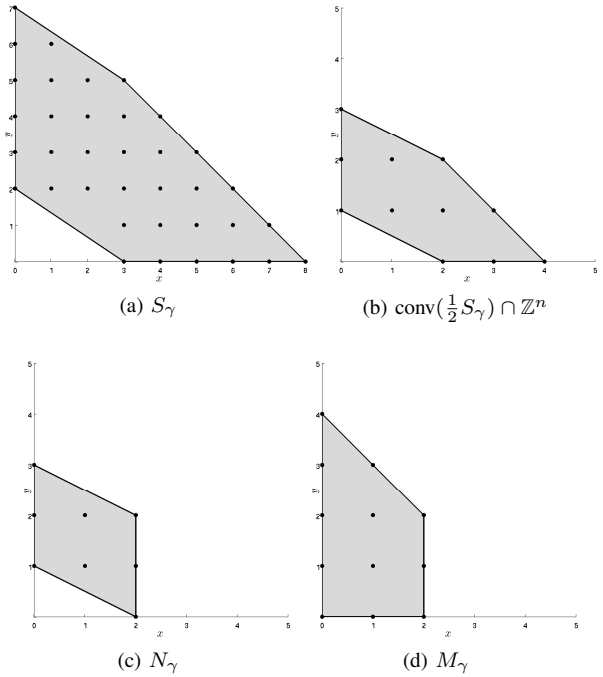
Consider a quotient ring formulation that searches for sum of squares polynomials $s(x, y)$ satisfying $in_\alpha(s(x, y)) \leq_\alpha y^8$. For this choice of term bound, the set of all term bounded standard monomials is given by

$$\{\ 1 \quad x \quad y \quad x^2 \quad xy \quad y^2 \quad x^2y \quad xy^2 \quad y^3 \quad x^2y^2 \quad xy^3 \quad y^4\ \}.$$

The corresponding set of exponents M_γ is shown in Figure 1d.

Applying Algorithm 1, we'll find a subset of exponents from which a smaller set of monomials can be constructed. Decomposing $s(x, y)$ as

$$\begin{aligned} s(x, y) &= \lambda(x, y)(x^3 + y^2) + \overline{-x^3 - xy^2} \\ &= \lambda(x, y)(x^3 + y^2) + y^2 - xy^2, \end{aligned}$$



(a) S_γ (b) conv$(\frac{1}{2}S_\gamma) \cap \mathbb{Z}^n$

(c) N_γ (d) M_γ

Fig. 1: Different sets of monomials arising in Example IV-A. Fig. 1a-1c show monomial sets calculated at different stages of Algorithm 1. Figure 1d shows all term bounded standard monomials. Note N_γ is a strict subset of M_γ, which implies Algorithm 1 leads to a smaller SDP for the given example.

we seek the set of monomials S_γ that can appear in $s(x, y)$ for all $\lambda(x, y)$ such that

$$in_\alpha(\lambda(x, y)(x^3 + y^2)) \leq_\alpha y^8.$$

This set is plotted in Figure 1a. Plotted in Figure 1b are the integer points contained in conv$(\frac{1}{2}(S_\gamma))$. Constructed from these integer points is N_γ, the desired subset of M_γ. Note from Figure 1 that N_γ is a strict subset of M_γ.

B. Example 2

The preceding example dealt with a positive dimensional variety. We now explore a zero-dimensional case, for which the set of standard monomials is always finite. One may perhaps think that as the degree of $s(\mathbf{x})$ in our search increases, all standard monomials must eventually be included and there is no benefit to running the proposed algorithm. This example illustrates this is not necessarily the case.

Consider proving $f(x) = -x^3$ is nonnegative on the variety V

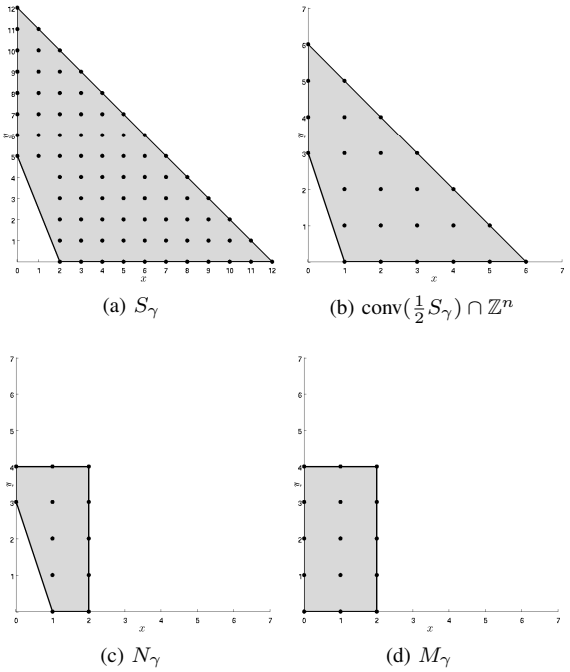
$$V = \{(x, y) : x^3 + y^6 = 0, x^2 - y^5 = 0\}.$$

Let I equal the ideal generated by the defining polynomials of the variety V:

$$I = \langle x^3 + y^6, x^2 - y^5 \rangle.$$

Taking $\geq_\alpha$ to again be the graded lex ordering with $x > y$, a Groebner basis for I is

$$\{x^2 - y^5, x^3 + x^2y\}.$$



(a) S_γ (b) $\text{conv}(\frac{1}{2}S_\gamma) \cap \mathbb{Z}^n$

(c) N_γ (d) M_γ

Fig. 2: Different sets of monomials arising in Example IV-B. Fig. 2a-2c show monomial sets calculated at different stages of Algorithm 1. Figure 1d shows all term bounded standard monomials. Note N_γ is a strict subset of M_γ, which implies Algorithm 1 leads to a smaller SDP for the given zero dimensional example.

We will consider all $s(x, y)$ such that $in_\alpha(s(x, y)) \leq_\alpha x^{12}$. For this term bound, M_γ contains the exponents of all standard monomials. We see in Figure 2 that even in this case, the proposed technique leads to a set of monomials N_γ that is a strict subset of M_γ. This will always happen for the given $f(x)$, even as $in_\alpha(s(x, y))$ is allowed to grow arbitrarily large.

C. Example 3

As a final example, we explore a variety V relevant to stability analysis of a cart-pole, a 4^{th} order mechanical system common in robotics. Here V takes the form

$$V = \{\mathbf{x} : s_\theta^2 + c_\theta^2 - 1 = 0,\ \ddot{x} - g(\mathbf{x}) = 0,\ \dot{W}(\mathbf{x}) = 0\},$$

where

$$\mathbf{x} = (x, \dot{x}, \ddot{x}, \dot{\theta}, c_\theta, s_\theta).$$

Notice here that all variables are treated as independent formal indeterminates, i.e., no algebraic relationship is assumed to hold between x and $\dot{x}$ (and similarly for all other variables).

The first two equations encode a polynomial representation of the cart-pole dynamics. The first equation is a unit circle constraint for trigonometric variables. The second equation encodes an implicit dynamical relationship between an acceleration and the other system variables. The 3rd equation

$\mathbf{x}^\gamma$	$\lvert N_\gamma\rvert$	$\lvert M_\gamma\rvert$	dim $\mathbb{S}_n$ $n = \lvert N_\gamma\rvert$	dim $\mathbb{S}_n$ $n = \lvert M_\gamma\rvert$	% Reduction
x^4	23	27	276	378	27
$\dot{\theta}^6$	32	34	528	595	11
x^6	63	75	2016	2850	29
$\dot{\theta}^8$	81	84	3321	3570	7.0
x^8	151	168	11476	14196	19
$\dot{\theta}^{10}$	176	179	15576	16110	3.3
x^{10}	300	323	45150	52326	14
$\dot{\theta}^{12}$	333	336	55611	56616	1.8
x^{12}	526	556	138601	154846	10

TABLE I: Comparative savings for different term bounds on $s(\mathbf{x})$ for Example IV-C. The $|M_\gamma|$ column shows the number of term bounded standard monomials. The $|N_\gamma|$ column shows the subset of monomials returned by Algorithm 1. Columns also show the dimension of the vector space of symmetric matrices appearing in the corresponding SDPs, a quantity relevant for SDP solvers. The percent reduction in dimension is shown in the last column.

$\dot{W}(\mathbf{x})$ is the time derivative of a quadratic Lyapunov function $W(x, \dot{x}, \dot{\theta}, s_\theta)$. The hypersurface where $\dot{W}(\mathbf{x})$ vanishes defines a verifiable region of stability.

If $\geq_\alpha$ is the graded lex ordering with

$$x > \dot{x} > \ddot{x} > \dot{\theta} > c_\theta > s_\theta,$$

a Groebner basis for I, the ideal generated by the defining polynomials of V, contains 8 polynomials in indeterminates $(x, \dot{x}, \ddot{x}, \dot{\theta}, c_\theta, s_\theta)$ in degrees ranging from 2 to 8 (for a particular choice of Lyapunov function and system parameters).

Local stability can be estimated by finding the largest scalar ρ such that

$$f(\mathbf{x}) = (s_\theta^2 + \dot{\theta}^2 + \dot{x}^2 + x^2)(W - \rho)$$

is congruent to a sum of squares polynomial $s(\mathbf{x})$ modulo I. For the same choice of Lyapunov function and system parameters, the initial monomial of $f(\mathbf{x})$ is x^4. Thus, we consider $s(\mathbf{x})$ with $in_\alpha(s(\mathbf{x}))$ bounded below by x^4. Table I compares the use of all term bounded standard monomials versus the subset returned by the proposed algorithm for various upper bounds on $in_\alpha(s(\mathbf{x}))$. Compared are the number of monomials and the dimension of the corresponding vector space of symmetric matrices $\mathbb{S}_n$. The latter quantity is relevant for SDP solvers such as SeDuMi [11] since it equals the number of scalar decision variables introduced by the solver.

V. CONCLUSIONS AND FUTURE WORK

A. Conclusions

We have proposed a technique for selecting a reduced set of monomials in semidefinite program formulations for sums of squares programs formulated over quotient rings. The proposed method can lead to smaller, better conditioned semidefinite programs for proving non-negativity of polynomials over algebraic varieties.

B. Future Work

For the zero-dimensional case, the set of standard monomials always has finite cardinality. This fact provides the justification for a degree bound in [9] for sums of squares representations of polynomials nonnegative on zero dimensional varieties. Can the proposed method be used to improve this bound?

Another open question addresses the issue of strict containment of N_γ in M_γ. Is it possible to characterize the cases when strict containment holds, and under what conditions it fails? In other words, in what cases is the proposed method guaranteed to reduce the size of the resulting semidefinite programs?

Finally, what role do the different possible term orderings play in producing structure useful for complexity reduction?

APPENDIX

Proof: (of Lemma 2)

Since $s(\mathbf{x}) = \sum_i p_i(\mathbf{x})^2$ is congruent modulo I to $f(\mathbf{x})$ we can write that

$$\overline{f(\mathbf{x})} = \overline{\sum_i p_i(\mathbf{x})^2}.$$

Using properties of normal form arithmetic given by (10) and (9), we see that

$$\begin{aligned}\overline{\sum_i p_i(\mathbf{x})^2} &= \sum_i \overline{p_i(\mathbf{x})^2} \\ &= \sum_i \overline{p_i(\mathbf{x})}\ \overline{p_i(\mathbf{x})} \\ &= \sum_i \hat{p}_i(\mathbf{x})^2 \\ &= \overline{\sum_i \hat{p}_i(\mathbf{x})^2},\end{aligned}$$

where we have taken $\hat{p}_i(\mathbf{x}) = \overline{p_i(\mathbf{x})}$.

Thus, $\hat{s}(\mathbf{x}) = \sum_i \hat{p}_i(\mathbf{x})^2$ is congruent modulo I to $f(\mathbf{x})$ and $\text{supp}(\hat{p}_i) = \text{supp}(\overline{p_i})$. ■

Proof: (of Lemma 3)

To prove this claim, we first recall a fact about valid term orderings: monomials can be compared with respect to the ordering by taking inner products of their exponents with the rows of a suitable *weight matrix*. Suppose we want to show $\mathbf{x}^\beta >_\alpha \mathbf{x}^\sigma$. We first check that $w_1^T\beta \geq w_1^T\sigma$, where w_1^T is the first row of the weight matrix. If $w_1^T\beta = w_1^T\sigma$, we move to the next row of the matrix w_2^T and check if $w_2^T\beta \geq w_2^T\sigma$. This process continues down the rows of the matrix until the inequality is strict. For any valid term ordering, this process terminates in finitely many steps.

From this we note if $\geq_\alpha$ is a valid term ordering, there is a unique maximal element of $\{\mathbf{x}^\beta : \beta \in conv(S_\gamma)\}$ and it can be found by solving a series of linear programs (LPs) over faces of $conv(S_\gamma)$. The cost vector of the i^{th} linear program is w_i and the feasible set of each LP is the face of $conv(S_\gamma)$ on which the $(i-1)^{th}$ LP achieves its optimal solution (the initial feasible set is just $conv(S_\gamma)$). We iterate until an LP has a unique optimum, which occurs only at extreme points of $conv(S_\gamma)$. Since the extreme points are contained in the set S_γ, we must have that the maximal element $\mathbf{x}^\zeta$ satisfies $\zeta \in S_\gamma$. ■

REFERENCES

[1] M. D. Choi, T. Y. Lam, and B. Reznick. Sums of squares of real polynomials. *Proceedings of Symposia in Pure Mathematics*, 58(2):103–126, 1995.

[2] D. A. Cox, J. Little, and D. O'Shea. *Using Algebraic Geometry*. Graduate Texts in Mathematics. Springer-Verlag, Berlin-Heidelberg-New York, March 2005.

[3] K. Gatermann and P. A. Parrilo. Symmetry groups, semidefinite programs, and sums of squares. *Journal of Pure and Applied Algebra*, 192(1–3):95–128, 2004.

[4] J. Löfberg. Pre- and post-processing sum-of-squares programs in practice. *IEEE Transactions on Automatic Control*, 54(5):1007–1011, 2009.

[5] J. Nie. Sum of squares method for sensor network localization. *Computational Optimization and Applications*, 43:151–179, 2009. 10.1007/s10589-007-9131-z.

[6] J. Nie and J. Demmel. Sparse SOS relaxations for minimizing functions that are summations of small polynomials. *SIAM Journal on Optimization*, 19(4):1534–1558, 2008.

[7] P. A. Parrilo. *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. PhD thesis, California Institute of Technology, Pasadena, CA, 2000.

[8] P. A. Parrilo. Exploiting algebraic structure in sum of squares programs. In D. Henrion and A. Garulli, editors, *Positive Polynomials in Control*, volume 312 of *Lecture Notes in Control and Information Sciences*, pages 580–580. Springer Berlin / Heidelberg, 2005. 10.1007/10997703_11.

[9] P.A. Parrilo. An explicit construction of distinguished representations of polynomials nonnegative over finite sets. Technical report, March 2002.

[10] P.A. Parrilo. Exploiting structure in sum of squares programs. In *IEEE Conference on Decision and Control*, Maui, Hawaii, December 2003.

[11] J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12:625–653, 1999.

[12] 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*, 17:218–242, 2006.