



Contents lists available at ScienceDirect

Journal of Symbolic Computation

www.elsevier.com/locate/jsc



Semialgebraic sets and real binary forms decompositions

M. Ansola^a, A. Díaz-Cano^b, M.A. Zurro^c^a Universidad Complutense de Madrid, Madrid, Spain^b IMI and Department of Algebra, Geometry and Topology, Faculty of Mathematics, Universidad Complutense de Madrid, Madrid, Spain^c Department of Mathematics, Faculty of Sciences, Universidad Autónoma de Madrid, Madrid, Spain

ARTICLE INFO

Article history:

Received 8 November 2019

Received in revised form 15 November 2020

Accepted 25 February 2021

Available online 17 March 2021

MSC:

14P10

12D10

Keywords:

Real binary forms

Waring decompositions

Semialgebraic sets

ABSTRACT

The Waring Problem over polynomial rings asks how to decompose a homogeneous polynomial p of degree d as a linear combination of d -th powers of linear forms. In this work we give an algorithm to obtain a real Waring decomposition of any given real binary form p of length at most its degree. In fact, we construct a semialgebraic family of Waring decompositions for p . We illustrate our results with some examples.

© 2021 Elsevier Ltd. All rights reserved.

1. Introduction

In this work we address the problem of effectively decompose a real binary form p of degree d as a linear combination of d -th powers of linear binary forms. This problem is called Waring Problem for real binary forms.

Fixed the degree $d > 1$ and two independent variables x, y over the field of real numbers \mathbb{R} , Waring Problem can be formulated as follows.

E-mail addresses: mansola@ucm.es (M. Ansola), adiazcan@ucm.es (A. Díaz-Cano), mangeses.zurro@uam.es (M.A. Zurro).

Let $p(x, y)$ be a binary form of degree d in $\mathbb{R}[x, y]$. Determine linear forms ℓ_1, \dots, ℓ_m in $\mathbb{R}[x, y]$ such that $p(x, y) = \lambda_1 \ell_1^d + \dots + \lambda_m \ell_m^d$ for some $\lambda_1, \dots, \lambda_m$ real numbers. (1.1)

If all λ_i in $(WD)_m$ are non zero, the number m in $(WD)_m$ is called *the length* of the decomposition. This decomposition of p (not necessarily unique) is known as a *Waring decomposition* (expression that we will abbreviate from now on as WD) of the polynomial. This family of problems, $(WD)_m$ for $m \geq 1$, is solved trivially for $m = d + 1$, with d the degree of p .

Since Alexander and Hirschowitz (1995), many authors have worked on this problem, both in $n > 2$ variables and in the binary case (see Helmke (1992); Comon and Mourrain (1996), Brachat et al. (2010) and the references therein). In Fröberg et al. (2018), the authors proposed a very interesting list of open questions on these issues when the field of coefficients is the field of complex numbers \mathbb{C} . As Causa and Re (2011) or Carlini et al. (2012) affirm, the real case becomes more complicated than the complex case. Ballico and Bernardi (2013) emphasize the importance of the real case for the applications. But, both in the real case and in the complex case, one of the most studied aspects has been the calculation of the length of a Waring decomposition.

Focusing on the binary case, since Comon and Ottaviani (2012), it is known that $(WD)_d$ can be solved for any real binary form p , see also Tokcan (2017) or Brustenga i Moncusí and Masuti (2019). This real binary case has been investigated by different authors; for instance, Boij et al. (2011), Comon and Ottaviani (2012) or Reznick (2010).

To study the Waring Problem $(WD)_d$ we have used some techniques that are based on the real semialgebraic structure of the projective space of real binary forms of fixed degree. Other authors have also used Semialgebraic Geometry to address problems with tensors, see the recent article Comon et al. (2020).

More precisely, we construct a semialgebraic set, whose complement \mathcal{G} , also semialgebraic, provides a parametric description of all Waring decompositions of p . In Section 2, we show that for each point (s_1, \dots, s_{d-1}) in \mathcal{G} we can construct a WD of p . In fact, we prove the next result.

Theorem (Theorem 2.1). *Let p be a real binary form of degree d . Then, there exists an open and dense semialgebraic set \mathcal{G} in \mathbb{R}^{d-1} such that p has the following Waring decomposition:*

$$p(x, y) = \sum_{j=1}^{d-1} \lambda_j (x + s_j y)^d + \lambda_d (R_1 x + R_2 y)^d, \quad (1.2)$$

with $\mathbf{s} = (s_1, \dots, s_{d-1}) \in \mathcal{G}$ and R_1, R_2 rational expressions in s_1, \dots, s_{d-1} .

The decomposition (1.2) is optimal for many cases. Bleckherman's work on typical ranks of real binary forms, Bleckherman (2015), implies that there is a nonempty open set of real binary forms whose minimum length for a WD is d . The monomials, except x^d and y^d , are in this open set, as it is shown in Boij et al. (2011). Then, applying the previous theorem to the monomials, we obtain a parametric presentation of the real monomials $x^\ell y^{d-\ell}$, $\ell > 1$, of optimal length, a situation that is quite different from the complex case, see for example Boij et al. (2011).

Finally, we point out some works focused on the algorithmic aspects such as Bender et al. (2016, 2020), García-Marco et al. (2017) or Pratt (2018), to mention some of the most recent. Our contribution to this topic is an algorithm, Algorithm 1, to compute WD for p of length at most d . For complex binary forms similar problems have been studied in Bender et al. (2016, 2020) using different techniques. We include an illustrative case, in Ex. 3.1, that emphasizes the differences between the real and the complex approaches to these problems.

Our Algorithm 1 is based on the computation of a semialgebraic subset \mathcal{G} of \mathbb{R}^{d-1} , associated with the real form p , where there is enough data to construct the required real linear forms. We also present a simple algorithm that, given p , obtains a point of its associated semialgebraic set \mathcal{G} as the output data. It is Algorithm 2. We must point out that this algorithm does not work for general semialgebraic sets, but only for \mathcal{G} . In Basu et al. (2006), it is given a general algorithm to identify a

point in a nonempty semialgebraic set, nevertheless we have not use this algorithm because of its complexity bounds. Information on the complexity of our algorithms is included in Remark 3.7.

As far as we know, Algorithm 1 is the first *algorithm* to decompose a real binary form of degree d as a linear combination of d -th powers of real linear forms. Moreover, our techniques give a semialgebraic family of such decompositions (see Theorem 2.1) and Algorithm 1 computes one of them.

The paper is organized as follows. In Section 2, Theorem 2.1 presents the parametric behavior of Waring decompositions in $(WD)_d$ for a real binary form p . Section 3 includes an algorithm for computing such WD (see Algorithm 1). It is based on an effective method to choose an appropriate set of parameters (see Algorithm 2). The correctness of the algorithms is proved in Proposition 3.5 and Theorem 3.6. Information on the complexity of Algorithm 1 can be found in Remark 3.7.

An illustrative example is included in Section 3.1. We have used Maple 18 to perform these computations.

2. Real Waring decomposition of length at most d

Let \mathcal{B}_d be the real vector space of real binary forms of degree d in the variables x, y , and let p be a real binary form in \mathcal{B}_d ,

$$p(x, y) = p_{\vec{c}}(x, y) = \sum_{i=0}^d \binom{d}{i} c_i x^i y^{d-i}, \quad (2.1)$$

with $\vec{c} = (c_0, \dots, c_d)^t \in \mathbb{R}^{d+1} \setminus \{\vec{0}\}$. A *Waring Decomposition (WD) over \mathbb{R} of length r for p* is any rewrite of the form p as a linear combination of d -th powers of linear forms $\ell_i = \alpha_i x + \beta_i y$, $i = 1, \dots, r$, say

$$p(x, y) = \sum_{i=1}^r \lambda_i \ell_i^d, \quad \text{for some non zero real numbers } \lambda_i. \quad (2.2)$$

We also require that this expression is not redundant, that is, ℓ_1, \dots, ℓ_r are pairwise linear independent. The number r is called the *length of the WD*. Moreover, when the expression (2.2) has minimal length, r is called the *real rank of p* .

Let us fix a real binary form p in \mathcal{B}_d . In this section we present a procedure to compute a WD for p with length at most d . This number is an upper bound for the real rank of p . This bound d was proved in Comon and Ottaviani (2012), but no effective constructions were given there.

Next we give a semialgebraic description of a family of WD for any real binary form. The proof is constructive, which provides a method to construct the WD that is stated.

Theorem 2.1. *Let p be a real binary form of degree d . Then, there exists an open and dense semialgebraic set \mathcal{G} in \mathbb{R}^{d-1} such that p has the following Waring decomposition:*

$$p(x, y) = \sum_{j=1}^{d-1} \lambda_j (x + s_j y)^d + \lambda_d (R_1 x + R_2 y)^d, \quad (2.3)$$

with $\mathbf{s} = (s_1, \dots, s_{d-1}) \in \mathcal{G}$ and R_1, R_2 rational expressions in s_1, \dots, s_{d-1} .

Proof. Let p be as in (2.1). We can assume that some coefficient c_i , $i \in \{1, \dots, d\}$, of the form p is nonzero, since otherwise $p = c_0 y^d$ and the result is trivial.

We consider two independent sets of variables $\{X_0, \dots, X_d\}$ and $\{S_1, \dots, S_{d-1}\}$, and the $(d+1) \times (d+1)$ matrix

$$V = \begin{pmatrix} X_0 & 1 & \cdots & 1 & c_d \\ X_1 & S_1 & \cdots & S_{d-1} & c_{d-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ X_d & S_1^d & \cdots & S_{d-1}^d & c_0 \end{pmatrix} \quad (2.4)$$

whose determinant is a linear polynomial in X_0, \dots, X_d with coefficients $\Delta_0, \dots, \Delta_d$ in the ring $\mathbb{R}[S_1, \dots, S_{d-1}]$, that is

$$\det(V) = h(X_0, \dots, X_d) = \Delta_0 X_0 + \Delta_1 X_1 + \dots + \Delta_d X_d. \quad (2.5)$$

It can be checked that

$$\Delta_d(\mathbf{S}) = (-1)^d \Pi_{d-1}(\mathbf{S}) \left(\sum_{k=1}^d (-1)^{k-1} c_k \sigma_{k-1}(\mathbf{S}) \right), \quad (2.6)$$

where $\Pi_{d-1}(\mathbf{S}) = \prod_{1 \leq i < j \leq d-1} (S_j - S_i)$ and $\sigma_k(\mathbf{S}) = \sum_{1 \leq i_1 < i_2 < \dots < i_k \leq d-1} S_{i_1} \dots S_{i_k}$ are the elementary symmetric polynomials (we take $\sigma_0 \equiv 1$).

We define $\Omega = \mathbb{R}^{d-1} \setminus \{\Delta_d(\mathbf{S}) = 0\}$, which is an open and dense subset in \mathbb{R}^{d-1} ; see, for example, Bochnak et al. (1998).

Now, we choose a point $\mathbf{s} = (s_1, \dots, s_{d-1}) \in \Omega$ and define the real polynomial h^* in the new variable T as

$$h^*(T) = h(1, T, T^2, \dots, T^d) = \Delta_0(\mathbf{s}) + \Delta_1(\mathbf{s})T + \dots + \Delta_{d-1}(\mathbf{s})T^{d-1} + \Delta_d(\mathbf{s})T^d$$

which, clearly, has d real roots since $h^*(s_i) = 0$, $i = 1, \dots, d-1$. Therefore, for some $R \in \mathbb{R}$,

$$h^*(T) = \Delta_d(\mathbf{s}) [(T - s_1) \dots (T - s_{d-1})(T - R)] = \Delta_d(\mathbf{s}) \left[T^d - \left(\sum_{i=1}^{d-1} s_i + R \right) T^{d-1} + \dots + (-1)^d R \prod_{i=1}^{d-1} s_i \right],$$

and $R = -\Delta_{d-1}(\mathbf{s})/\Delta_d(\mathbf{s}) - \sum_{i=1}^{d-1} s_i$.

According to Sylvester's algorithm, see Brachat et al. (2010), to find the coefficients $\{\lambda_i\}_1^d$ of decomposition (2.3) we have to solve the linear system:

$$M \vec{\lambda} = \vec{c} \quad (2.7)$$

where $\vec{c} = (c_d, \dots, c_0)^t$, $\vec{\lambda} = (\lambda_1, \dots, \lambda_d)^t$ and the matrix M is defined as

$$M = \begin{pmatrix} 1 & \dots & 1 & 1 \\ s_1 & \dots & s_{d-1} & R \\ \vdots & \ddots & \vdots & \vdots \\ s_1^d & \dots & s_{d-1}^d & R^d \end{pmatrix}. \quad (2.8)$$

We can ensure that this system has a solution if $\text{rank}(M) = d = \text{rank}(M | \vec{c})$. As M is a Vandermonde matrix, it is sufficient that the roots of $h^*(T)$ are pairwise different for, in that case, $\text{rank}(M) = d$, and $\det(M | \vec{c}) = h^*(R) = 0$, which implies that $\text{rank}(M | \vec{c}) = d$.

Thus, if we take $\mathbf{s} = (s_1, \dots, s_{d-1}) \in \mathcal{G}$, where

$$\mathcal{G} = \Omega \setminus \left(\bigcup_{i=1}^{d-1} \{\Delta_{d-1}(\mathbf{S}) + S_i \Delta_d(\mathbf{S}) = 0\} \right), \quad (2.9)$$

all the roots of $h^*(T)$ are different and taking $R_1 = 1$, $R_2 = R$ we are done. \square

Remark 2.2. a) From the expressions for $h^*(T)$, it can be seen that

$$\Delta_{d-1}(\mathbf{S}) = (-1)^d \Pi_{d-1}(\mathbf{S}) \left(\sigma_1(\mathbf{S}) \sum_{k=1}^d (-1)^k c_k \sigma_{k-1}(\mathbf{S}) - \sum_{k=0}^{d-1} (-1)^k c_k \sigma_k(\mathbf{S}) \right). \quad (2.10)$$

b) Suppose we take $\mathbf{s} = (s_1, \dots, s_{d-1}) \in \mathbb{R}^{d-1}$ such that $\Pi_{d-1}(\mathbf{s}) \neq 0$ but $\Delta_d(\mathbf{s}) = 0$. Then

$$p(x, y) = \sum_{j=1}^{d-1} \lambda_j (x + s_j y)^d + \lambda_d y^d, \quad (2.11)$$

since in this case the linear system $M\vec{\lambda} = \vec{c}$ with

$$M = \begin{pmatrix} 1 & \cdots & 1 & 0 \\ s_1 & \cdots & s_{d-1} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ s_1^d & \cdots & s_{d-1}^d & 1 \end{pmatrix}, \quad (2.12)$$

is compatible. In fact, in this case $\text{rank}(M) = d = \text{rank}(M | \vec{c})$ because the s_i are pairwise different and $\det(M | \vec{c}) = -\Delta_d(\mathbf{s}) = 0$.

In other words, Theorem 2.1 is still true if we enlarge \mathcal{G} with the set $\{\Pi_{d-1}(\mathbf{s}) \neq 0\} \cap \{\Delta_d(\mathbf{s}) = 0\} \subset \mathbb{R}^{d-1}$.

c) An important property of the WD in (2.3) is that if all the coefficients of p are rational numbers then taking suitable $s_i \in \mathbb{Q}$, for all i , the WD is also rational (i.e., $\lambda_i \in \mathbb{Q}$, for all i).

Next, we give some examples of the previous procedure.

Example 2.3. Take $p(x, y) = 5226y^5 + 4970xy^4 + 1860x^2y^3 + 340x^3y^2 + 30x^4y + x^5$. We define the matrix

$$V = \begin{pmatrix} X_0 & 1 & 1 & 1 & 1 & 1 \\ X_1 & S_1 & S_2 & S_3 & S_4 & 6 \\ X_2 & S_1^2 & S_2^2 & S_3^2 & S_4^2 & 34 \\ X_3 & S_1^3 & S_2^3 & S_3^3 & S_4^3 & 186 \\ X_4 & S_1^4 & S_2^4 & S_3^4 & S_4^4 & 994 \\ X_5 & S_1^5 & S_2^5 & S_3^5 & S_4^5 & 5226 \end{pmatrix} \quad (2.13)$$

and its determinant, according to (2.4),

$$\det(V) = h(X_0, \dots, X_5) = \Delta_0 X_0 + \Delta_1 X_1 + \cdots + \Delta_5 X_5. \quad (2.14)$$

Choosing $\mathbf{s} = (1, -1, 2, -2)$, we obtain $R = \frac{120}{23}$ and we have a WD of length 5, but if $\mathbf{s} = (3, -3, 4, -4)$ or if $\mathbf{s} = (1, 2, 3, 4)$, for example, then $R = 5$ and we have found a WD with length 2, value that, in this case, gives us the real rank of p . Hence, we can write

$$\begin{aligned} p(x, y) &= \frac{70}{97}(x+y)^5 - \frac{28}{143}(x-y)^5 - \frac{35}{37}(x+2y)^5 + \frac{5}{83}(x-2y)^5 + \frac{57927087}{42597841}\left(x + \frac{120}{23}y\right)^5 \\ &= -(x+4y)^5 + 2(x+5y)^5. \end{aligned}$$

This last result is also reached if we choose $\mathbf{s} = (4, 5, s_1, s_2)$. In that case $\Delta_5 = 0$, but we can define the corresponding WD following item b) of the Remark 2.2.

Example 2.4. Observe that for $p(x, y) = 3x^2y + y^3$, we have $\Delta_3 = \begin{vmatrix} 1 & 1 & 0 \\ s_1 & -s_1 & 1 \\ s_1^2 & s_1^2 & 0 \end{vmatrix} = 0$ if we choose $s_2 = -s_1$, for any $s_1 \in \mathbb{R}$. So, we construct M as (2.12) and the corresponding linear system outputs

$$\lambda_1 = \frac{1}{2s_1}, \lambda_2 = -\frac{1}{2s_1}, \lambda_3 = 1 - s_1^2,$$

for every $s_1 \neq 0$. Then,

$$3x^2y + y^3 = \frac{1}{2s_1} (x + s_1y)^3 - \frac{1}{2s_1} (x - s_1y)^3 + (1 - s_1^2)y^3$$

and we obtain a one-parameter family of WD.

Note that if $s_1 = \pm 1$ we obtain a shorter expression that shows us that p is a real binary form of real rank 2.

Example 2.5. Take $p(x, y) = 240y^4 + 224xy^3 + 72x^2y^2 + 8x^3y + x^4$. We can choose $\mathbf{s} = (0, 1, -1)$ and then $R = \frac{38}{9}$. But, for $\mathbf{s} = (0, 2, -2)$ we obtain $R = 4$. Hence, we get two decompositions of length 4 and 3 (the one that gives us the real rank), respectively:

$$\begin{aligned} p(x, y) &= \frac{34}{19}x^4 - \frac{40}{29}(x + y)^4 - \frac{8}{47}(x - y)^4 + \frac{19683}{25897} \left(x + \frac{38}{9}y\right)^4 = \\ &= -(x + 2y)^4 + (x + 4y)^4 + x^4. \end{aligned}$$

Remark 2.6. Consider the following analytic path

$$\gamma : [0, 1] \rightarrow \mathcal{B}_d, \gamma(\varepsilon) = (\varepsilon^2 + 1)y^4 + 6\varepsilon^2x^2y^2 + 4\varepsilon x^3y.$$

It joins the binary forms $\gamma(0) = y^4$ and $\gamma(1) = 2y^4 + 6x^2y^2 + 4x^3y$. Next we apply our construction to the form $\gamma(\varepsilon)$. Observe that the system associated with the matrix (2.8), with $\mathbf{s} = (1, -1, s^*)$, for any value of s^* , gives $R = -1/\varepsilon$ and

$$\lambda_1 = 0, \quad \lambda_2 = \frac{\varepsilon(\varepsilon^2 + \varepsilon + 1)}{2(\varepsilon + 1)}, \quad \lambda_3 = \frac{\varepsilon(\varepsilon^2 - \varepsilon + 1)}{2(\varepsilon - 1)}, \quad \lambda_4 = -\frac{\varepsilon^4}{\varepsilon^2 - 1},$$

and then, for $\varepsilon \in [0, 1)$, we have

$$\gamma(\varepsilon) = \lambda_2 (x + y)^4 + \lambda_3 (x - y)^4 - \frac{1}{\varepsilon^2 - 1} (\varepsilon x - y)^4.$$

Therefore, real rank 1 forms can be in the closure of the set of forms of real rank 3.

3. The algorithm Real Waring Decomposition (RWD)

In this section we present an algorithm to compute a real WD of length at most d for any real binary form. Its correctness is based in the construction of a suitable Vandermonde matrix and the resolution of its associated linear system. The idea of the algorithm is to guarantee a computable choice of an element of the set \mathcal{G} in Theorem 2.1. When the s_i in this theorem are considered as parameters, an appropriate use of Lemma 3.2 allows to compute the desired WD and hence to solve effectively $(WD)_d$.

Clearly, given a real binary form one can consider it as a complex binary form and try to use the recent algorithms to find its Waring decomposition where now the λ_i would be, in principle, complex numbers, as in Bender et al. (2016, 2020). Observe that if we apply the Sylvester's Algorithm to p , see Brachat et al. (2010), there is no guarantee that the linear forms we obtain have real coefficients. This fact is quite delicate and it relies on the fact that \mathbb{R} is a real closed field in an essential way. We will review this method before exposing our algorithm. We would like to point out that the linear forms we output as a decomposition of a real binary form p in Theorem 2.1 have real coefficients (and even rational coefficients if all the input data are rational).

The Fast Algorithm in Bender et al. (2016) is based on Sylvester's Theorem and the algorithm of Comon and Mourrain (1996) improving some computational steps by avoiding the incremental

construction that involves successively computing the kernel of Hankel matrices. Both algorithms allow to find a WD with the minimum possible length for the given p with complex coefficients. However, the sufficient condition that ensures the existence of a WD is that there exists a square-free binary form. This is no longer true in the real case, see Example 3.1.

We propose an algorithmic treatment for the choice of the linear forms with which to construct a WD associated with a real form $p \in \mathbb{R}[x, y]$. Consequently, this new algorithm gives another proof of Theorem 2.1 which reduces the number of required parameters. Theorem 3.6 establishes the correctness of this new algorithm. Examples are included in Section 3.1.

From the computational point of view, the problem of choosing algorithmically a point in a semi-algebraic set (for what interests us, the set \mathcal{G} of the Theorem 2.1) can be a hard problem. It could be done using the Algorithm on Connected Components in Basu et al. (2006), (Algorithm 13.1, page 549) but it has high complexity. In Algorithm 2, given the special characteristics of the set \mathcal{G} we are able to determine an appropriate choice of real linear binary forms to solve with the techniques of Real Geometry the problem $(WD)_d$ over the real field.

Here is an example to illustrate some of the differences between the two approaches, over \mathbb{R} and over \mathbb{C} , of the WD problem for real binary forms.

Example 3.1. Consider the real binary form

$$p(x, y) = ax^3 + 3bx^2y - 3axy^2 - by^3, \text{ for } a \neq 0, b \neq 0. \quad (3.1)$$

It is easy to check it is not possible to find a WD of length 2 over \mathbb{R} . Next, consider the matrix as in the Sylvester's Algorithm, see for instance Brachat et al. (2010),

$$\begin{pmatrix} -b & -a & b \\ -a & b & a \end{pmatrix}.$$

Its kernel is generated by $(1, 0, 1)$, that derives in a square-free polynomial and following the steps 4, 5 and 6 of the Algorithm 1 in Bender et al. (2016), we can find a complex WD of length 2; in fact:

$$p(x, y) = \left(\frac{a}{2} - \frac{bi}{2}\right)(x + iy)^3 + \left(\frac{a}{2} + \frac{bi}{2}\right)(x - iy)^3.$$

But, over \mathbb{R} , it is necessary to go to length 3. For example, we can compute a one-parameter family of WD for p :

$$p(x, y) = -\frac{a^2 + b^2}{2s(as - b)}(x + sy)^3 - \frac{a^2 + b^2}{2s(as + b)}(x - sy)^3 + \frac{a^3(1 + s^2)}{a^2s^2 - b^2}\left(x + \frac{b}{a}y\right)^3, \quad (3.2)$$

for all $s \neq 0, \pm b/a$.

Moreover, observe that the situations in which the shortest decomposition is unique are certainly scarce, see García-Marco et al. (2017), when we consider real WD. It is easy to see that polynomial (3.1) has a unique minimal WD over \mathbb{C} , but not over \mathbb{R} :

$$p(x, y) = -\frac{a^2 + b^2}{s(bs^2 + 2as - b)}(x + sy)^3 - \frac{(a + bs)^3}{(as - b)(bs^2 + 2as - b)}\left(x + \frac{-as + b}{a + bs}y\right)^3 + \frac{(a^2 + b^2)(s^2 + 1)}{s(as - b)}x^3, \quad \text{for } s \in \mathbb{R} \setminus \left\{0, \frac{a}{b}, \frac{b}{a}, \frac{-a \pm \sqrt{a^2 + b^2}}{b}\right\},$$

that is a different WD than (3.2) since $b \neq 0$ and $s \neq 0$.

Next, we will first establish the main algorithm, Algorithm 1, then we will go into details with the subroutine for choosing a finite set of appropriate parameters in Algorithm 2. In general, the linear forms for the WD (2.3) can be chosen in the semialgebraic set \mathcal{G} as seen in Theorem 2.1.

Algorithm 1: Real Waring Decomposition (RWD)).**Input:** $p(x, y) = \sum_{i=0}^d \binom{d}{i} c_i x^i y^{d-i}$, a real binary form of degree d .**Output:** an array $[\mathbf{s}, \mathbf{R}, \boldsymbol{\lambda}]$ such that $\mathbf{s} = (s_1, \dots, s_{d-1}) \in \mathbb{R}^{d-1}$, $\mathbf{R} = (R_1, R_2)$ and $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_d]$ that verify formula (2.3) in Theorem 2.1.

```

1 Call Algorithm 2, WDPParameters( $p$ ) =  $[\mathbf{s}, \mathbf{R}, \tilde{\Delta}]$ .
2 if  $\tilde{\Delta} = 1$  then
3   Define the matrix  $M$  as in equation (2.8).
4   Define  $\mathbf{R} = (1, R)$ 
5 else
6   Define the matrix  $M$  as in equation (2.12).
7   Define  $\mathbf{R} = (0, 1)$ .
8 end
9 Define the list  $\boldsymbol{\lambda}$  as the unique solution of the system  $M\tilde{\boldsymbol{\lambda}} = \bar{\mathbf{c}}$ , where  $\bar{\mathbf{c}} = (c_d, \dots, c_0)^t$  and  $\tilde{\boldsymbol{\lambda}} = (\lambda_1, \dots, \lambda_d)^t$ .
10 return  $[\mathbf{s}, \mathbf{R}, \boldsymbol{\lambda}]$ .
```

Lemma 3.2 (see for instance Łojasiewicz (1991), p. 86). Let $z^n + a_1 z^{n-1} + \dots + a_n$ be a monic polynomial with complex coefficients. Let ζ_1, \dots, ζ_n be a complete sequence of its roots. Then, we have

$$(|a_i| \leq r, i = 1, \dots, n) \implies (|\zeta_j| \leq 2r, j = 1, \dots, n). \quad (3.3)$$

Next we will proceed to detail the algorithm that allow us to construct an effective choice of the family of real linear forms to obtain a WD of length at most d . We introduce some definitions in order to prove the correctness of Algorithm 1.

Definition 3.3. Let $p(x, y)$ be a real binary form of odd degree d . Let $\tilde{\mathbf{s}} = (s_1, \dots, s_{d-1}, s_d) = (\mathbf{s}, s_d)$ be a vector in \mathbb{R}^d . We say that $\tilde{\mathbf{s}}$ is a vector of suitable parameters for a WD of p if it satisfies the following conditions:

- $s_i \neq s_j$, for $i \neq j$,
- $s_d = -\frac{\Delta_{d-1}(\mathbf{s})}{\Delta_d(\mathbf{s})} - \sum_{i=1}^{d-1} s_i$ whenever $\Delta_d(\mathbf{s}) \neq 0$, where $\Delta_i(\mathbf{s})$ stands for the evaluation at \mathbf{s} of the polynomial Δ_i defined in (2.5).

Remark 3.4. One can easily verify the following result.

Let p be a real binary form of degree d and let $\tilde{\mathbf{s}} = (\mathbf{s}, s_d)$ be a vector in \mathbb{R}^d of suitable parameters for a WD of p such that $\Delta_d(\mathbf{s}) \neq 0$. Then the point \mathbf{s} belongs to the semialgebraic set \mathcal{G} defined in Theorem 2.1.

The following proposition guarantees the correctness of Algorithm 2.

Proposition 3.5. Let p be a real binary form of degree d . The output list $[\mathbf{s}, \mathbf{R}, \tilde{\Delta}]$ of Algorithm 2 applied to p defines a vector $\tilde{\mathbf{s}} = (s_1, \dots, s_{d-1}, R)$ of suitable parameters for a WD of p .

Moreover, if $\tilde{\Delta} = 1$ (respectively $\tilde{\Delta} = 0$) then the linear system (2.8) (resp. (2.12)) using the vector $\tilde{\mathbf{s}}$ has a unique solution.

In the following proof we use the notations in Algorithm 2.

Proof. Let $[\mathbf{s}, \mathbf{R}, \tilde{\Delta}]$ be the returned array of Algorithm 2. First assume $\tilde{\Delta} = 1$. As $\tilde{\mathbf{s}} := (s_1, \dots, s_{d-1}, R)$ with $R = -\frac{\Delta_{d-1}(s_{d-1})}{\Delta_d(s_{d-1})} - \sum_{i=1}^{d-1} s_i$, by the choice of \mathbf{s} in the algorithm we have $s_i \neq s_j$ if $1 \leq i < j \leq d-1$, so for $\tilde{\mathbf{s}}$ to be a vector of suitable parameters it only remains to prove $R \neq s_i$ for $i = 1, \dots, d-1$.

Algorithm 2: WDPParameters.

Input: $p(x, y) = \sum_{i=0}^d \binom{d}{i} c_i x^i y^{d-i}$, a real binary form.

Output: an array $[s, R, \tilde{\Delta}]$ where $(s, R) \in \mathbb{R}^{d-1} \times \mathbb{R}$ is a vector of suitable parameters for the WD of p and $\tilde{\Delta}$ is a switch that returns 0 if $\Delta_d(s) = 0$ or 1 otherwise.

```

1 Define  $s_i := (-1)^{i+1} \lfloor \frac{i+1}{2} \rfloor$ , for  $i = 1, \dots, d-1$ , and  $s = (s_1, \dots, s_{d-1})$ .
2 Compute  $\Delta_d(s)$ .
3 if  $\Delta_d(s) = 0$  then
4   | Set  $R = 0$  and  $\tilde{\Delta} = 0$ .
5 else
6   | Set  $\tilde{\Delta} = 1$ .
7   | Compute  $\Delta_{d-1}(s)$  and  $R = -\frac{\Delta_{d-1}(s)}{\Delta_d(s)} - \sum_{i=1}^{d-1} s_i$ .
8   | if  $R \in \{s_i\}_{i=1}^{d-1}$  then
9     | Compute  $\Delta_d(X) = \Delta_d(s_1, \dots, s_{d-2}, X)$ ,  $\Delta_{d-1}(X) = \Delta_{d-1}(s_1, \dots, s_{d-2}, X)$  and  $a = \sum_{i=1}^{d-2} s_i$ .
10    | Define  $\delta_d$  as the maximum absolute value of the coefficients of  $\Delta_d(X)$ .
11    | Define  $\delta_{d-1,0}$  as the maximum absolute value of the coefficients of  $\Delta_{d-1}(X) + (a + 2X)\Delta_d(X)$ .
12    | Define  $\delta_{d-1,i}$  as the maximum absolute value of the coefficients of
13      |  $\Delta_{d-1}(X) + (a + s_i + X)\Delta_d(X)$ ,  $i = 1, \dots, d-2$ .
14    | Compute  $m = \max\{\lfloor \frac{d+1}{2} \rfloor, \delta_d, \delta_{d-1,0}, \delta_{d-1,1}, \dots, \delta_{d-1,d-2}\}$ .
15    | Define  $s_{d-1} = 2m + 1$ .
16    | Compute  $R = -\frac{\Delta_{d-1}(s_{d-1})}{\Delta_d(s_{d-1})} - \sum_{i=1}^{d-1} s_i$ .
17  end
18 return  $[s, R, \tilde{\Delta}]$ .
```

Suppose $R = s_i$ for some i . Then

$$R = -\frac{\Delta_{d-1}(s_{d-1})}{\Delta_d(s_{d-1})} - \sum_{k=1}^{d-1} s_k = -\frac{\Delta_{d-1}(s_{d-1})}{\Delta_d(s_{d-1})} - a - s_{d-1} = s_i,$$

that is,

$$\Delta_{d-1}(s_{d-1}) + (a + s_i + s_{d-1})\Delta_d(s_{d-1}) = 0.$$

But this means that s_{d-1} is a root of the polynomial $\Delta_{d-1}(X) + (a + s_i + X)\Delta_d(X)$ if $i \in \{1, \dots, d-2\}$ or a root of $\Delta_{d-1}(X) + (a + 2X)\Delta_d(X)$ when $i = d-1$, which contradicts the choice of s_{d-1} in Algorithm 2 and the statement of Lemma 3.2.

In case $\tilde{\Delta} = 0$, as $\Delta_d(s) = 0$ and $R = 0$ we only have to prove that all the components of \tilde{s} are different, but this is immediate by the choice of s .

The uniqueness of the solutions of the linear system in both cases is an easy consequence of the proof of Theorem 2.1 and item b) of Remark 2.2. \square

Finally, next theorem guarantees the correctness of Algorithm 1.

Theorem 3.6. Let p be a real binary form of degree d . The output list $[s, R, \lambda]$ from Algorithm 1 applied to p verifies formula (2.3) in Theorem 2.1.

In the following proof we use the notations in Algorithms 1 and 2.

Proof. A consequence of the proof of Proposition 3.5 is that the point $s = (s_1, \dots, s_{d-1})$, if $\Delta_d(s) \neq 0$, belongs to the semialgebraic set \mathcal{G} as defined in the proof of Theorem 2.1, since $s_i \neq s_j$, for $i \neq j$, by the choice of s_{d-1} in Step 14 of Algorithm 2.

In case $\Delta_d(s) = 0$, the point s belongs to the semialgebraic set \mathcal{G} enlarged as indicated in item b) of Remark 2.2.

In both cases, the corresponding linear systems have a unique solution which verifies formula (2.3) in Theorem 2.1. Hence Algorithm 1 computes a WD of length at most d for any real binary form. \square

Remark 3.7. It can be seen that the arithmetic complexity of Algorithm 1, together with Algorithm 2, is $O(d \log^2 d)$. In fact, the Vandermonde linear system to be solved in Algorithm 1 is known to have arithmetic operational complexity not more than $O(d \log^2 d)$, cf. Li (2000). On the other hand, to evaluate R in Algorithm 2, we have to evaluate $\Delta_{d-1}(\mathbf{s})$ and $\Delta_{d-1}(\mathbf{s})$. By equations (2.6) and (2.10) one has to evaluate the elementary symmetric polynomials at \mathbf{s} , that is, to compute the coefficients of the polynomial $\prod_{i=1}^{d-1} (x - s_i)$, problem which is known to be equivalent to polynomial interpolation at points $(s_1, 0), \dots, (s_{d-1}, 0)$ and for this problem there are fast polynomial algorithms of arithmetic complexity $O(d \log^2 d)$, Pan et al. (1993).

Next we present an example to illustrate our results.

3.1. A detailed example

Next consider the binary form

$$p_a(x, y) = x^5 + 10x^4y + 10x^3y^2 + 10x^2y^3 + 5axy^4 + y^5, \text{ for } a = 2, 1. \quad (3.4)$$

Running Algorithm 1 we obtain:

- For $a = 2$:
 1. Set $d := 5$ and $c := (1, 2, 1, 1, 2, 1)$.
 2. Call Algorithm 2, that output the list $[\mathbf{s}, R, \tilde{\Delta}] = [(1, -1, 2, -2), 4, 1]$.
 3. As $\tilde{\Delta} = 1$, we define the linear system

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 2 & -2 & 4 \\ 1 & 1 & 4 & 4 & 16 \\ 1 & -1 & 8 & -8 & 64 \\ 1 & 1 & 16 & 16 & 256 \\ 1 & -1 & 32 & -32 & 1024 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 1 \\ 2 \\ 1 \end{pmatrix}.$$

4. Define $\mathbf{R} = (1, 4)$.
5. RETURN $\left[(1, -1, 2, -2), (1, 4), \left(\frac{31}{18}, -\frac{7}{10}, -\frac{1}{8}, \frac{7}{72}, \frac{1}{180} \right) \right]$.

Therefore,

$$p_2(x, y) = \frac{31}{18}(x+y)^5 - \frac{7}{10}(x-y)^5 - \frac{1}{8}(x+2y)^5 + \frac{7}{72}(x-2y)^5 + \frac{1}{180}(x+4y)^5.$$

- For $a = 1$:
 1. Set $d := 5$ and $c := (1, 1, 1, 1, 2, 1)$.
 2. Call the Algorithm 2 that output the list $[\mathbf{s}, R, \tilde{\Delta}] = [(1, -1, 2, -2), 0, 0]$.
 3. As $\tilde{\Delta} = 0$, we define the linear system

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & -1 & 2 & -2 & 0 \\ 1 & 1 & 4 & 4 & 0 \\ 1 & -1 & 8 & -8 & 0 \\ 1 & 1 & 16 & 16 & 0 \\ 1 & -1 & 32 & -32 & 1 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

4. Define $\mathbf{R} = (0, 1)$.

5. RETURN $\left[(1, -1, 2, -2), (0, 1), \left(\frac{5}{3}, -\frac{2}{3}, -\frac{1}{12}, \frac{1}{12}, 4 \right) \right]$.

Therefore

$$p_1(x, y) = \frac{5}{3}(x+y)^5 - \frac{2}{3}(x-y)^5 - \frac{1}{12}(x+2y)^5 + \frac{1}{12}(x-2y)^5 + 4y^5.$$

Remark 3.8. For $a \neq 1$, Algorithm 2 outputs

$$[\mathbf{s}, R, \tilde{\Delta}] = \left[(1, -1, 2, -2), \frac{4}{a-1}, 1 \right]$$

and Algorithm 1 returns $\left[(1, -1, 2, -2), \left(1, \frac{4}{a-1} \right), \lambda \right]$, with

$$\lambda = \left(-\frac{a^2 - 12a + 51}{6(a-5)}, -\frac{a^2 + 2a + 13}{6(a+3)}, \frac{a^2 - 4a + 7}{24(a-3)}, \frac{a^2 + 3}{24(a+1)}, \frac{(a-1)^5}{4(a+1)(a+3)(a-3)(a-5)} \right),$$

properly defined for $a \notin \{-1, 3, -3, 5\}$.

Note that Algorithm 2 is quite restrictive in the way of choosing the parameters for the WD, since it takes integers with opposite sign in its construction. Nevertheless the calculations are simple and fast. In some cases, it is not possible to find a shorter WD with this restriction, but, removing it, we can find an improved one. For instance,

$$p_1(x, y) = \frac{1}{11} \left[\lambda_1(x + \alpha_1 y)^5 + \bar{\lambda}_1(x + \bar{\alpha}_1 y)^5 + \lambda_3(x + \alpha_3 y)^5 + \bar{\lambda}_3(x + \bar{\alpha}_3 y)^5 \right],$$

with $\lambda_1 = -6 + \frac{14}{3}\sqrt{3}$, $\alpha_1 = \frac{1}{2}(1 + \sqrt{3})$, $\lambda_3 = -\frac{23}{2} + \frac{51}{10}\sqrt{5}$ and $\alpha_3 = \frac{1}{2}(-1 + \sqrt{5})$, using the notation $\bar{w} = m - n\sqrt{p}$, when $w = m + n\sqrt{p}$. However, in this article our goal is not to reduce the length of the possible WD, but to compute them with simple and fast calculations.

Remark 3.9. Note that for fixed value of a , we can determine a real value s^* so that, for all $\alpha \in [s^*, \infty)$ such that $(1, -1, \alpha, -\alpha) \in \mathcal{G}$, with \mathcal{G} the semialgebraic set defined in (2.9) depending on p_a . Following the steps 10 to 14, we can calculate that, if $0 \leq a \leq 2$, $s^* = 17$ and $s^* = 16(1-a) + 1$ in another case.

This gives, in general, a way to find the values of $\mathbf{s} \in \mathcal{G}$. Certainly, there are smaller values of α such that $(1, -1, \alpha, -\alpha)$ is also in \mathcal{G} . For instance, the vector $\tilde{\mathbf{s}} = (1 - 1, 2, -2, \frac{4}{a-1})$, as we have seen before, is also a vector of suitable parameters, but our bound guarantees that $\tilde{\mathbf{s}} = (1 - 1, \alpha, -\alpha, \frac{\alpha^2}{a-1})$ is a vector of suitable parameters for all $\alpha \geq s^*$ and accordingly the associated linear system for these values has a unique solution. In fact, if we follow the steps of the Algorithm 1, with the previous vector $\tilde{\mathbf{s}}$, we obtain the one-parameter family of WD:

$$p_2(x, y) = \frac{3\alpha^4 - 5\alpha^2 + 3}{2(\alpha^2 - 1)^2}(x+y)^5 - \frac{\alpha^4 + \alpha^2 + 1}{2(\alpha^2 + 1)(\alpha^2 - 1)}(x-y)^5 - \frac{\alpha^2 - \alpha + 1}{2\alpha^2(\alpha - 1)^2(\alpha + 1)}(x + \alpha y)^5 + \frac{\alpha^2 + \alpha + 1}{2\alpha^2(\alpha - 1)(\alpha + 1)^2}(x - \alpha y)^5 + \frac{1}{\alpha^2(\alpha^2 + 1)(\alpha - 1)^2(\alpha + 1)^2}(x + \alpha^2 y)^5,$$

for any real number α such that $\alpha \neq 0, \pm 1$.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We thank the referees for their useful comments and suggestions which greatly improved the final version of this paper.

The second and third authors acknowledge the support of Grupo UCM 910444.

References

- Alexander, J., Hirschowitz, A., 1995. Polynomial interpolation in several variables. *J. Algebraic Geom.* 4, 201–222.
- Ballico, E., Bernardi, A., 2013. Real and complex rank for real symmetric tensors with low ranks. *Algebra* 2013, 794054.
- Basu, S., Pollack, R., Roy, M.F., 2006. Algorithms in Real Algebraic Geometry, vol. 10. Springer-Verlag Berlin Heidelberg. <https://perso.univ-rennes1.fr/marie-francoise.roy/bpr-ed2-posted3.pdf>.
- Bender, M.R., Faugère, J.C., Perret, L., Tsigaridas, E., 2016. A superfast randomized algorithm to decompose binary forms. In: Proceedings of the ACM on International Symposium on Symbolic and Algebraic Computation. ACM, New York, NY, USA, pp. 79–86.
- Bender, M.R., Faugère, J.C., Perret, L., Tsigaridas, E., 2020. A nearly optimal algorithm to decompose binary forms. *J. Symbolic Comput.* <https://doi.org/10.1016/j.jsc.2020.06.002>.
- Blekherman, G., 2015. Typical real ranks of binary forms. *Found. Comput. Math.* 15, 793–798. <https://www.springerprofessional.de/en/typical-real-ranks-of-binary-forms/11689832>.
- Bochnak, J., Coste, M., Roy, M.F., 1998. Real Algebraic Geometry. *Ergebnisse der Mathematik und ihrer Grenzgebiete (3)*, vol. 36. Springer-Verlag, Berlin.
- Boij, M., Carlini, E., Geramita, A., 2011. Monomials as sums of powers: the real binary case. *Proc. Am. Math. Soc.* 139, 3039–3043. <https://www.ams.org/journals/proc/2011-139-09/S0002-9939-2011-11018-9/S0002-9939-2011-11018-9.pdf>.
- Brachat, J., Comon, P., Mourrain, B., Tsigaridas, E., 2010. Symmetric tensor decomposition. *Linear Algebra Appl.* 433, 1851–1872.
- Brustenga i Moncusí, L., Masuti, S.K., 2019. On the Waring rank of binary forms: the binomial formula and a dihedral cover of rank two forms. *arXiv:1901.08320*.
- Carlini, E., Catalisano, M., Geramita, A.V., 2012. The solution to the Waring problem for monomials and the sum of coprime monomials. *J. Algebra* 370, 5–14.
- Causa, A., Re, R., 2011. On the maximum rank of a real binary form. *Ann. Mat. Pura Appl.* 190, 55–59.
- Comon, P., Lim, L.H., Qi, Y., Ye, K., 2020. Topology of tensor ranks. *Adv. Math.* 367, 107128.
- Comon, P., Mourrain, B., 1996. Decomposition of quantics in sums of powers of linear forms. *Signal Process.* 53, 93–107.
- Comon, P., Ottaviani, G., 2012. On the typical rank of real binary forms. *Linear Multilinear Algebra* 60, 657–667.
- Fröberg, R., Lundqvist, S., Oneto, A., Shapiro, B., 2018. Algebraic stories from one and from the other pockets. *Arnold Math. J.* 4, 137–160. <https://arxiv.org/pdf/1801.01692v1.pdf>.
- García-Marco, I., Koiran, P., Pecatte, T., 2017. Reconstruction algorithms for sums of affine powers. In: Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation. ACM, New York, NY, USA, pp. 317–324.
- Helmke, U., 1992. Waring's problem for binary forms. *J. Pure Appl. Algebra* 80, 29–45. [https://doi.org/10.1016/0022-4049\(92\)90067-P](https://doi.org/10.1016/0022-4049(92)90067-P).
- Li, L., 2000. On the arithmetic operational complexity for solving Vandermonde linear equations. *Jpn. J. Ind. Appl. Math.* 17, 15–18.
- Łojasiewicz, S., 1991. Introduction to Complex Analytic Geometry. Ed. Birkhäuser.
- Pan, V., Sadikou, A., Landowne, E., Tiga, O., 1993. A new approach to fast polynomial interpolation and multipoint evaluation. *Comput. Math. Appl.* 25, 25–30.
- Pratt, K., 2018. Faster algorithms via Waring decompositions. In: The Proceedings of FOCS.
- Reznick, B., 2010. Laws of inertia in higher degree binary forms. *Proc. Am. Math. Soc.* 138, 815–826.
- Tokcan, N., 2017. On the Waring rank of binary forms. *Linear Algebra Appl.* 524, 250–262.