# Deciding positivity of real polynomials

Eberhard Becker, Victoria Powers, and Thorsten Wörmann

Abstract. We describe an algorithm for deciding whether or not a real polynomial is positive semidefinite. The question is reduced to deciding whether or not a certain zero-dimensional system of polynomial equations has a real root.

## 1. Introduction

A real polynomial $f \in \mathbb{R}[X_1, \ldots, X_n]$ is **positive semi-definite**, or **psd**, if $f(x) \geq 0$ for all $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$. In this note we give an algorithm for the following problem: Given a particular polynomial $f$, how can one decide if $f$ is psd or not?

Psd polynomials have been studied extensively by many researchers (see, for example, [**5**]), mainly in the context of finding examples of psd polynomials that are not sums of squares. For all but a few isolated examples, the psd polynomials given in these papers come from monomial substitution into the arithmetic-geometric inequality so one knows *a priori* that the polynomial in question is psd.

The question of deciding the positivity or non-negativity of a real polynomial arises in many problems in Engineering, for example in Control Theory. Algorithms for special types of polynomials have been given, and a test for positivity of arbitrary binary forms in described in [**2**]. In [**3**], a general algorithm for deciding positivity is given. The main idea is to treat all but one of the variables as parameters and apply the Sturm-Habicht algorithm for real root counting in the univariate case. This technique is not practical for any but very small cases since it involves calculation of determinants of large matrices with polynomial entries. The example computed in [**3**] is a sparse polynomial of degree 4 in 3 variables, and in this example some ad-hoc methods are used. Furthermore, part of the procedure involves calculation of real roots of very large degree univariate polynomials, thus some floating point approximations would be needed. Our algorithm has the advantage of being able to handle somewhat larger examples, and uses only purely symbolic methods.

## 2. Real Root Counting

We fix $n \in \mathbb{N}$ and let $\mathbb{R}[X]$ denote $\mathbb{R}[X_1, \ldots, X_n]$. Suppose $f(X) \in \mathbb{R}[X]$. We first note that $f(x) < 0$ for some $x \in \mathbb{R}^n$ if and only if the polynomial

$$t^2 + f(X) \in \mathbb{R}[X, t]$$

1991 *Mathematics Subject Classification.* 12D10, 14P99.

has a real root. Thus our problem can be reduced to the problem of deciding whether or not a polynomial in one more variable has a real root.

More generally, suppose we have a finite number of multivariate real polynomials $f_1, \ldots, f_k$ and we want to decide if the system of equations $f_1 = 0, \ldots, f_k = 0$ has a solution or not. If we are interested in solutions over $\mathbb{C}$, then there are well-known methods for finding them, e.g., Groebner Basis techniques. However, if we are only interested in real solutions, then the problem becomes much harder. (Groebner Basis techniques can not, in general, tell us whether or not there are real solutions.) However, there is one case where algorithms exist for finding real roots, namely, the case where there are only finitely many complex solutions to the polynomial system.

More precisely, given $f_1, \ldots, f_k \in \mathbb{R}[X]$, let $V = V(f_1, \ldots, f_k)$ denote the variety in $\mathbb{C}^n$ of common zeros of the $f_i$'s, and let $V_{\mathbb{R}}(f_1, \ldots, f_k) = V(f_1, \ldots, f_k) \cap \mathbb{R}^n$. If $V(f_1, \ldots, f_k)$ is finite, then there exists an algorithm for counting the number of points in $V_{\mathbb{R}}(f_1, \ldots, f_k)$. In particular, we can decide if $V_{\mathbb{R}}(f_1, \ldots, f_k)$ is empty or not. Furthermore, there exists software implementing this algorithm: The RealSolving package, written by F. Rouillier.

Now consider the problem of deciding, for a single $f \in \mathbb{R}[X]$, whether or not $V_{\mathbb{R}}(f)$ is empty. Of course, $V(f)$ will not be zero-dimensional for $n > 1$, hence we cannot apply RealSolving directly. We need a way to reduce to the case of a zero-dimensional system of polynomials. If $f$ is smooth over $\mathbb{R}$, i.e., the partial derivatives of $f$ do not simultaneously vanish at any point of $V_{\mathbb{R}}(f)$, then a way to do this is to look for critical points of some other polynomial on $V(f)$, using Lagrange multipliers. The set of critical points will be zero-dimensional almost always. (This will be made more precise below).

Definition. Suppose $f, \phi \in \mathbb{R}[X]$ and $\lambda$ is a new indeterminate. For $g \in \mathbb{R}[X]$, let $g_{X_i}$ denote $\dfrac{\partial g}{\partial X_i}$. The *ideal of Lagrange multipliers of f with respect to* $\phi$, denoted $L(f, \phi)$, is

$$(f, \lambda f_{X_1} - \phi_{X_1}, \ldots, \lambda f_{X_n} - \phi_{X_n}),$$

the ideal in $\mathbb{R}[X_1, \ldots, X_n, \lambda]$ generated by $f$ and the partial derivatives of $\lambda f - \phi$.

Lemma 1. *Suppose we are given $f, \phi \in \mathbb{R}[X]$, and suppose $f$ is smooth. If $V_{\mathbb{R}}(f) \neq \emptyset$ and $\phi$ attains a minimum on $V_{\mathbb{R}}(f)$, then $V_{\mathbb{R}}(L(f, \phi)) \neq \emptyset$. If $V_{\mathbb{R}}(L(f, \phi)) \neq \emptyset$, then $V_{\mathbb{R}}(f) \neq \emptyset$.*

Proof. Set $L := L(f, \phi)$. From elementary analysis we know that the real points of the projection of $V(L)$ contain all locally extremal points of $\phi$ under the constraint $f = 0$. Thus if $V_{\mathbb{R}}(f)$ is not empty and $\phi$ attains a minimum on $V_{\mathbb{R}}(f)$, then $V_{\mathbb{R}}(L)$ is not empty. Since the projection of $V_{\mathbb{R}}(L)$ is contained in $V_{\mathbb{R}}(f)$, $V_{\mathbb{R}}(L(f, \phi)) \neq \emptyset$ implies $V_{\mathbb{R}}(f) \neq \emptyset$. □

It follows from Sard's Theorem that $\{\, \phi(x) \mid (x, \lambda) \in V(L(f, \phi)) \,\}$ is finite for a generic choice of $\phi$. (For a proof see, e.g., [**7**], Chap. 3). Thus we can proceed as follows: Choose a "distance function" $\phi$ and check if $L(f, \phi)$ is zero-dimensional as a complex variety. (RealSolving does this automatically, using a Groebner Basis calculation.) If so, then we can test whether $V_{\mathbb{R}}(L(\phi, f))$ is empty. If $V(L(f, \phi))$ is not zero-dimensional, we can change the distance function $\phi$ with the aim of making $V(L(f, \phi))$ zero-dimensional. In practice, we use $\phi = \sum r_i X_i^2$, where $r_i$ are small integers.

REMARK. The assumption that $V(f)$ is smooth is necessary. For example, consider $f(x,y) = (x-1)^3 - y^2$, which obviously has real roots. In this case $V_{\mathbb{R}}(L(f,\phi)) = \emptyset$ for $\phi = x^2+y^2$. The problem is that the point of minimum distance from $(0,0)$ is a singular point of $V_{\mathbb{R}}(f)$. On the other hand, we can get around this problem if $f$ has only finitely many singular points, by first checking for a real singular point. If $V(f)$ has a real singular point we are done, and if not, then we know $V_{\mathbb{R}}(L(f,\phi))$ has no real points iff $V(f)$ has no real points, and the method works in this case.

## 3. The Positivity Algorithm

Given $f \in \mathbb{R}[X]$, we want to decide if $f$ is psd or not. Our method is to reduce to the case of a zero-dimensional system of polynomial equations as follows: Let $t$ be a new indeterminate, and define $F := ft^2 + 1 \in \mathbb{R}[X_1, \dots, X_n, t]$. Then $f$ is psd iff $V_{\mathbb{R}}(F)$ is empty. Furthermore, $F$ is always smooth, thus we can use the method of Lagrange multipliers described in the previous section. If we apply the method with $\phi = r_1X_1^2 + \cdots + r_nX_n^2 + t^2$, then we can simplify $L(F,\phi)$ and eliminate one variable:

PROPOSITION 2. *Given $f \in \mathbb{R}[X]$, define $F = t^2f + 1 \in R[X_1, \dots, X_n, t]$. Then for fixed $r_1, \dots, r_n \in \mathbb{R}^+$, $f$ is psd iff $V_{\mathbb{R}}(F, t^4f_{X_1} + 2r_1X_1, \dots, t^4f_{X_n} + 2r_nX_n) = \emptyset$.*

PROOF. Let $\phi = r_1X_1^2 + \cdots + r_nX_n^2$ and set

$$V := V_{\mathbb{R}}(F, t^4f_{X_1} + 2r_1X_1, \dots, t^4f_{X_n} + 2r_nX_n),$$

$$L := L_{\mathbb{R}}(F,\phi) = V_{\mathbb{R}}(F, \lambda t^2 f_{X_1} - 2r_1X_1, \dots \lambda t^2 f_{X_n} - 2r_nX_n, 2t\lambda f - 2t).$$

From Lemma 1, we have that $f$ is psd iff $V_{\mathbb{R}}(L) = \emptyset$, thus we want to show $L \neq \emptyset$ iff $V \neq \emptyset$. Given $x \in \mathbb{R}^n$ and $\lambda_0, t_0 \in \mathbb{R}$ such that $\alpha = (x, \lambda_0, t_0) \in L$, then since $F(\alpha) = 0$ we must have $f(x) \neq 0$, and $t_0 \neq 0$. Thus $F(\alpha) = 0$ implies $t_0^2 = -1/f(x)$ and $2t_0\lambda_0 f(x) - 2t_0 = 0$ implies $1/f = \lambda_0$. It now follows easily that $(x, t_0) \in V$ and so $V \neq \emptyset$. Conversely, given $(x, t_0) \in V$, setting $\lambda_0 = -t_0^2$ yields $(x, \lambda_0, t_0) \in L$. ☐

EXAMPLE. Let $f = 2x^6 + y^6 - 3x^4y^2 + x^2y^2 - 6y + 5$ and $F = t^2f + 1$. By Proposition 2, $f$ is psd iff

$$V := V(F, t^4(12x^5 - 12x^3y^2 + 2xy^2) + 2x, t^4(6y^5 - 6x^4y + x^2y - 6) + 2y)$$

has no real point. According to RealSolving, $V$ is zero-dimensional, there are 128 points in $V$, and 4 of them are real. Thus $f$ is not psd.

The polynomial $f$ is a special case of the following: For each $a \in \mathbb{R}^+$, set

$$f_a := 2x^6 + y^6 - 3x^4y^2 - 6y + 5 + ax^2y^2.$$

We have just shown that $f_1$ is not psd. We can follow the procedure above to decide if $f_2$ is psd or not. In this case, with $F = t^2f_2 + 1$ and $V$ as above, $V$ has 128 complex roots and none of them are real. Thus $f_2$ is psd. This implies that there exists $b$, $1 < b \leq 2$, such that $f_a$ is not psd for $a \leq b$, and is psd for $a > b$. We would like to find $b$. We cannot follow the exact procedure used for $f_1$, $f_2$ since the RealSolving software cannot handle parameters. However we can get rid of the parameter using the following technique, suggested by Reznick: First note that $f_a(x,y) \geq 0$ trivially when $xy = 0$, and also $f_a(x,y) \geq f_a(x,|y|)$, hence it suffices to assume $x > 0$ and $y > 0$. Then we have $f_a(x,y) \geq 0$ iff $(2x^6 + y^6 - 3x^4y^2 - 6y + 5)/x^2y^2 + a \geq 0$. Hence $-b$ is the minimum of the rational function $G(x,y) := 2x^4y^{-2} - 3x^2 - 6x^{-2}y^{-1} + 5x^{-2}y^{-2} + y^4x^{-2}$. Set $t := x^2$, take derivatives and clear denominators to get that the critical points of $G$ are $V(g_1, g_2)$, where $g_1 = -y^6 - 3t^2y^2 + 6y + 4t^3 - 5$ and $g_2 = 4y^6 + 6y - 4t^3 - 10$. A Groebner Basis of $(g_1, g_2)$ in lex order contains a polynomial in $t$ of degree 13, which has a root at 0, and three other real roots. Solving numerically, we find that $G$ has one real critical value with $x$ between 1 and 2, and that $b$ is approximately 1.2099.

We note in passing that the above trick works for any family of polynomials of the form $G + aH$, where $H$ is psd.

## 4. Practical Considerations

A program for creating the zero-dimensional system in Proposition 2 is trivial to implement; we used Mathematica for this and to create the file needed for RealSolving. We start with the distance function $\phi = X_1^2 + \cdots + X_n^2 + t^2$, and if the system is not zero-dimensional, we add some randomly chosen small integer coefficients to the $X_i^2$'s. For the many examples we computed, it never took more than two tries to create a zero-dimensional system.

The RealSolving software uses a modern extension of classical ideas of Hermite et. al. for real root counting in zero-dimensional varieties, which was discovered by Becker and Wörmann, and by Pederson, independently. For details of the theory behind the algorithm, see [**1**] or [**4**]. The first part of the implementation in RealSolving constructs a multiplication table for the finite-dimensional vector space corresponding to the zero-dimensional ideal; this involves a Groebner Basis calculation and is the most time-consuming part of the program. Then a certain quadratic form is computed with the property that its signature gives the number of real roots of the system. Finally, the signature of the quadratic form is computed. To learn more about RealSolving, or to use it, go to the URL

`http://www.loria.fr/~rouillie/docrs/rs/rs.html`

A detailed description of the theory and the implementation can be found here and in [**6**].

An obvious question to ask is what size problem can be computed in a reasonable amount of time? If we start with $f$ of degree $2d$ in $n$ variables, then the resulting zero-dimensional system (from Proposition 2) has degree at most $2d + 3$ in $n + 1$ variables. For a zero-dimensional system, the amount of time and memory needed for RealSolving depends only on the degree of the variety, i.e., the number of (complex) roots, counted with multiplicities. The current version of RealSolving can, in practice, handle varieties with maximum degree between 200 and 300; a future version will be able to handle degree 1000 or more.

Our computations were done on a 250 megahertz Ultrasparc computation server. For the example above, in which the polynomial has degree 6 in 3 variables and the complex variety has degree 128, the computation took 100 minutes. The problem computed in [**3**] is a polynomial of degree 4 in 3 variables. Using our algorithm, the degree of the variety is 150, and the computation took 2 hours. For several examples of degree 8 in 2 variables, the degree of the variety was around 250, and it took about 6 hours of computation time. Examples of degree 4 in 4 variables were similar. Examples of degree 6 in 4 variables and of degree 4 in 5 variables had corresponding zero-dimensional varieties of degree around 2000. In these cases, RealSolving was able to compute the degree of the varieties in a few minutes, but quickly ran out of memory when attempting to count real solutions.

## References

[1] E. Becker and T. Wörmann, *On the trace formula for quadratic forms*, Cont. Math. **155** (1994), 271–291.

[2] N.K. Bose and E.I. Jury, *Inner algorithm to test for positive definiteness of arbitrary binary forms*, IEEE Trans. Automat. Contr. (1975), 169–170.

[3] N.K. Bose and A.R. Modarressi, *General procedure for multivariable polynomial positivity test with control applications*, IEEE Trans. Automat. Contr. (1976), 696–701.

[4] P. Pederson, M.-F. Roy, and A. Szpirglas, *Counting real zeros in the multivariate case*, in Computational Algebraic Geometry, (Nice 1992), Birkhauser, Boston, MA, 1993, 203–224.

[5] B. Reznick, *Forms derived from the arithmetic-geometric inequality*, Math. Ann. **283** (1989), 431–464.

[6] F. Rouillier, *Algorithmes efficaces pour l'étude des zéros réels des systèmes polynomiaux*, PhD. thesis, Université Rennes, France, 1997.

[7] N. Wallach, *On a theorem of Milnor and Thom*, in Progress in Nonlinear Differential Equations and Their Applications, Birkhäuser, Boston, 1996, 331–348.

Fachbereich Mathematik, Lehrstuhl VI, Universität Dortmund, 44221 Dortmund, Germany

*E-mail address*: `becker@math.uni-dortmund.de`

Department of Mathematics and Computer Science, Emory University, Atlanta, GA 30322 USA

*E-mail address*: `vicki@mathcs.emory.edu`

Fachbereich Mathematik, Lehrstuhl VI, Universität Dortmund, 44221 Dortmund, Germany

*E-mail address*: `thorsten.woermann@mathematik.uni-dortmund.de`