Real algebraic numbers and polynomial systems of small degree

Ioannis Z. Emiris

National Kapodistrian University of Athens, HELLAS.

Elias P. Tsigaridas

INRIA Sophia-Antipolis Méditerranée, FRANCE.

Abstract

Based on precomputed Sturm-Habicht sequences, discriminants and invariants, we classify, isolate with rational points, and compare the real roots of polynomials of degree up to 4. In particular, we express all isolating points as rational functions of the input polynomial coefficients. Although the roots are algebraic numbers and can be expressed by radicals, such representation involves some roots of complex numbers. This is inefficient and hard to handle in applications to geometric computing and quantifier elimination. We also define rational isolating points between the roots of the quintic. We combine these results with a simple version of Rational Univariate Representation to isolate all common real roots of a bivariate system of rational polynomials of total degree ≤ 2 and to compute the multiplicity of these roots. We present our software within SYNAPS and perform experiments and comparisons with several public-domain implementations. Our package is 2–10 times faster than numerical methods and exact subdivision-based methods, including software with intrinsic filtering.

Key words: algebraic number, bivariate polynomial, quartic, Sturm sequence

1 Introduction

Although the roots of rational polynomials of degree up to 4 can be expressed explicitly with radicals, their computation, even in the case of real roots,

Email addresses: emiris (AT) di.uoa.gr (Ioannis Z. Emiris), Elias.Tsigaridas (AT) sophia.inria.fr (Elias P. Tsigaridas).

requires square and cubic roots of *complex* numbers; this is the famous *casus irreducibilis*. In addition, even if only the smallest (or largest) root is needed, the customary algorithms compute all real roots, see e.g. [17]. Our approach isolates and determines the multiplicity of a specific polynomial root (given by its index), without computing all roots. Another critical issue is that there has been no formula that provides isolating rational points between the real roots of polynomials, in terms of the input coefficients. This problem is settled in this paper for degree ≤ 5 using the floor function for square roots of integers.

In isolating and comparing algebraic numbers, we rely on these isolating points, thus avoiding iterative methods, which depend on separation bounds and, hence, lead to an explosion of the tested quantities. Our approach is based on pre-computed (static) Sturm sequences; essentially, we implement straight-line programs for each computation. In order to further reduce the computational effort, we factorize various quantities by the use of invariants and/or the elements of the Bezoutian matrix. These quantities were computed in an automated way by our MAPLE software, then used in our algorithms.

Lazard in [21] derives necessary conditions for a quartic polynomial to take only positive values and for an ellipse to lie inside a unit circle. In that paper optimal solutions were derived, that could not be obtained by general-purpose algorithms. Inspired by such examples, we enumerate and isolate the real roots of integer polynomials of degree up to 4 and present algorithms for comparing two real algebraic numbers of degree up to 4. Moreover, we derive an efficient algorithm for isolating in rational boxes all common real roots of systems of bivariate integer polynomials of total degree ≤ 2. For each root we also compute its multiplicity.

Our package for algebraic numbers and bivariate polynomial system solving compares favorably with other software. Our implementation is part of the SYNAPS [1] library [25], which is an open source software library for symbolic and numeric computations. Our software implements the algorithms presented in the sequel as well as certain specific optimized functions.

Important applications are in computer-aided geometric design and nonlinear computational geometry, where predicates, which rely on algebraic numbers of small degree, must be decided exactly in all cases, including degeneracies. Efficiency is critical because comparisons and real solving of small degree polynomials lie in the inner loop of several algorithms, most notably those for computing arrangements and Voronoi diagrams of curved objects, see e.g. [7, 8, 11] and related kinetic data-structures [31]. Moreover, isolating points can be used as starting points for iterative algorithms and present independent interest for geometric applications, see e.g. [6, 38]. All of the above are also

[1] `www-sop.inria.fr/galaad/logiciels/synaps/`

basic operations for software libraries supporting geometric computing, such as ESOLID [18], EXACUS [2], and the upcoming algebraic kernel of CGAL [3] [10].

Our work also provides a special-purpose quantifier elimination method for one or two variables, and for parametric polynomial equalities and inequalities of degree ≤ 4; an extension is possible to degree ≤ 9. Our approach extends that of [39], which covers the case of degree 3.

The rest of the paper is organized as follows. The next section overviews relevant existing work and our main contributions. Sec. 3 formalizes Sturm sequences and the representation of algebraic numbers. The following two sections study discrimination systems, and their connection to invariants and root classification, for cubic and quartic polynomials, respectively; in particular, Sec. 5.1 obtains rational isolating points between all real roots of the quartic. Sec. 6 bounds the complexity of comparing algebraic numbers, and applies our tools to real solving bivariate polynomial systems. Sec. 7 sketches our implementation and illustrates it with experiments. Sec 8 extends our results to the quintic. Future work is mentioned throughout the paper.

2 Previous work and our contribution

In quantifier elimination, seminal works optimize low level operations, e.g. [21, 39]. However, by these approaches, the comparison of real algebraic numbers requires multiple Sturm sequences or sign evaluations of polynomials over algebraic numbers. In our approach, we evaluate only one Sturm-Habicht sequence, over two rational numbers.

Rioboo in [28] studies the arithmetic of real algebraic numbers of arbitrary degree with coefficients from a real closed field. This is the only package that can handle non-trivial examples and is implemented in AXIOM. The proposed sign evaluation method is essentially the same as in Thm. 1.

Iterative methods based on the approach of Descartes/Uspensky offer an efficient means for isolating real roots in general [13, 30]. Such a method, based on the Bernstein basis, is implemented in SYNAPS 2.1 [24]. An iterative method, using subdivision and Sturm sequences, has been implemented in [31]. These methods have their source codes available and are tested in Sec. 7. On numerical algorithms for univariate solving we refer the reader to [26], and for real solving to [27] and the references therein.

[2] `www.mpi-sb.mpg.de/projects/EXACUS/`
[3] `www.cgal.org`

LEDA [4] and CORE [5] evaluate expression trees built recursively from integer operations and $\sqrt[k]{}$, and rely on separation bounds. LEDA treats arbitrary algebraic numbers by the diamond operator, based on Descartes/Uspensky iteration and Newton's method [34]. It faces, however, efficiency problems in computing isolating intervals for the roots of polynomials of degree 3 and 4, since Newton's iteration needs special tuning in order to work with interval coefficients. CORE provides a `rootOf` operator for dealing with algebraic numbers using Sturm sequences; it is tested in Sec. 7.

Precomputed quantities for the comparison of quadratic algebraic numbers were used in [8], with static Sturm sequences. In generalizing these methods to higher degree, it is not clear how to determine the (invariant) quantities to be tested in order to minimize the bit complexity. Another major issue has been the isolating points as well as the need of several Sturm sequences. Here we settle both problems.

The basis of our work are the discrimination systems, which are the same as in [40], but they are derived differently; we also correct a small typographical error concerning the quartic. For a polynomial of degree up to 4, we use the quantities involved in its discrimination system not only to determine the number of its roots, but also to compute their multiplicity, to express them as rationals when this is possible, to compute the polynomial's square-free part, and to provide rational points that isolate its roots. The derivation of rational isolating points allows us to compare two real algebraic numbers using a *single* Sturm-Habicht sequence (Thm. 1).

For quadratic numbers and for the efficiency of our implementation see [10]. For algebraic numbers of degree 3 and 4, preliminary results are in [9]. Here we compare our software with the univariate Bernstein solver of SYNAPS [24], RKG of [31], FGB/RS [6] [30], CORE, and NiX, the polynomial library of EXACUS. Our software is faster, even compared to those software packages that have intrinsic filtering. However, our code is slower than the continued fractions implementation of [37], which uses an approach completely different from subdivision. Lastly, we show that the classical closed-form expressions for the roots of cubic and quartic polynomials with large coefficients are very impractical because they involve complex numbers and square and cubic roots.

Solving polynomial systems in a real field is an active area of research. There are several algorithms that tackle this problem, refer for example to [1] and the references therein. To solve quadratic bivariate systems, without assuming generic position, we precompute resultants and Sturm-Habicht sequences in two variables and combine the rational isolating points with a simple version

[4] `www.algorithmic-solutions.com/enleda.htm`
[5] `www.cs.nyu.edu/exact/core`
[6] `http://fgbrs.lip6.fr/salsa/Software/`

of Rational Univariate Representation.

For real-solving of bivariate systems, we experimentally compared our algorithms with the existing solvers in SYNAPS, that is NEWMAC, based on normal forms [23], STH, based on [14], `res`, based on computing the generalized eigenvalues of a Bezoutian matrix [2]. Additionally, we tested against FGB/RS through its MAPLE interface, which uses Gröbner bases and the Rational Univariate Representation [29]. Our implementation is 2–10 times faster, even compared to approximate methods.

3 Sturm Sequences and real algebraic numbers

Sturm (and Sturm-Habicht), e.g. [1, 13, 15, 22], sequences is a well known and useful tool for isolating the roots of any polynomial. For a detailed description the reader may refer to e.g. [1, 13]. In the sequel, $\mathbf{D}$ is a ring, $\mathbf{Q}$ is its fraction field and $\overline{\mathbf{Q}}$ the algebraic closure of $\mathbf{Q}$. Typically $\mathbf{D} = \mathbb{Z}, \mathbf{Q} = \mathbb{Q}$; we shall also consider problems where $\mathbf{D} = \mathbb{R}$. Let $\mathtt{VAR}_{P_1,P_2}(a)$ denote the number of sign variations of the evaluation of the Sturm sequence of polynomials P_1 and P_2, over a.

Theorem 1 *Let $P, Q \in \mathbf{D}[x]$ be relatively prime polynomials and P square-free. If $a < b$ are both non-roots of P, and γ ranges over the roots of P in $[a, b]$, then*

$$\mathtt{VAR}_{P,Q}[a, b] := \mathtt{VAR}_{P,Q}(a) - \mathtt{VAR}_{P,Q}(b) = \sum_{\gamma} \text{sign}\,(P^{'}(\gamma)Q(\gamma)),$$

where $P^{'}$ is the derivative of P. The theorem also holds if we replace Q by the pseudo-remainder of Q divided by P.

Corollary 2 *If $Q = P' \in \mathbf{D}[x]$ and $a < b$ are as above, then the previous theorem counts the number of real roots of P in (a, b).*

The isolating-interval representation of a real algebraic number $\alpha \in \overline{\mathbf{Q}}$ is

$$\alpha \cong (A(X), I),$$

where $A(X) \in \mathbf{D}[X]$ is square-free, $A(\alpha) = 0$, $\alpha \in I = [a, b]$, $a, b, \in \mathbf{Q}$, and A has no other root in I. Let $B(X) \in \mathbf{D}[X]$ and define a real algebraic number $\beta = B(\alpha)$, where $\alpha \cong (A, [a, b])$. By Thm. 1, $\text{sign}(B(\alpha)) = \text{sign}(\mathtt{VAR}_{A,B}[a, b] \cdot A^{'}(\alpha))$.

Let us compare two algebraic numbers $\gamma_1 \cong (P_1(x), I_1)$ and $\gamma_2 \cong (P_2(x), I_2)$ where $I_1 = [a_1, b_1]$, $I_2 = [a_2, b_2]$. Let $J = I_1 \cap I_2$. When $J = \emptyset$, or only one of γ_1

and γ_2 belong to J, we can easily order the two algebraic numbers. These tests are implemented by Thm. 1. If $\gamma_1, \gamma_2 \in J$, then $\gamma_1 \geq \gamma_2 \Leftrightarrow P_2(\gamma_1) \cdot P_2^{'}(\gamma_2) \geq 0$. We easily obtain the sign of $P_2^{'}(\gamma_2)$ and, from Thm. 1, we obtain the sign of $P_2(\gamma_1)$.

In order to test if the two real algebraic numbers are equal, it suffices to test if their gcd (i.e. the last nonzero polynomial in their Sturm-Habicht sequence) changes sign, when evaluated over the endpoints of J. This algorithm is similar to that in [13, 28]. The reader may refer to [5, 13] for details and generalizations of this procedure.

4 The cubic

For a given polynomial we can always compute a system of discriminants, the signs of which determine the number and the multiplicities of the real roots. For the quadratic polynomial the system of discriminants is trivial. For the cubic, it is well known, e.g. [39]. We will present it in the sequel and we will also compute isolating points for the real roots.

Consider the cubic equation

$$f = a\,x^3 + b\,x^2 + c\,x + d, \tag{1}$$

where $f \in \mathbb{R}[x]$ and $a > 0$. We need the following quantities, that are either invariants of the cubic polynomial [3] or elements of the Bézout matrix of f and its derivative f'.

$$\begin{aligned} \Delta_2 &= b^2 - 3\,a\,c, \quad \Delta_3 = c^2 - 3\,b\,d, \\ W &= b\,c - 9\,a\,d, \quad P = 2\,b\Delta_2 - 3\,a\,W. \end{aligned} \tag{2}$$

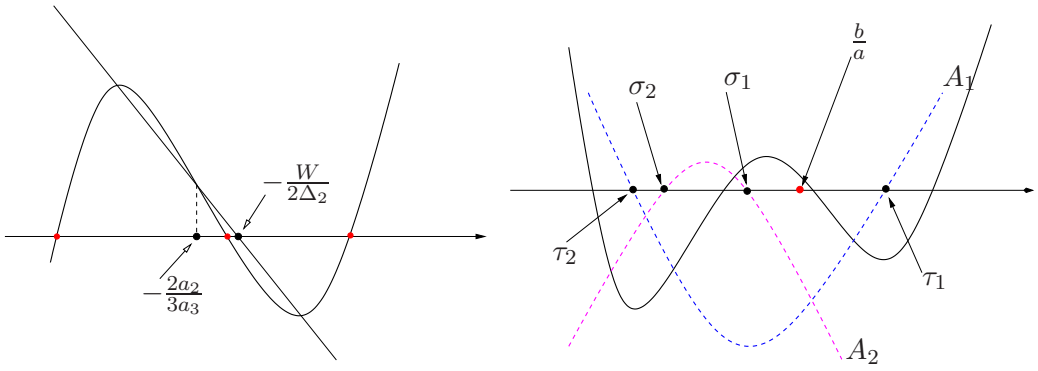
The discriminant of the cubic is

$$\Delta_1 = -\frac{1}{3}(W^2 - 4\,\Delta_2\,\Delta_3).$$

The Sturm-Habicht sequence of f is

$$\mathbf{StHa}(f, f') = (f, f', 2\,\Delta_2 x + W, -3\Delta_1),$$

and the number of real roots of f is $\mathtt{VAR}(\mathbf{StHa}(f; -\infty)) - \mathtt{VAR}(\mathbf{StHa}(f; \infty))$, which means that it depends on the signs of the leading coefficients of the sequence.



(a) Isolator line of the cubic having three real roots; here, $\Delta_1 > 0 \wedge P > 0$.

(b) A quartic and the two isolator polynomials. Recall that $\frac{b}{a}$ is the mean of the roots.

Fig. 1. Isolator polynomials for the cubic and the quartic.

By elementary algebra, we can prove that $\text{sign}(f(-\frac{b}{3a})) = \text{sign}(2b\Delta_2 - 3aW) = \text{sign}(P)$. We denote the roots of f by $\gamma_1, \gamma_2, \gamma_3$ and let $\overline{f} = 9a^2x^2 + 6abx - 2b^2 + 9ac$, which is the quotient of the pseudo-division of f by $3a\,x - b$.

Lemma 3 *Consider f as in expression (1), such that it has three real roots. Then the local minimum, the local maximum, and the saddle point of f are all colinear. The line through them is called* isolating line*, its equation has rational coefficients with respect to the coefficients of f and intersects the $x-$axis at a rational point.*

Proof. The abscissae of the extreme points of the graph of f are the solutions of the quadratic $f' = 0$, which are

$$w_1 = \frac{-b - \sqrt{\Delta_2}}{3\,a} \text{ and } w_2 = \frac{-b + \sqrt{\Delta_2}}{3\,a}.$$

By some elementary computations we can prove that the equation of the isolating line is $2\Delta_2 x + ay + W = 0$. The coordinates of the saddle point are $\left(-\frac{b}{3a}, f(-\frac{b}{3a})\right)$, which satisfies the equation of the isolating line. The isolating line intersects the $x-$axis at point $(-\frac{W}{2\Delta_2}, 0)$.

For the case $P, \Delta_1 > 0$, see Fig. 1(a). □

Theorem 4 *Consider the cubic $f(x) = a\,x^3 + b\,x^2 + c\,x + d$ with three real roots. The rational numbers $\frac{-b}{3a}$ and $-\frac{W}{2\Delta_2}$ isolate the real roots.*

The proof follows from the previous lemma; for details, see [9], where Prop. 6 is applied.

The previous discussion allows us not only to compute the discrimination system of the cubic, but also to compute the isolating interval representation

(1)	$\Delta_1 < 0 \wedge P = 0$	$\gamma_1 \cong [\overline{f}, (-\infty, -\frac{b}{3a})]$ $\gamma_2 = -\frac{b}{3a}$ $\gamma_3 \cong [\overline{f}, (-\frac{b}{3a}, +\infty)]$
(2)	$\Delta_1 < 0 \wedge P < 0$	$\gamma_1 \cong [f, (-\infty, -\frac{W}{2\Delta_2})]$ $\gamma_2 \cong [f, (-\frac{W}{2\Delta_2}, -\frac{b}{3a})]$ $\gamma_3 \cong [f, (-\frac{b}{3a}, +\infty)]$
(3)	$\Delta_1 < 0 \wedge P > 0$	$\gamma_1 \cong [f, (-\infty, -\frac{b}{3a})]$ $\gamma_2 \cong [f, (-\frac{b}{3a}, -\frac{W}{2\Delta_2})]$ $\gamma_3 \cong [f, (-\frac{W}{2\Delta_2}, +\infty)]$
(4)	$\Delta_1 > 0 \wedge d = 0$	$\gamma_1 = 0$
(5)	$\Delta_1 > 0 \wedge d < 0$	$\gamma_1 \cong [f, (0, +\infty)]$
(6)	$\Delta_1 > 0 \wedge d > 0$	$\gamma_1 \cong [f, (-\infty, 0)]$
(7)	$\Delta_1 = 0 \wedge \Delta_2 \neq 0$	$\gamma_1 = \min(\frac{-W}{2\Delta_2}, \frac{b\Delta_2 - aW}{a\Delta_2})$ $\gamma_2 = \max(\frac{-W}{2\Delta_2}, \frac{b\Delta_2 - aW}{a\Delta_2})$
(8)	$\Delta_1 = 0 \wedge \Delta_2 = 0$	$\gamma_1 = -\frac{b}{3a}$

Table 1
Discrimination system and isolating points of the cubic.

and the multiplicities of its real roots. This is summarized in table 1. The cases (1), (2) and (3) correspond to a cubic with three distinct real roots. Cases (4), (5) and (6) correspond to cubics with one real root. In this case we can easily isolate the root using the sign of the trailing coefficient. Case (7) corresponds to cubics with two distinct real roots, meaning that one of them is a double root; then, the roots are rational functions in the coefficients of the cubic. The double root is always equal to $-\frac{W}{2\Delta_2}$, whereas $-\frac{W}{2\Delta_2}$ cannot be a root of a cubic with three distinct real roots, since $\text{sign}(f(-\frac{W}{2\Delta_2})) = \text{sign}(P\Delta_1) \neq 0$. Finally, the last case corresponds to cubics with one real root of multiplicity 3.

5 The quartic

We study the quartic and determine its roots as rationals, if this is possible, otherwise we provide isolating rationals between every pair of real roots. Consider the quartic polynomial equation, where $a, b, c, d, e \in \mathbb{Z}$ and $a > 0$:

$$f(x) = ax^4 - 4bx^3 + 6cx^2 - 4dx + e. \tag{3}$$

We study the quartic using Sturm-Habicht sequences and the Bézout matrix,

while [40] used a resultant-like matrix. We use invariants to characterize the different cases; for background see [3, 33]. We consider the rational invariants of f, i.e. the invariants in $GL(2,\mathbb{Q})$. They form a graded ring generated by $A = W_3 + 3\Delta_3$ and $B = -dW_1 - e\Delta_2 - c\Delta_3$ [3], where the W_i, Δ_i are defined in expression (4). Every other invariant is isobaric in A, B, hence homogeneous in the coefficients of the quartic. Let $\Delta_1 = A^3 - 27B^2$ be the *discriminant*. The semivariants (i.e. the leading coefficients of the covariants) are A, B and

$$\Delta_2 = b^2 - ac,\ R = aW_1 + 2b\Delta_2,\ Q = 12\Delta_2^2 - a^2 A.$$

We also define the following quantities, which are not necessarily invariants but they are elements of the Bezoutian matrix of f, f'.

$$\begin{array}{lll} \Delta_3 = c^2 - bd, & W_1 = ad - bc, & T = -9W_1^2 + 27\Delta_2\Delta_3 - 3W_3\Delta_2, \\ \Delta_4 = d^2 - ce, & W_2 = be - cd, & T_1 = -W_3\Delta_2 - 3\,W_1^2 + 9\,\Delta_2\Delta_3, \\ & W_3 = ae - bd, & T_2 = AW_1 - 9\,bB. \end{array} \tag{4}$$

In [40] there is a small typographical error in defining T. Since our discrimination system is based on Sturm-Habicht sequences and, essentially, on the principal subresultant coefficients, we use the Bezoutian matrix to compute them symbolically.

Proposition 5 *[40] Let $f(x)$ be as in expression (3) and consider the quantities defined above. The following table gives the real roots and their multiplicities. In case (2) there are 4 complex roots, while in case (8) there are two complex double roots:*

(1) $\Delta_1 > 0 \wedge T > 0 \wedge \Delta_2 > 0$	$\{1,1,1,1\}$
(2) $\Delta_1 > 0 \wedge (T \leq 0 \vee \Delta_2 \leq 0)$	$\{\}$
(3) $\Delta_1 < 0$	$\{1,1\}$
(4) $\Delta_1 = 0 \wedge T > 0$	$\{2,1,1\}$
(5) $\Delta_1 = 0 \wedge T < 0$	$\{2\}$
(6) $\Delta_1 = 0 \wedge T = 0 \wedge \Delta_2 > 0 \wedge R = 0$	$\{2,2\}$
(7) $\Delta_1 = 0 \wedge T = 0 \wedge \Delta_2 > 0 \wedge R \neq 0$	$\{3,1\}$
(8) $\Delta_1 = 0 \wedge T = 0 \wedge \Delta_2 < 0$	$\{\}$
(9) $\Delta_1 = 0 \wedge T = 0 \wedge \Delta_2 = 0$	$\{4\}$

In correspondence with prop. 5, we give the rational or quadratic roots of equation (3), when they exist, obtained in a straightforward manner from (pseudo-)remainders in the Sturm sequence. Then, we derive rational isolating points for the other cases.

(1) $\{1,1,1,1\}$
In Thm. 9, we specify 3 rational isolating points.
(3) $\{1,1\}$
In Cor. 10, we specify a rational isolating point.
(4) $\{2,1,1\}$
The double root is rational since it is the only root of $\gcd(f,f')$ and if $\Delta_2 = 0$ then its value is $\frac{W_3}{3W_1}$, otherwise it is $-\frac{T_2}{3T_1}$, with quantities defined in expression (4). The other two roots are the roots of the quadratic polynomial $a^2x^2 - 2abx + 6ac - 5b^2$.
(5) $\{2\}$
As in the previous case, the double root is rational. If $\Delta_2 = 0$, then the root equals $\frac{W_3}{3W_1}$, otherwise it equals $-\frac{T_2}{3T_1}$.
(6) $\{2,2\}$
The roots are the smallest and largest root of the derivative i.e. a cubic. Alternatively, we express them as the roots of $3\Delta_2 x^2 + 3W_1 x - W_3$.
(7) $\{3,1\}$
The triple root is $-\frac{W_1}{2\Delta_2}$ and the simple root is $\frac{3\,aW_1+8\,b\Delta_2}{2\,a\Delta_2}$.
(9) $\{4\}$
The real root is $\frac{b}{a} \in \mathbb{Q}$.

For the cases above, where rational points are not available from the Sturm sequence, we shall establish rational isolating points in the sequel. First, let us state a useful result.

Proposition 6 [35] *Given a polynomial $P(x)$, with any two adjacent real roots denoted by $\gamma_1 < \gamma_2$, and any two other polynomials $B(x)$, $C(x)$, let $A(x) := B(x)P'(x) + C(x)P(x)$, where P' is the derivative of P. Then $A(x)$ and $B(x)$ are called* isolating polynomials *because at least one of them has at least one real root in the closed interval $[\gamma_1, \gamma_2]$. In addition, it is possible to have* $\deg A + \deg B \leq \deg P - 1$.

Hence, we isolate the real roots of any quartic, in the worst case, by two quadratic numbers and a rational. In the sequel, we use these to obtain *rational* points.

Lemma 7 *Given $a_1, a_2 \in \mathbb{Z}, \delta_1, \delta_2 \in \mathbb{N}$, let $\tau > \sigma > 0$ be defined below. Then,*

it is possible to determine $r \in \mathbb{Q}$ *as a function of* $a_1, a_2, \delta_1, \delta_2$*, such that*

$$\sigma = a_1 + \sqrt{\delta_1} \leq r \leq a_2 + \sqrt{\delta_2} = \tau.$$

Proof. Now, σ is a root of the polynomial $h_1(x) = x^2 - 2\,a_1 x - \delta_1 + {a_1}^2$, while τ is a root of $h_2(x) = x^2 - 2\,a_2 x - \delta_2 + {a_2}^2$. We consider a resultant w.r.t. y:

$$h(x) = \mathrm{Res}(h_1(y), h_2(x+y)) = x^4 + n_3 x^3 + n_2 x^2 + n_1 x + n_0, \qquad (5)$$

where $n_3 = 4\,a_1 - 4\,a_2$,
$n_2 = -12\,a_2 a_1 + 6\,{a_2}^2 - 2\,\delta_1 - 2\,\delta_2 + 6\,{a_1}^2$,
$n_1 = -4\,(a_1 - a_2)\,(-{a_1}^2 + 2\,a_2 a_1 + \delta_1 + \delta_2 - {a_2}^2)$,
$n_0 = 4\,a_2 a_1 \delta_1 + 4\,a_2 a_1 \delta_2 + {\delta_1}^2 - 2\,\delta_1 \delta_2 - 2\,\delta_1 {a_1}^2 - 2\,\delta_1 {a_2}^2 + {\delta_2}^2 - 2\,\delta_2 {a_1}^2 - 2\,\delta_2 {a_2}^2 + 6\,{a_1}^2 {a_2}^2 - 4\,a_2 {a_1}^3 - 4\,{a_2}^3 a_1 + {a_1}^4 + {a_2}^4$.

Polynomial $h(x)$ has $\tau - \sigma > 0$ as one of its (four) real roots. We consider any of the possible lower bounds $k > 0$ on the positive roots of h, see [16, 19, 36, 41]. Independently of the precise value of k, the following holds:

$$a_1 + \sqrt{\delta_1} < k + a_1 + \sqrt{\delta_1} < a_2 + \sqrt{\delta_2}.$$

If $k \geq 1$, then we set

$$r = k + a_1 + \left\lfloor \sqrt{\delta_1} \right\rfloor,$$

which satisfies the inequalities because $\left\lfloor \sqrt{\delta_1} \right\rfloor + k \geq \sqrt{\delta_1}$. In this case, we could also choose $r = 1 + a_1 + \left\lfloor \sqrt{\delta_1} \right\rfloor$.

If $k < 1$, then $k = \frac{\lambda}{\mu}$ for integers $1 \leq \lambda < \mu$, and it holds that

$$\mu\sigma = \mu a_1 + \mu\sqrt{\delta_1} < \lambda + \mu a_1 + \mu\sqrt{\delta_1} < \mu a_2 + \mu\sqrt{\delta_2} = \mu\tau. \qquad (6)$$

Then, we choose

$$r = \frac{\lambda}{\mu} + a_1 + \frac{\left\lfloor \mu\sqrt{\delta_1} \right\rfloor}{\mu},$$

because $r < \tau \Leftrightarrow \mu r < \mu\tau$, which follows from the right inequality (6). Moreover, $\mu r \geq \lambda + \mu a_1 + \mu\sqrt{\delta_1} - 1 \geq \mu\sigma$. □

To decrease the bit-size of the numbers involved in the definition of r, we can use, instead of k, the simplest rational in $(0, k]$. This can be computed using an algorithm due to Gosper [20, Sec. 4.5.3, ex. 39]. We can extend the construction of Lem. 7 to compute a a number between two real algebraic numbers, as a rational function of the coefficients of the of the polynomials that define them [12].

Moreover, we can always compute a lower bound on the positive roots of equation (5) which lies in $(0,1)$ and thus unify the two cases in the proof of Lem. 7. This is done in the next corollary.

Corollary 8 *Given $a_1, a_2 \in \mathbb{Z}, \delta_1, \delta_2 \in \mathbb{N}$, let $\mu = 1 + \max\{|n_1|, |n_2|, |n_3|\}$, where n_i are the coefficients of equation (5). Then, it is possible to separate the algebraic numbers σ, τ, defined below, where $\sigma\tau > 0$, by some $r \in \mathbb{Q}$ as follows:*

$$\sigma = a_1 + \sqrt{\delta_1} \leq r = \frac{1}{\mu} + a_1 + \frac{\left\lfloor \mu\sqrt{\delta_1} \right\rfloor}{\mu} \leq a_2 \pm \sqrt{\delta_2} = \tau,$$

$$\sigma = a_1 - \sqrt{\delta_1} \leq r = \frac{1}{\mu} + a_1 + \frac{\left\lfloor -\mu\sqrt{\delta_1} \right\rfloor}{\mu} \leq a_2 \pm \sqrt{\delta_2} = \tau.$$

Proof. Take $\sigma = a_1 + \sqrt{\delta_1}$ and both $\tau > \sigma > 0$, as in Lem. 7. Based on Cauchy's lower bound on the roots' absolute value [41, Lem.6.7], we set $k = 1/(1 + \max\{|n_1|, |n_2|, |n_3|, 1\})$, where $\lambda = 1$ in the notation of the proof of Lem. 7. Since $n_i \in \mathbb{Z}$, the maximum can be taken over $\{|n_1|, |n_2|, |n_3|\}$. Exactly the same approach computes r when $\tau = a_2 - \sqrt{\delta_2}$.

When $\sigma = a_1 - \sqrt{\delta_1}$ and $\tau > \sigma > 0$, we use again the quartic $h(x)$ in (5) to compute the lower bound, since its roots are the possible differences of the two algebraic numbers. Notice that $h(x)$ is symmetric w.r.t. indices $1, 2$.

If both algebraic numbers are negative, i.e. $\sigma < \tau < 0$, the same proof applies. □

Of course, if σ, τ have opposite signs, then we pick $r = 0$.

Suppose the quartic in equation (3) has 4 simple real roots. Let us reduce the number of parameters, i.e. some of the coefficients of the quartic. In particular, we can set the coefficient of x^3 to zero, by applying $x \mapsto x/a$ in equation (3), then multiply the resulting expression by a^3; recall that $a > 0$. W.l.o.g., after renaming the coefficients, we can write the quartic as

$$f(x) = x^4 + cx^2 + dx + e, \tag{7}$$

where $c, d, e \in \mathbb{Z}$. Since equation (7) has 4 simple real roots, Descartes' rule of signs counts exactly the positive (and negative) real roots. This rule implies the following sign conditions, according to the quartic's roots:

- If f has two negative and two positive roots, then $c < 0$, $e > 0$ and d can have any sign condition, including 0.
- If f has 3 negative and one positive root, then $c < 0$, $d < 0, e < 0$.

- If f has one negative and 3 positive roots, then $c < 0$, $d > 0$, $e < 0$.

We apply Prop. 6 using $B_1(x) = -x$ and $C_1(x) = 4$, then a first isolating polynomial is

$$A_1(x) = 2\,cx^2 + 3\,dx + 4\,e.$$

It has discriminant δ_1 and roots σ_1, σ_2:

$$\frac{-3d - \sqrt{\delta_1}}{4c} = \sigma_1 < \sigma_2 = \frac{-3d + \sqrt{\delta_1}}{4c},\ \delta_1 = 9\,d^2 - 32\,ce.$$

Using $B_2(x) = d\,x + 4e$ and $C_2(x) = -4d$, the second isolating polynomial becomes

$$A_2(x) = (16\,ex^2 - 2\,dcx - 3\,d^2 + 8\,ce)x.$$

It has discriminant δ_2; besides zero, it has real roots τ_1, τ_2:

$$\frac{2dc - \sqrt{\delta_2}}{32e} = \tau_1 < \tau_2 = \frac{2dc + \sqrt{\delta_2}}{32e},\ \delta_2 = 192\,ed^2 - 512\,e^2c + 4\,c^2d^2.$$

Either one or two pairs of the roots of equation (7) are separated by a rational root of $B_i,\ i = 1, 2$. Consider a pair separated by real, non-rational roots of $A_1(x)$ and $A_2(x)$; we shall compute a rational separating point for this pair.

If one of δ_1 and δ_2 is a square, then one of these roots is rational. Assuming δ_1, δ_2 are not squares, we show that the roots of $A_1(x), A_2(x)$ are different. If not, $\sigma_i = \tau_j$, for some $i, j \in \{1, 2\}$, which implies $\delta_1 = \delta_2$, and $-24de = 2dc^2$ i.e. $e = -c^2/12$. Since $c, e \in \mathbb{R}$, it follows $c = e = 0$, then equation (7) has 0 as root and we have reduced our problem to a simpler question.

Now we consider positive $\sigma_i \neq \tau_j$, for some $i, j \in \{1, 2\}$. To compute a rational number between them, i.e. between $(-24de \pm 8e\sqrt{\delta_1})/(32ce)$ and $(2dc^2 \pm c\sqrt{\delta_2})/(32ce)$, it suffices to compute a rational between $-24de \pm 8e\sqrt{\delta_1}$ and $2dc^2 \pm c\sqrt{\delta_2}$ and divide it by $32ce$.

Following Cor. 8, and the proof of Lem. 7, we let

$$h(x) = x^4 + n_3x^3 + n_2x^2 + n_1x + n_0,$$

where

$$\begin{aligned}
n_0 &= -16384\,c^2e^3\left(27\,d^4 - 256\,e^3 - 16\,ec^4 + 128\,c^2e^2 + 4\,c^3d^2 - 144\,ecd^2\right),\\
n_1 &= 256\,dce\left(12\,e + c^2\right)\left(-16\,ec^2 + 3\,d^2c - 64\,e^2\right),\\
n_2 &= 2304\,d^2e^2 + 16\,d^2c^4 + 192\,d^2ec^2 + 4096\,e^3c + 1024\,c^3e^2\\
n_3 &= -8\,d\left(12\,e + c^2\right),
\end{aligned}$$

Then we set $\mu = 1 + \max\{|n_1|, |n_2|, |n_3|\}$, and apply Cor. 8, under the assumption $\sigma_i < \tau_j,\ i, j \in \{1, 2\}$. This yields two candidate rational isolating points.

In the case that $\tau_j < \sigma_i$, Cor. 8 yields another two rational points. The roots of $B_1(x)$ and $B_2(x)$ yield 0 and $-\frac{4e}{d}$ as candidates for isolating points. This proves the main theorem.

Theorem 9 *Consider a quartic f as in (7), with 4 distinct real roots. At least three of the following 6 rational points:*

$$\left\{0, -\frac{4e}{d}, \frac{1 - 24de\mu + \left\lfloor \pm 8e\mu\sqrt{\delta_1} \right\rfloor}{32ce\mu}, \frac{1 + 2dc^2\mu + \left\lfloor \pm c\mu\sqrt{\delta_2} \right\rfloor}{32ce\mu}\right\},$$

isolate the real roots of the quartic, where $\mu = 1 + \max\{|n_1|, |n_2|, |n_3|\}$ and the $n_i \in \mathbb{Z}$ are defined above. One way of deciding the three isolating points is by sorting them and evaluating f on them.

We were also able to prove the previous theorem using the RAGlib library [32].

Corollary 10 *If the quartic f in (7) has two simple real roots, then one of the rational numbers in the previous theorem is an isolating point.*

Remark 11 *If f has two real roots, and since the leading coefficient is positive, then any point from Thm. 9 with negative value over f serves as isolating point.*

If f has real roots $\gamma_1 < 0 < \gamma_2 < \gamma_3 < \gamma_4$, then 0 is an isolating point. Notice that f is positive between γ_2 and γ_3, and negative between γ_3 and γ_4. The smallest positive rational from Thm. 9, whose evaluation over f is positive, isolates γ_2 and γ_3. The next greater rational whose evaluation over f is negative isolates γ_3 and γ_4.

If f has real roots $\gamma_1 < \gamma_2 < 0 < \gamma_3 < \gamma_4$, again 0 is an isolating point. Notice that f is negative over (γ_1, γ_2) as well as over (γ_3, γ_4). The positive (resp. negative) rationals from Thm. 9 where f becomes negative isolate γ_3 and γ_4, resp. γ_1 and γ_2.

Example 12 *Consider the quartic*

$$x^4 - 12\,x^2 - 20\,x - 8,$$

that has 4 real roots, $\gamma_1 < \gamma_2 < \gamma_3 < 0 < \gamma_4$, the approximations of which are -2, -1.525427561, -0.6308976138 and 4.156325175, respectively.

We have: $A_2(x) = -128\,x^3 - 480\,x^2 - 432\,x$, with real roots $-\frac{9}{4}, -\frac{3}{2}$, and 0 (approximately, -2.25, -1.5, and 0). Moreover, $B_2(x) = -20\,x - 32$, with root $-\frac{8}{5} = -1.6$.

Since all roots of $A_2(x), B_2(x)$ are rationals, it is enough to use them as isolating points for the roots of f, thus $\gamma_1 < -\frac{8}{5} < \gamma_2 < -\frac{3}{2} < \gamma_3 < 0 < \gamma_4$.

Example 13 *Consider the quartic*

$$f(x) = x^4 - 15\,x^2 + 20\,x - 4,$$

that has 4 real roots, $\gamma_1 < 0 < \gamma_2 < \gamma_3 < \gamma_4$*, the approximations of which are* -4.439*,* 0.244*,* 1.2504 *and* 2.944*, respectively.* $A_1(x) = -30\,x^2 + 60\,x - 16$*, with* $(15 \pm \sqrt{105})/15$ *(or* 0.316 *and* 1.683*) as real roots.* $A_2(x) = -64\,x^3 + 600\,x^2 - 720\,x$*, with real roots* $0, (75 \pm 3\,\sqrt{305})/16$ *(or* 0*,* 1.412 *and* 7.962*). The graph of* f *and the two isolator polynomials,* $A_1(x)$ *and* $A_2(x)$*, in the positive* x *semi-axis is shown in Fig. 2.*

The auxiliary quartic of (5) is

$$h(x) = x^4 - 28320\,x^3 + 218261760\,x^2 - 251427225600\,x - 193182931353600,$$

and a lower bound on its positive real roots is $k = 1/(1+251427225600)$*, where* $\mu = 251427225600$*. The 6 rationals of Thm. 9, in increasing order, are*

$$\left\{ 0,\ \tfrac{152965885753921}{482740273153920},\ \tfrac{4}{5},\ \tfrac{682089450409001}{482740273153920},\ \tfrac{812514660553921}{482740273153920},\ \tfrac{1281200203469667}{160913424384640} \right\},$$

and their approximations are $\{0, 0.3168, 0.8, 1.4129, 1.6831, 7.9620\}$*. The evaluation of* f *over them gives the following signs* $\{-, +, +, -, -, +\}$*. Thus, the second point* $\frac{152965885753921}{482740273153920} \approx 0.3168$*, separates* γ_2 *and* γ_3*. Similarly, the fourth point* $\frac{682089450409001}{482740273153920} \approx= 1.419$*, separates* γ_3 *and* γ_4*.*

Notice that both rationals following 0 *separate* γ_2 *and* γ_3*, since* f *is positive over them. Similarly, the two rationals after them, where* f *is negative, separate* γ_3 *and* γ_4*.*

Remark 14 *In our experiments, we observed that among the 6 rationals* $\left\{0, -4e/d, \frac{-3d \pm \lceil\sqrt{\delta_1}\rceil}{4c}, \frac{-3d \pm \lfloor\sqrt{\delta_1}\rfloor}{4c}\right\}$ *one can always find all isolating points, but we are not able to provide a formal proof.*

6 Comparison and real solution

Comparison of algebraic numbers. We consider the problem of comparing real algebraic numbers. In what follows, $\mathcal{O}$ and $\mathcal{O}_B$ denote respectively asymptotic arithmetic and bit complexity bounds, whereas the notation $\widetilde{\mathcal{O}}$ and $\widetilde{\mathcal{O}}_B$ is used when we are ignoring polylogarithmic factors.

Using the discussion as well as the isolating points computed in the previous section, we arrive at the following:

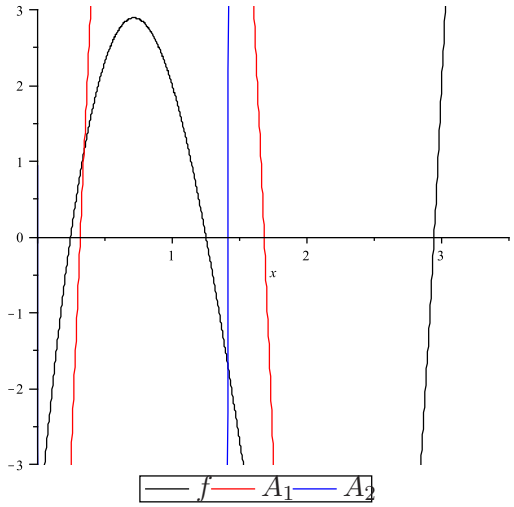


Fig. 2. The quartic $x^4 - 15\,x^2 + 20\,x - 4$ and two isolator polynomials.

Theorem 15 *Given an integer polynomial of degree $d \leq 4$ and maximum coefficient bit-size τ, we can isolate its real roots and compute their multiplicities in $\widetilde{\mathcal{O}}(1)$ or $\widetilde{\mathcal{O}}_B(\tau)$.*

This theorem improves the general-purpose real root isolation algorithms by a factor of τ. The general bounds are $\widetilde{\mathcal{O}}(d^2\tau)$ and $\widetilde{\mathcal{O}}_B(d^4\tau^2)$, cf. e.g. [13]. In order to compute the floor of the square root of a non-negative integer, one may employ the bisection method [4], with complexity logarithmic in the bit-size of the integer.

We measure the complexity of an algorithm by the maximum algebraic degree of the tested quantities, in terms of the input polynomials' coefficients. Take two univariate polynomials of degree d with symbolic coefficients; the degree of all coefficients in their (symbolic) resultant is $2\,d$ in the input polynomial coefficients. A lower bound on the complexity of comparing two roots of these polynomials is the algebraic degree of their resultant, namely $2d$. It is an open question whether a better lower bound exists.

For quadratic polynomials, there is a straightforward algorithm for the comparison of two quadratic algebraic numbers, with maximum algebraic degree 4, hence optimal, e.g. [8]. Next, we examine cubic and quartic polynomials. Our algorithms cover all degeneracies, including the case that a polynomial drops degree.

Theorem 16 *Given polynomials of degree ≤ 4, there is an algorithm that compares any two of their real roots using a constant number of arithmetic operations. For two cubics, the tested quantities are of algebraic degree 6 in the input coefficients, hence optimal; for quartics, the degrees are between 8 and 14.*

Proof. We give an overview of the method and specify certain details for the case of two quartics. For further information, the reader may refer to [9]. The complete proof for the cubic can be found in the Appendix.

In order to compare two polynomial roots, we use the algorithm at the end of Sec. 3. The crucial step is to compute isolating intervals for the real roots. For quartics, the results of Sec. 5 yield quadratic algebraic numbers as endpoints, if one does not wish to rely on the floor operation of square roots. So we have to compute the sign of the Sturm-Habicht sequence over a quadratic algebraic number, by the algorithm from Sec. 3. Hence we have sketched an algorithm for comparison.

As for the maximum algebraic degree involved, we focus on quartics and consider the hardest case. This is the determination of the sign of the linear polynomial in the Sturm-Habicht sequence over a quadratic algebraic number. The algebraic degree of the coefficients of the linear polynomial is 6 in the input coefficients. In the worst case, the evaluation involves testing the signs of quantities of algebraic degree 14 [8]. □

For quartics, our algorithm requires up to 172 additions and multiplications; the precise algebraic degree depends on the degree of the isolating points. In particular, the algorithm has algebraic degree 8 or 9 when comparing roots of square-free quartics.

Fig. A.1 in the Appendix shows the whole evaluation procedure to compare the two largest roots of two cubic polynomials. The number down and left of each rhombus denotes the maximum algebraic degree of the expression, while the number in parentheses denotes the minimum. The number down and right of each rhombus denotes the maximum number of operations needed to evaluate the expression, while the one in parentheses denotes the minimum.

Bivariate system solving. We now apply our results to bivariate system solving. Consider the system $f_1 = f_2 = 0$, where $f_1, f_2 \in \mathbb{Z}[x, y]$ are bivariate polynomials of total degree at most two. In what follows, we assume that the system is $0-$dimensional (we can detect that it is not so, because then some resultant that we compute below, would vanish). The real solutions of the system are points in $\overline{\mathbb{Q}}^2$.

First, we compute the resultants R_x, R_y of f_1, f_2 by eliminating y and x respectively, thus obtaining degree-4 polynomials in x and y. We isolate the real solutions of R_x, R_y, define a grid of boxes, where the common roots of f_1 and f_2 are located. The grid has 1 to 4 rows and 1 to 4 columns in $\mathbb{R}^2$. It remains to decide, for the boxes, whether they are empty and, if not, whether they

contain a simple or multiple root.

The hardest cases are when R_x and R_y do not have multiple roots; otherwise, the roots are rational or quadratic algebraic numbers, as shown in the previous section. In this case, f_1 and f_2 are in generic position (the intersection points have distinct $x-$coordinates) and thus we can solve the system using a simple version of Rational Univariate Representation, e.g. [14, 29]. Now the $y-$coordinate is the solution of the first subresultant, which is univariate w.r.t. y, and its coefficients are univariate polynomials evaluated over the solutions of R_x, that is $\gamma_y = F(\gamma_x) = \frac{-B(\gamma_x)}{A(\gamma_x)}$. This is an implicit representation. To have an isolating interval representation, we use the following trick. Since we have the solutions of R_y and their isolating points, we find the isolating interval at which each $F(\gamma_x)$ lies. This can be done by testing the signs of univariate polynomials evaluated over algebraic numbers.

The computation of the resultants is not costly, since the degree is small. Unlike e.g. [7], where the boxes cannot contain any critical points of f_1 and f_2, in our algorithm we make no such an assumption. Hence there is no need to refine them. Our approach can be extended to computing the intersection points of bivariate polynomials of arbitrary degree, provided that we obtain isolating points for the roots of the two resultants, statically or dynamically (see [5]).

7 Implementation and experimental results

We have implemented a software package, as a part of library SYNAPS (v2.1) [25], for dealing with algebraic numbers and bivariate polynomial system solving, which is optimized for small degree. Our implementation is generic in the sense that it can be used with any number type and any polynomial class that supports elementary operations and evaluations and can handle all degenerate cases. We developed `C++` code for real solving, comparison and sign determination functions. In what follows `root_of` is a class that represents real algebraic numbers, computed as roots of polynomials, and `UPoly` and `BPoly` are classes for univariate and multivariate polynomials. All classes are parametrized by the ring number type (`RT`); the reader may refer to the documentation of SYNAPS for more details. We provide the following functionalities:

- `Seq<root_of<RT> > solve(UPoly<RT>` f`)`: Solves a univariate polynomial.
- `int compare(root_of<RT>` α`, root_of<RT>` β`)`: Compares two algebraic numbers. For degree up to 4 we use static Sturm sequences. For higher degree we use Sturm-Habicht sequences, computed on the fly.
- `int sign_at(UPoly<RT>` f`, root_of<RT>` α`)`: Computes the sign of a univariate polynomial evaluated over an algebraic number.

- `int sign_at(BPoly<RT>` f`, root_of<RT>` γ_x`, root_of<RT>` γ_y`)`: Computes the sign of a bivariate polynomial evaluated over two real algebraic numbers. We use cascaded Sturm-Habicht sequences.
- `Seq < pair<root_of<RT> > > solve(BPoly<RT>` f_1`, BPoly<RT>` f_2`)`: Computes the real solutions of a bivariate polynomial system.

We performed all tests on a 2.6GHz Pentium with 512MB memory, running Linux, with kernel version 2.6.10. We compiled the programs with `g++`, v.3.3.5, with option `-O3`. We mark our implementation by S^3 which stands for *Static Sturm Sequences* and use "f" to denote the filtered version. The other methods are described in Sec. 2.

Table 2
Univariate root comparison

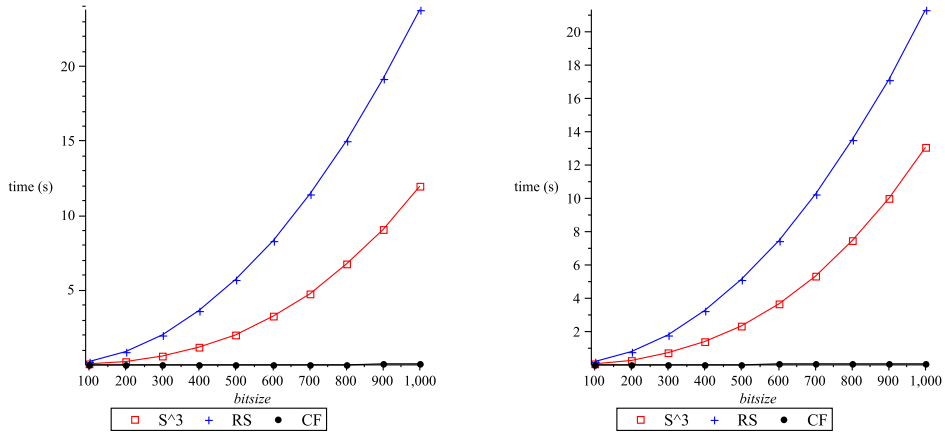
msec	A	B	C	D
f-S^3	0.142	0.153	0.150	0.177
S^3	0.291	0.320	0.142	0.112
RS	5.240	6.320	4.930	5.180
SYNAPS	1.058	1.011	0.717	1.850
CORE	3.050	3.520	2.240	1.470
RKG	2.287	2.973	2.212	1.595
NiX	0.358	0.362	0.215	0.377

Table 3
Bivariate real-solving

msec	A	B
f-S^3	0.17	0.18
S^3	0.14	0.54
FGB/RS	6.40	6.90
STH	0.51	0.57
`res`	0.36	-
NEWMAC	3.19	3.26

Univariate polynomials. We performed 4 kinds of tests concerning the comparison of real algebraic numbers of degree up to 4. For every polynomial we "computed" all its real roots, with every package since, except for our code (S^3), no other package can compute a specific root only. We repeated each test 10000 times. The results are in table 2. Column A refers to polynomials with exactly 4 distinct rational roots in $[-1, 1]$, the bit size of the coefficients is 40 bits. Column B refers to random polynomials, produced by interpolation in $[-1, 1] \times [-1, 1]$, the bit size of the coefficients is 90 bits. Column C refers to Mignotte polynomials, of the form $a(x^4 - 2(Lx - 1)^2)$, where the bit size of a and L is 40 bits. Column D refers to degenerate polynomials, that is polynomials with at least one multiple root. All roots are random rationals in $[-1, 1]$ and the bit size of the coefficients is 30 bits.

RKG is the package of [31], NiX is the polynomial library of EXACUS that has intristic filtering, since it is based on LEDA, CORE is version 1.7 and RS [30] is used through its MAPLE interface. SYNAPS refers to the algorithm of [24] in SYNAPS. We have also tested MAPLE and AXIOM, but we do not show



(a) Cubic polynomials. (b) Quartic polynomials.

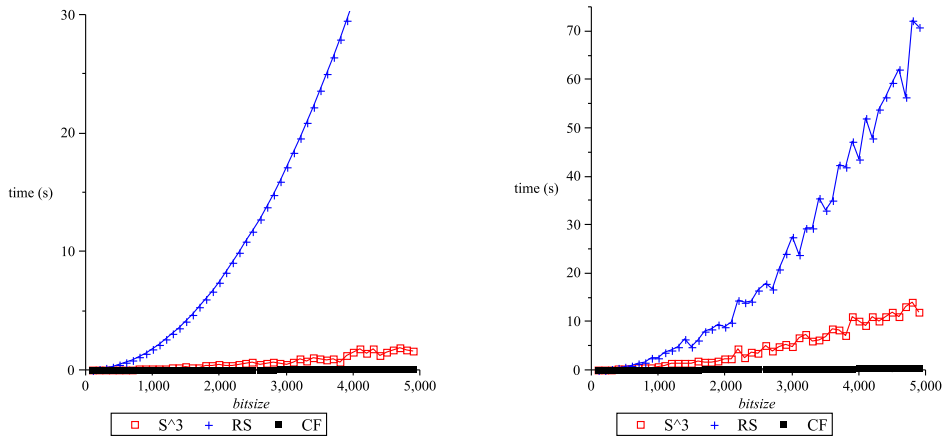
Fig. 3. Experiments with Mignotte polynomials.

their timings here, since they are too slow, see [9] for details. S^3 is our code implemented in SYNAPS 2.1 and f-S^3, is our code using the filtered number type `Lazy_exact_nt` from CGAL, i.e. a number type that initially performs all the operations with doubles and, if this fails, it switches to exact arithmetic.

SYNAPS has some problems when the roots are endpoints of a subdivision, while CORE has some problems with subdivision, since Newton's method is used for refinement. By considering table 2, S^3 is clearly faster than CORE, SYNAPS and RKG, even without filtering. Against filtered methods (NIX), S^3 is still faster. Special attention must be paid to column D, where our code is significantly faster. The slow performance of RS is partly due to the fact that we use its MAPLE interface in order to call the appropriate functions, since the source code is not available. Now consider the first row of table 2. The adoption of a filtered number type improves the running times in most cases, otherwise it leaves them essentially unchanged. Column B represents the hardest case, due to the bit-size of the coefficients, but even in this column f-S^3 is two times faster than the next fastest software NIX, which has intrinsic filtering.

We also performed experiments on real solving polynomials of variable bit-size in order to further test the efficiency of our specialized algorithms and to illustrate the almost-linear behavior of our algorithms with respect to bit-size. We tested against RS, using its function `rs_time()` in order to measure it times, and against a very efficient, recent implementation of the continued fractions algorithm (CF) [37]. The latter algorithm computes successively closer approximations of the root based on the continued fractions representation, but does not use any polynomial remainder sequence.

First we tested our algorithms versus cubic and quartic Mignotte polynomials of the form $x^d - 2(ax-1)^2$, where $d \in \{1,2\}$ and $a = 2^n - \frac{1}{2^n}$, and $n = \{100, 200, \ldots, 1000\}$. The results can be found in Fig. 3. Mignotte polynomials



(a) Cubic polynomials. (b) Quartic polynomials.

Fig. 4. Experiments with random polynomials.

are "easy" polynomials for the CF implementation.

Then we tested random cubic and quartic polynomials obtained by interpolation in the box $\left[-\frac{1}{2^n}, \frac{1}{2^n}\right] \times \left[-\frac{1}{2^n}, \frac{1}{2^n}\right]$, where $n \in \{100, 200, \ldots, 4900\}$. The points were chosen such that the cubics (resp. quartics) have three (resp. at least two) real roots. The results can be found in Fig. 4. Even though CF is the fastest implementation, we can also see in the figures that the graph of S^3 is almost linear with respect to the bit-size of the input polynomials.

The tests show that our implementation is faster than RS. This could be expected since the latter is a general-dimension solver and is used through is MAPLE interface. On the other hand, our code is slower than CF, which confirms the efficiency of the latter approach [37].

We also tested the MAPLE's `solve` function (together with the `evalf` function), which uses the direct formulas for the roots, based on radical and complex numbers. In the cubic case, MAPLE was not able to complete the experiments, after 30 hours. This illustrates the fact that the closed form expressions are not useful from the computational point of view.

Bivariate systems. We performed two kinds of experiments on real solving of bivariate polynomial systems of degree ≤ 2, and the results are in table 3. For every test we picked two polynomials at random and solved them; we repeat this 10000 times. Column A refers to 1000 bivariate polynomials, with integer coefficients sampled in $[-10, 10]$, with few intersections: every polynomial has common real roots with 135 others in the list, on average. Column B refers to 1000 conics sampled by 5 random integer points in $[-10, 10] \times [-10, 10]$, where two random conics probably intersect: every conic has common real roots with 970 others in the list, on the average.

We tested our algorithms versus NEWMAC [23], which is a general purpose polynomial system solver. STH, in SYNAPS, is based on Sturm-Habicht sequences and subresultants, following [14]. `res` is a bivariate polynomial solver based on the Bezoutian matrix and LAPACK [2]. For FGB/RS [29] we use its MAPLE interface, since the source code is not available, which explains the slow times of this package. S^3 refers to our code, while f-S^3 is our code based on `Lazy_exact_nt`.

We emphasize that our approach is exact, i.e. it outputs isolating boxes with rational endpoints containing a unique root whose multiplicity is also calculated. This is not the case for STH and `res`. STH, uses a double approximation in order to compute the ordinate of the solution. `res` works only width doubles, since it has to compute generalized eigenvalues and eigenvectors and that is why it cannot perform the tests of Column B. NEWMAC, also relies on the computation of eigenvalues and computes also the complex solutions of the system. S^3 is faster on both data sets. When we use filtering, our code is 3 times faster than any other approach, but somewhat slower in Column A.

For additional experiments we refer the reader to [10]. In a nutshell, our code is one of the fastest and the most robust concerning real solving of bivariate systems of polynomials of degree 2.

8 Beyond the quartic

One important consequence of Prop. 6 is is that we can compute isolating points between the real roots of polynomials of degree up to 5 using square roots. We show how this fact leads to *rational* isolating points.

Consider the general quintic

$$f(x) = x^5 + bx^3 + cx^2 + dx + e.$$

Using as $B_1(x) = (-4\,d^2 + 10\,ce)\,x^2 + 5\,dex + 25\,e^2$ and $C_1(x) = (20\,d^2 - 50\,ce)x - 25\,de$, from Prop. 6 we get $A_1(x) = (125\,e^2 + 8\,bd^2 - 20\,bce)x^4 + (-10\,deb + 12\,cd^2 - 30\,c^2e)x^3 + (75\,e^2b - 55\,dec + 16\,d^3)\,x^2$. Thus, the isolating points of the real roots of the quintic are:

$$0, \quad \frac{5de \pm e\sqrt{17\,d^2 - 40\,ce}}{8\,d^2 - 10\,ce}, \quad \frac{-5\,deb + 6\,cd^2 - 15\,c^2e \pm \sqrt{\delta_1}}{125\,e^2 + 8\,bd^2 - 20\,bce},$$

where $\delta_1 = -575\,d^2e^2b^2 + 700\,d^3ebc - 950\,de^2bc^2 + 36\,c^2d^4 - 180\,c^3d^2e + 225\,c^4e^2 - 9375\,e^4b + 6875\,e^3dc - 2000\,e^2d^3 - 128\,bd^5 + 1500\,b^2ce^3$.

Using as $B_2(x) = -10\,bx^2 + 15\,cx - 4\,b^2$ and $C_2(x) = 50\,bx - 75\,c$, we get

$A_2(x) = (-12\,b^3 - 45\,c^2 + 40\,bd)\,x^2 + (-8\,b^2c - 60\,dc + 50\,be)\,x - 4\,b^2d - 75\,ec$. Thus, we get another set of isolating points, which are

$$\frac{15\,c \pm \sqrt{225\,c^2 - 160\,b^3}}{20b}, \quad -\frac{30\,dc - 25\,be + 4\,b^2c \pm \sqrt{\delta_2}}{12\,b^3 + 45\,c^2 - 40\,bd},$$

where $\delta_2 = 900\,d^2c^2 + 1500\,dcbe + 60\,dc^2b^2 + 625\,b^2e^2 - 1100\,b^3ec + 16\,b^4c^2 - 48\,b^5d - 3375\,c^3e + 160\,b^3d^2$.

By applying Lem. 7, one computes *rational* isolating points between all real roots of the quintic, in all cases.

Lastly, we can compute isolating points for the real roots of polynomials of degree up to 9, using quartic roots.

Acknowledgment. The authors are grateful to Victor Pan for various suggestions. The second author thanks M-F. Roy and M. Safey El Din for discussions, and B. Mourrain for his help with the implementation and the experiments. Both authors acknowledge partial support by IST Programme of the EU as a Shared-cost RTD (FET Open) Project under Contract No IST-006413-2 (ACS - Algorithms for Complex Shapes). The second author is also partially supported by contract ANR-06-BLAN-0074 "Decotes".

References

[1] S. Basu, R. Pollack, and M-F.Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer-Verlag, 2nd edition, 2006.

[2] L. Busé, H. Khalil, and B. Mourrain. Resultant-based methods for plane curves intersection problems. In V.G. Ganzha, E.W. Mayr, and E.V. Vorozhtsov, editors, *Proc. 8th Int. Workshop Computer Algebra in Scientific Computing*, volume 3718 of *LNCS*, pages 75–92. Springer, 2005.

[3] J. E. Cremona. Reduction of binary cubic and quartic forms. *LMS J. Computation and Mathematics*, 2:62–92, 1999.

[4] E. W. Dijkstra. *A Discipline of Programming*. Prentice Hall, 1998.

[5] D. I. Diochnos, I. Z. Emiris, and E. P. Tsigaridas. On the complexity of real solving bivariate systems. In C. W. Brown, editor, *Proc. ACM Intern. Symp. on Symbolic & Algebraic Comput. (ISSAC)*, pages 127–134, Waterloo, Canada, 2007.

[6] L. Dupont, D. Lazard, S. Lazard, and S. Petitjean. Near-optimal parameterization of the intersection of quadrics. In *Proc. Annual ACM Symp. on Comp. Geometry (SoCG)*, pages 246–255. ACM, June 2003.

[7] A. Eigenwillig, L. Kettner, E. Schömer, and N. Wolpert. Complete, exact, and efficient computations with cubic curves. In *Proc. Annual ACM Symp. on Computational Geometry (SoCG)*, pages 409–418, 2004.

[8] I. Z. Emiris and M.I. Karavelas. The predicates of the Apollonius diagram: algorithmic analysis and implementation. *Comp. Geom.: Theory & Appl., Spec. Issue on Robust Geometric Algorithms and their Implementations*, 33(1-2):18–57, 2006.

[9] I. Z. Emiris and E. P. Tsigaridas. Computing with real algebraic numbers of small degree. In *Proc. ESA*, volume 5045 of *LNCS*, pages 652–663. Springer, 2004.

[10] I. Z. Emiris, A.V. Kakargias, M. Teillaud, S. Pion, and E. P. Tsigaridas. Towards an open curved kernel. In *Proc. Annual ACM Symp. on Comp. Geometry (SoCG)*, pages 438–446, New York, 2004. ACM Press.

[11] I. Z. Emiris, E. P. Tsigaridas, and G. Tzoumas. Predicates for the exact Voronoi diagram of ellipses under the Euclidean metric. *Int. J. of Computational Geometry and its Applications*, 2007. Special issue devoted to SoCG 2006.

[12] I. Z. Emiris, B. Mourrain, and E. P. Tsigaridas. A rational in between. In *Proc. ACM Intern. Symp. on Symbolic & Algebraic Comput. (ISSAC)*, 2008. (Poster presenation). To appear.

[13] I. Z. Emiris, B. Mourrain, and E. P. Tsigaridas. Real Algebraic Numbers: Complexity Analysis and Experimentation. In P. Hertling, C. Hoffmann, W. Luther, and N. Revol, editors, *Reliable Implementations of Real Number Algorithms: Theory and Practice*, volume 5045 of *LNCS*, pages 57–82. Springer Verlag, 2008. Also available in www.inria.fr/rrrt/rr-5897.html.

[14] L. Gonzalez-Vega and I. Necula. Efficient topology determination of implicitly defined algebraic plane curves. *Computer Aided Geometric Design*, 19(9):719–743, December 2002.

[15] L. González-Vega, H. Lombardi, T. Recio, and M-F. Roy. Sturm-Habicht Sequence. In *Proc. ACM Intern. Symp. on Symbolic & Algebraic Comput. (ISSAC)*, pages 136–146, 1989.

[16] H. Hong. Bounds for absolute positiveness of multivariate polynomials. *J. Symbolic Computation*, 25(5):571–585, May 1998.

[17] D. Kaplan and J. White. Polynomial equations and circulant matrices. *The Mathematical Association of America (Monthly)*, 108:821–840, 2001.

[18] J. Keyser, T. Culver, D. Manocha, and S. Krishnan. ESOLID: A system for exact boundary evaluation. *Comp. Aided Design*, 36(2):175–193, 2004.

[19] J. Kioustelidis. Bounds for the positive roots of polynomials. *J. of Computational and Applied Mathematics*, 16:241–244, 1986.

[20] D. E. Knuth. *The art of computer programming, vol. 2: seminumerical algorithms*. Addison-Wesley, 3rd edition, 1997.

[21] D. Lazard. Quantifier elimination: optimal solution for two classical examples. *J. Symbolic Computation*, 5(1-2):261–266, 1988.

[22] T. Lickteig and M-F. Roy. Sylvester-Habicht Sequences and Fast Cauchy Index Computation. *J. Symbolic Computation*, 31(3):315–341, 2001.

[23] B. Mourrain and Ph. Trébuchet. Algebraic methods for numerical solving. In *Proc. of the 3rd Int. Workshop on Symbolic and Numeric Algorithms for Scientific Computing'01 (Timisoara, Romania)*, pages 42–57, 2002.

[24] B. Mourrain, M. Vrahatis, and J.C. Yakoubsohn. On the complexity of isolating real roots and computing with certainty the topological degree. *J. Complexity*, 18(2), 2002.

[25] B. Mourrain, P. Pavone, P. Trébuchet, E. P. Tsigaridas, and J. Wintz. SYNAPS, a library for dedicated applications in symbolic numeric computations. In M. Stillman, N. Takayama, and J. Verschelde, editors, *IMA Volumes in Mathematics and its Applications*, pages 81–110. Springer, New York, 2007.

[26] V. Pan. Solving polynomial equation: Some history and recent progress. *SIAM Review*, 39:187–220, 1997.

[27] V.Y. Pan, B. Murphy, R.E. Rosholt, G. Qian, and Y. Tang. Real root-finding. In *Proc. Workshop on Symbolic-Numeric Computation (SNC)*, pages 161–169. ACM Press, NY, USA, 2007.

[28] R. Rioboo. Towards faster real algebraic numbers. In Teo Mora, editor, *Proc. ACM Intern. Symp. on Symbolic & Algebraic Comput. (ISSAC)*, pages 221–228, New York, USA, 2002. ACM Press. ISBN 1-58113-484-3.

[29] F. Rouillier. Solving zero-dimensional systems through the rational univariate representation. *J. of Applicable Algebra in Engineering, Communication and Computing*, 9(5):433–461, 1999.

[30] F. Rouillier and Z. Zimmermann. Efficient isolation of polynomial's real roots. *J. of Computational and Applied Mathematics*, 162(1):33–50, 2004.

[31] D. Russel, M.I. Karavelas, and L.J. Guibas. A package for exact kinetic data structures and sweepline algorithms. *Comput. Geom.: Theory & Appl.*, 38:111–127, 2007. Special Issue.

[32] M. Safey El Din. Testing sign conditions on a multivariate polynomial and applications. *Mathematics in Computer Science*, 1(1):177–207, Dec 2007.

[33] G. Salmon. *Lessons Introductory to the Modern Higher Algebra*. Chelsea Publishing Company, New York, 1885.

[34] S. Schmitt. The diamond operator for real algebraic numbers. Technical Report ECG-TR-243107-01, MPI Saarbrücken, 2003.

[35] T. W. Sederberg and G.-Z. Chang. Isolating the real roots of polynomials using isolator polynomials. In C. Bajaj, editor, *Algebraic Geometry and Applications*. Springer, 1993.

[36] D. Ştefănescu. Inequalities on polynomial roots. *Mathematical Inequalities and Applications*, 5(3):335–347, 2002.

[37] E. P. Tsigaridas and I. Z. Emiris. On the complexity of real root isolation using Continued Fractions. *Theoretical Computer Science*, 392:158–173, 2008.

[38] C. Tu, W. Wang, B. Mourrain, and J. Wang. Signature sequence of intersection curve of two quadrics for exact morphological classification. 2005. URL `citeseer.ist.psu.edu/tu05signature.html`.

[39] V. Weispfenning. Quantifier elimination for real algebra – the cubic case. In *Proc. ACM Intern. Symp. on Symbolic & Algebraic Comput. (ISSAC)*, pages 258–263, Oxford, 1994. ACM Press.

[40] L. Yang. Recent advances on determining the number of real roots of parametric polynomials. *J. Symbolic Computation*, 28:225–242, 1999.
[41] C.K. Yap. *Fundamental Problems of Algorithmic Algebra*. Oxford University Press, New York, 2000.

A Cubic real algebraic number

Lemma 17 *There is an algorithm that compares any two roots of two cubic polynomials testing expressions of algebraic degree at most 6. The algorithm is optimal w.r.t. algebraic degree.*

Proof. Consider two cubic polynomials $f_i(x) = a_i x^3 - 3b_i x^2 + 3c_i x - d_= 0$, where $a_i > 0$ and $i \in \langle 1, 2\rangle$, having roots $x_1 < x_2 < x_3$ and $y_1 < y_2 < y_3$, in isolating interval representation. We assume that we want to compare $x_a \cong (f_1, I_1)$ and $y_a \cong (f_2, I_2)$ and that both lie in $J = I_1 \cap I_2 = [\alpha, \beta]$.

We consider the Sturm-Habicht sequence S with $S_0 = f_1$ and $S_1 = f_2$. Since $f_1^{'}(x_2) < 0$ if $\mathtt{VAR}_S[\alpha, \beta] = 1$ then $x_1 > y_1$, if $\mathtt{VAR}_S[\alpha, \beta] = -1$ then $x_1 < y_1$. Finally if $\mathtt{VAR}_S[\alpha, \beta] = 0$ then $x_1 = y_1$.

We consider the Sturm-Habicht sequence S with $S_0 = f_1(x)$ and $S_1 = f_2(x)$, that is the sequence $(S_0, S_1, S_2, S_3, S_4)$, where $S_0(x) = f_1(x)$, $S_1(x) = f_2(x)$, $S_2(x) = -3Jx^2 + 3Gx - M$, $S_3(x) = [J(P + 12J_1) - 3G^2]\, x - (GM - 3JM_1) = Z_1 x + Z_2$, $S_4(x) = J^2(P^3 - 27Q) = J^2\,[M(M^2 - 9JM_2) - 9M_1Z_2 - 9M_2Z_1]$. In order to simplify the computations we apply various geometric invariants, namely: $\Delta_{11} = W_1^2 - 4\Delta_{12}\Delta_{13}$, $\Delta_{21} = W_2^2 - 4\Delta_{22}\Delta_{23}$, $J = [ab] = a_1b_2 - a_2b_1$, $J_1 = [bc] = b_1c_2 - b_2c_1$, $M = [ad] = a_1d_2 - a_2d_1$, $P = M - 3J$, $Z_1 = J(M + 9J_1) - 3G^2$ and $Z_2 = GM - 3JM_1$.

We also use the following expressions, that are not invariants, but look like one: $G = [ac] = a_1c_2 - a_2c_1$, $M_1 = [bd] = b_1d_2 - b_2d_1$, $M_2 = [cb] = c_1d_2 - c_2d_1$, $W_1 = a_1d_1 - c_1b_1$ and $W_2 = a_2d_2 - c_2b_2$. If R is the resultant of the two polynomials, then $R = P^3 - 27Q$ and Q is an invariant. For a detailed treatment of the invariants of a system of two polynomials the reader can refer to [33].

We need to evaluate the Sturm sequence at, at most two, points of the set

$$\left\{\frac{b_1}{a_1}, \frac{b_2}{a_2}, +\infty, -\infty, -\frac{W_1}{2\Delta_{12}}, -\frac{W_2}{2\Delta_{22}}\right\}.$$

Every number in the above set is of algebraic degree at most 2 and the coefficients of $S_0, \ldots S_3$ are of degree at most 4. Moreover, $\deg S_4 = 6$, so we can conclude that the maximum degree involved is 6. By the theory of resultants,

the algorithm is optimal with respect to algebraic degree. □

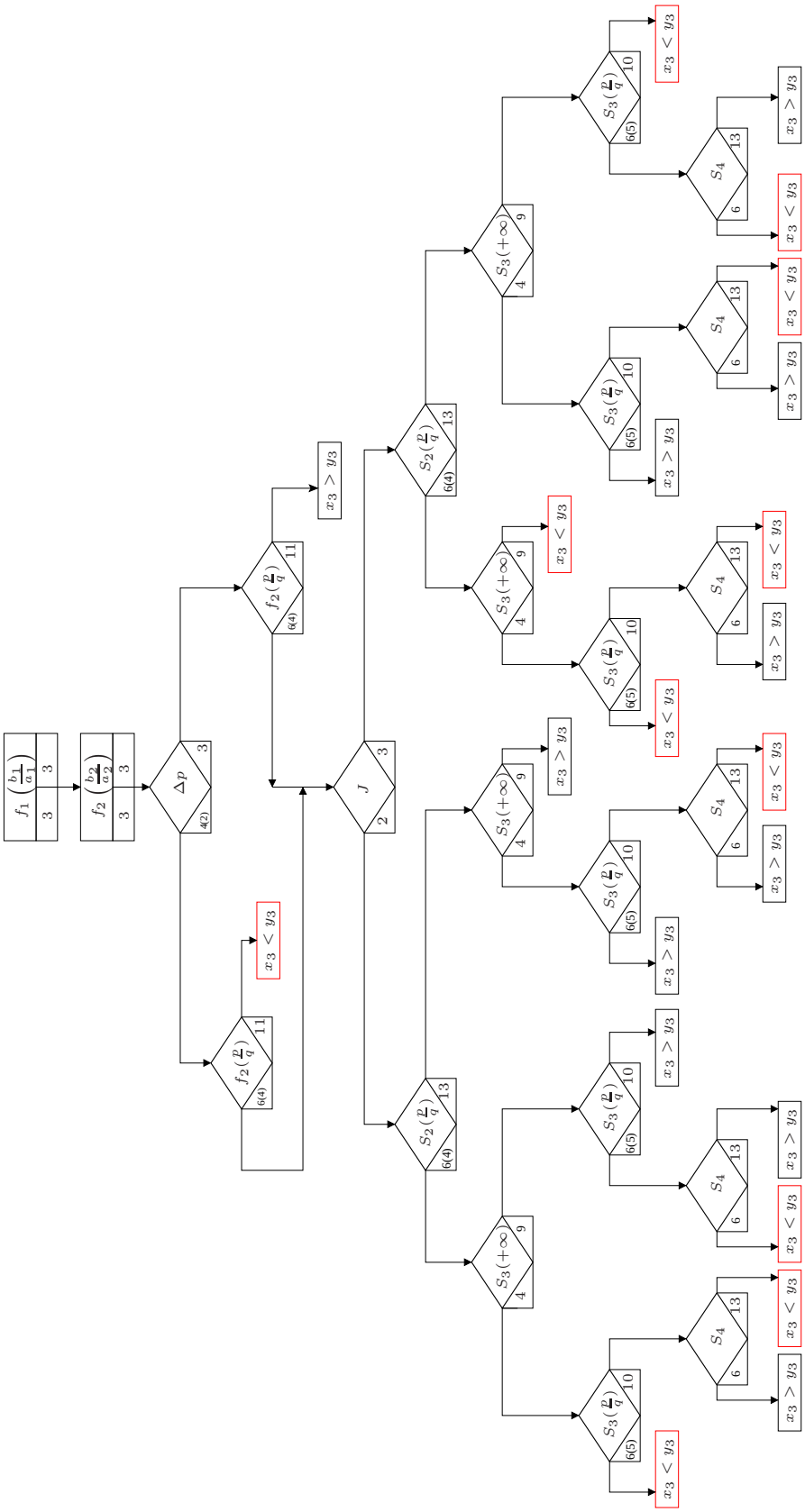


Fig. A.1. Evaluation procedure for comparing the largest roots x_3, y_3 of cubic polynomials f_1, f_2, where $x_3, y_3 \in (\frac{p}{q}, +\infty)$. The S_i are polynomials in the Sturm-Habicht sequence of f_1, f_2, defined in the proof of Lem. 17.