# ECG

*IST-2000-26473*

Effective Computational Geometry for Curves and Surfaces

ECG Technical Report No. : *ECG-TR-302206-03*

# *Comparison of fourth-degree algebraic numbers and applications to geometric predicates (revised version)*

Ioannis Z. Emiris    Elias P. Tsigaridas

Deliverable: 30 22 06 (item 03)
Site: INRIA
Month: 30

Project funded by the European Community
under the "Information Society Technologies"
Programme (1998–2002)

**Ioannis Z. Emiris**
Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, GREECE, and INRIA Sophia-Antipolis, FRANCE (emiris@di.uoa.gr)

**Elias P. Tsigaridas**
Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, GREECE (et@di.uoa.gr)

**Abstract**

We present algorithms for the exact comparison of the real roots of two polynomials of degree 4. The algorithm precomputes Sturm sequences and isolating intervals for the representation of the roots and, additionally, uses various invariants in order to minimize the computational effort. In most cases, the algorithm is optimal with respect to the algebraic degree of the tested quantities in the input coefficients. Our treatment is complete, in the sense that we handle all special cases, including when one of the polynomials has degree smaller than 4. Our algorithms have been implemented, and some preliminary experimental results are presented in order to show their efficiency when compared to the CORE library.

We apply these methods to answer certain geometric predicates that arise in computing the planar arrangement of conic arcs.

In this revised version we added a formal proof about the isolating points of the quartic, more geometric predicates that can be used in arrangement and two sections about the solution of a bivariate system of two equations of degree at most 2 and about the sign of a bivariate polynomial of degree 2 over two algebraic numbers.

# 1 Introduction

This work continues upon [12], which settled the case of algebraic numbers of degree up to 3. New tools are needed here, especially algorithms for computing separator points between the polynomial's real roots. The problem of finding such points of low algebraic degree (ideally rational) is a deep question of independent interest; we only scratch the surface of this problem, which leads us towards computational number theory.

# 2 The quartic polynomial

The polynomial equation of degree 4 is a well studied equation. It is one of the few polynomial equations that can be solved explicitly with radicals, but one needs to operate with $\sqrt{-1}$ even for computing the real roots. Several approaches exist in order to solve the quartic. Refer to [**?**] for the general solution of the quartic and to [15] and [**?**] for a unified approach, using circulant matrices.

Consider the quartic polynomial equation, where $a > 0$ WLOG.

$$f(X) = aX^4 - 4bX^3 + 6cX^2 - 4dX + e = 0 \tag{1}$$

In the entire report, we shall consider as input the coefficients $a, b, c, d, e$. Our algorithms typically test the sign of certain polynomial quantities in these coefficients. From a complexity viewpoint, we wish to minimize the degree of the tested quantities in the input data, namely the coefficients $a, b, c, d, e$.

Let us discuss the invariants of $f$, which shall be instrumental in the computations below. For background on invariant theory, see [28, 25]. For a more a more comprehensive view of the invariants of cubic and quartic the reader can refer to [5, 4]. For applications in comparing real algebraic numbers of degree up to 3, see [12]. We consider the rational invariants of $f$, this means the invariants for all transformation matrices in $GL(2, \mathbb{Q})$. The invariants form a graded ring [4], generated by two invariants of degree 2 and 3, which are conventionally denoted by $A$ and $B$:

$$\begin{aligned} A &= W_3 + 3\Delta_3 \\ &= ae - 4bd + 3c^2 \qquad (2) \\ B &= -dW_1 - e\Delta_2 - c\Delta_3 \\ &= ace + 2bcd - ad^2 - eb^2 - c^3 \qquad (3) \end{aligned}$$

These invariants are algebraically independent. Every other invariant is an isobaric polynomial in $A$ and $B$, this means that every other invariant is homogeneous in the coefficients of the quartic and occurs from combinations of powers of $A$ and $B$. We will denote the invariant $A^3 - 27B^2$ by $\Delta_1$ and we refer to it as the *discriminant.*

The semivariants of the quartic (which are the leading coefficients of the covariants [5, 4]) are the invariants $A$ and $B$ together with:

$$\begin{aligned} \Delta_2 &= b^2 - ac \\ R &= aW_1 + 2b\Delta_2 \\ &= -2b^3 - a^2d + 3abc \\ Q &= 12\Delta_2^2 - a^2A \\ &= 9a^2c^2 - 24acb^2 + 12b^4 - ea^3 + 4a^2db \end{aligned} \qquad (4)$$

We shall also need the quantities, which are not necessarily invariants

$$\begin{aligned} \Delta_3 &= c^2 - bd \\ \Delta_4 &= d^2 - ce \\ W_1 &= ad - bc \\ W_2 &= be - cd \\ W_3 &= ae - bd \\ T &= -9W_1^2 + 27\Delta_2\Delta_3 - 3W_3\Delta_2 \end{aligned} \qquad (5)$$

**Proposition 1** *Let $f(X)$ be a quartic as in Equation (1). The following table gives the numbers of real roots and their multiplicities in all cases ([30]).*

| | | |
|---|---|---|
| (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) | $\Delta_1 = 0 \wedge T = 0 \wedge \Delta_2 = 0$ | $\{4\}$ |

*The right column of the above table describes the situation of the roots. For example, $\{1,1,1,1\}$ means four simple real roots and $\{2,2\}$ means two double real roots. It is worth to say that in cases (2) and (8) there are no real roots, but in case (2) there are no repeated roots, while in case (8) there are two imaginary double roots.*

We have to mention that our notation differs from the one at [30] because we use the quartic with normalized coefficients and that in [30] there is a little error in the definition of $T$. Additionally, we use Sturm sequences in order to derive Proposition 1 while [30] used a discrimination system.

# 3 Isolating polynomials

**Theorem 1 (Isolating polynomials)** *Given a polynomial $P(X)$ with two adjacent roots $\gamma_1$ and $\gamma_2$, and given two other polynomials $B(X)$ and $C(X)$, let us define:*

$$A(X) := B(X)P'(X) + C(X)P(X),$$

*where $P'(x)$ is the first derivative of $P(X)$. Then $A(X)$ or $B(X)$ has at least one real root in the closed interval $[\gamma_1, \gamma_2]$.*

For a proof of the above theorem see [27]. We can use the above Theorem in order to isolate the roots of the quartic in all cases that appear in Proposition 1. For what follows separating points and isolating points mean the same thing.

Considering the polynomial remainder sequence of $P$ and $P'$, we can obtain, as a corollary, that $\deg A + \deg B \leq \deg P - 1$.

## 3.1 First isolating polynomial

In order to find points on the $x-$axis that isolate the roots of a quartic we use the Theorem of isolating polynomials.

Let $B(X) = ax - b$ and $C(X) = -4a$ then

$$A(X) = 3\,\Delta_2\,X^2 + 3\,W_1\,X - W_3. \tag{6}$$

Since $\frac{b}{a}$, which is the solution of $B(X) = 0$, is the arithmetic mean of the four roots, it is certainly somewhere between the roots of the quartic. The other two isolating points are the solutions of Equation (6), which are

$$\tau_{1,2} = \frac{-3W_1 \pm \sqrt{9W_1^2 + 12\Delta_2 W_3}}{6\Delta_2} \tag{7}$$

We can easily verify that $\text{sign}\left(f(\frac{b}{b})\right) = \text{sign}\left(a^2A - 3\Delta_2^2\right)$ and so the order of the isolating points is

$$\begin{cases} \tau_1 < \frac{b}{a} < \tau_2, & \text{if } f(\frac{b}{a}) > 0; \\ \tau_1 < \tau_2 < \frac{b}{a}, & \text{if } f(\frac{b}{a}) < 0 \;\&\; R > 0; \\ \frac{b}{a} < \tau_1 < \tau_2, & \text{if } f(\frac{b}{a}) < 0 \;\&\; R < 0. \end{cases} \tag{8}$$

If $f(\frac{b}{a}) = 0$ then we know exactly one root of the quartic and if we do division we can express the other three roots as roots of a cubic. Notice that if $R\,f(\frac{b}{a}) = 0$ then the quartic has a double root.

We must mention that the discriminant of this isolating polynomial is a an invariant of the original quartic with respect to translation.

### 3.2 Second isolating polynomial

We apply again the theorem about the isolating polynomials but now $B(X) = dx - e$ and $C(X) = -4d$. So we have

$$A(X) = W_3\, X^3 - 3\, W_2\, X^2 - 3\,\Delta_4\, X \tag{9}$$

The theorem about isolating polynomials tells us that at least two of the numbers below are between the roots of the quartic. Notice that since there is no constant term, the polynomial has zero as root.

$$0, \sigma_{1,2} = \frac{3W_2 \pm \sqrt{9W_2^2 + 12\Delta_4 W_3}}{6W_3} \tag{10}$$

WLOG we assume that the roots are all positive and so 0 is not an isolating point. The order of the isolating points can be determined by a similar way. The results of this paragraph can be obtained by the results of the previous subsection by considering the reverse polynomial $X^4 f(\frac{1}{X})$.

### 3.3 Finding more isolating polynomials

We consider the quartic $f(X) = \sum_{i=1}^{4} = a_i X^i$. Let $B(X) = a_i X - K a_j$ and $C(X) = L$ where $a_i$ and $a_j$ are coefficients of the quartic (but they could be any rationals) and $K$ and $L$ are parameters. We consider the equation

$$A(X) = B(X) f^{'}(X) + C(X) f(X). \tag{11}$$

The polynomial $A(X)$ is now a quartic of the form

$$A(X) = b_4 X^4 + b_3 X^3 + b_2 X^2 + b_1 X + b_0 \tag{12}$$

where the coefficients $b_i$ are functions in $K$ and $L$. We can choose to eliminate two of these coefficients by equating them to zero. We solve the corresponding system for the parameters and we have the isolating polynomials $A(X)$ and $B(X)$.

If we choose to eliminate $b_4$ and $b_3$ then we have the first isolating polynomial. If we choose to eliminate $b_3$ and $b_1$ then we have an isolating quartic which is

$$3\, a W_1 x^4 + 6(3b\Delta_3 - d\Delta_2)x^2 - eW_1 - 8b\Delta_3 \tag{13}$$

The smallest and the largest root of the above quartic always isolate the smallest and the largest root of the original quartic.

At this point we have to mention that by the above method we can always find a biquadratic that isolates the roots of every quartic.

## 4 Isolate the roots of every quartic

Let us now find the isolating points for all the cases of Proposition 1.

$\{1, 1, 1, 1\}$ We apply the theorem of isolating polynomials.

$\{\}$ Nothing to do since the quartic has no real roots.

$\{1,1\}$ We apply the theorem of the isolating polynomials in order to derive the one and only isolating point by testing the sign of $f$ over the isolating points. The isolating point may *not* be rational.

$\{2,1,1\}$ We can compute the double root from the pseudo-remainder sequence $\overline{P}_{f,f'}$. The double root is rational since it is the only root of $\mathrm{GCD}(f,f')$ and its value is $\frac{T_1}{T_2}$. In theory, we could divide it out and use the isolating points of the cubic.

When the double root is the middle root then $\frac{b}{a}$ and $-\frac{W_1}{2\Delta_2}$ are isolating points for the other two roots. When the double root is the smallest or the biggest root we apply the theorem of isolating polynomials in order to find one more isolating point in $\mathbb{Q}$.

$\{2\}$ We can compute the double root from the pseudo-remainder sequence $\overline{P}_{f,f'}$. This root is rational since it is the only root of $\mathrm{GCD}(f,f')$.

$\{2,2\}$ The two roots of the quartic are also roots of the derivative. To be more specific they are the smallest and the biggest root of the derivative. So in order to encode and isolate them we use the derivative which is a cubic. Additionally we can express the two roots as the roots of the polynomial $abX^2 - 2b^2X + ad$. We prefer the cubic since in this case the algebraic degree of the coefficients is one. In any case, the isolating points $\in \mathbb{Q}$ because they are obtained from those of a cubic.

$\{3,1\}$ In this case we can compute the roots exactly, ie. as rationals. The triple root is $-\frac{W_1}{2\Delta_2}$ and the single root is $\frac{3\,aW_1+8\,b\Delta_2}{2\,a\Delta_2}$.

$\{\}$ Nothing to do since the quartic has now real roots.

$\{4\}$ The one real root is $\frac{b}{a} \in \mathbb{Q}$.

We are considering the case where the quartic has 4 or 2 simple real roots exactly, since otherwise it is clear from the previous paragraph that we can easily find rational points that isolate the roots. The case $\{1,1\}$ is easier than $\{1,1,1,1\}$, so we focus on the latter. Additionally, we assume that 0 is not a root of the quartic.

### 4.1 Rational points that isolate the roots of the quartic

Consider the quartic polynomial equation, where $a > 0, b = 0$.

$$f(X) = aX^4 + 6cX^2 - 4dX + e = 0.$$

If we specialize Equation (7) with $b = 0$ we get the equations

$$\tau_{1,2} = \frac{3\,d \pm \sqrt{9\Delta_4 - 3\,ce}}{6c} \tag{14}$$

If we specialize Equation (10) with $b = 0$ we get the equations

$$\sigma_{1,2} = \frac{-3\,dc \pm \sqrt{9\,d^2c^2 + 12\,ae\Delta_4}}{2ae} \tag{15}$$

We use the following lemma

**Lemma 1** *For any rationals $0 < \frac{m}{n} < \frac{m'}{n'}$ the following inequality holds*

$$\frac{m}{n} < \frac{m+m'}{n+n'} < \frac{m'}{n'}$$

**Proof.** The proof is easy by considering the inequality $mn' < m'n$. □

The most difficult case is when $\tau_i$ and $\sigma_j$, $i, j \in \{1, 2\}$, isolate the same pair of adjacent roots. Without loss of generality assume that these are $\tau_1$ and $\sigma_1$. In order to simplify the notation let

$$\begin{array}{rcl} A & = & 9\Delta_4 - 3\,ce \\ B & = & 12\,ae\Delta_4 + 9\,d^2c^2 \end{array} \tag{16}$$

So the isolating point for these two adjacent roots is $\frac{3d-3dc+\sqrt{A}+\sqrt{B}}{6c+2ae}$. If we can find a rational number $\frac{P}{Q}$ between $\sqrt{A}$ and $\sqrt{B}$, then we are done since we can replace their sum with $2\frac{P}{Q}$.

We assume that the quartic has 4 real roots hence by Newton's Theorem (see [31]), the following inequalities hold

$$b^2 - ac \geq 0 \Rightarrow ac \leq 0, c^2 - ba \geq 0, d^2 - ce \geq 0,$$

where the 2nd inequality gives no special information, but the first one yields $a > 0 \Rightarrow c \leq 0$. Since $b = 0$, then by Descartes' rule of signs we can conclude that there are the following cases:
$e > 0$, when there are 2 positive and 2 negative roots (2 sign changes).
$e < 0$ when there are 3 positive and one negative roots (3 changes), or vice versa (1 sign change).
$e = 0$ then there is exactly one zero root and $d > 0$ (or $d < 0$) depending on whether there are 2 (or 1) positive and 1 (or 2) negative roots.

**Theorem 2** *For every quartic with 2 distinct or 4 distinct real roots (and $b = 0$)*

$$\sqrt{9\Delta_4 - 3\,ce} \leq \lfloor \sqrt{9\Delta_4 - 3\,c\,e} \rfloor + 1 \leq \sqrt{9\,d^2c^2 + 12\,ae\Delta_4} \tag{17}$$

*or alternatively* $|\sqrt{9\Delta_4 - 3\,ce} - \sqrt{9\,d^2c^2 + 12\,ae\Delta_4}| \geq 1$.

**Proof.** Let $A = 9d^2 - 12ce$ and $B = 12aed^2 - 12ace^2 + 9d^2c^2$. It is enough to show that:

$$\begin{array}{rcll} \sqrt{B} & \geq & 1 + \sqrt{A} & \Leftrightarrow \\ \sqrt{\frac{B}{A}} & \geq & 1 + \frac{1}{\sqrt{A}} & \Leftarrow \\ \sqrt{\frac{B}{A}} & \geq & 2 & \Leftrightarrow \\ \frac{B}{A} & \geq & 4 & \Leftrightarrow \\ \frac{B}{A} = \frac{4aed^2 - 4ace^2 + 3d^2c^2}{3d^2 - 4ce} & \geq & 4 & \Leftrightarrow \\ 4aed^2 - 4ace^2 + 3d^2c^2 & \geq & 12d^2 - 16ce & \Leftrightarrow \\ 4aed^2 - 4ace^2 + 3d^2c^2 - 12d^2 + 16ce & \geq & 0. & \end{array} \tag{18}$$

By letting $g(a, c, d, e) = 4aed^2 - 4ace^2 + 3d^2c^2 - 12d^2 + 16ce$, our problem is to find the minimum of $g$, subject to the constraints $-a \leq 1$, $c \leq -5$, $-d \leq 0$ and $-e \leq -5$ (we treat the case where

$c > -5$ and $e < 5$ later). We introduce slack variables $y_1$, $y_2$ and $y_2$ and we use Lagrange multipliers. So our problem now is

$$\begin{aligned} \min\ L(c,e,y_1,y_2,\lambda_1,\lambda_2) \ = \ & g(c,e)+ \\ & \lambda_1(c+y_1^2+5)+ \\ & \lambda_2(-e+y_2^2+5) \\ & \lambda_3(-a+y_3^2+1) \end{aligned} \tag{19}$$

We take partial derivatives

$$\begin{array}{lcll} \frac{\partial}{\partial a}L & = & 12\,e\left(d^2-ce\right)-\lambda_3 & = \ 0 \\ \frac{\partial}{\partial c}L & = & -12\,ae^2+18\,d^2c+48\,e+\lambda_1 & = \ 0 \\ \frac{\partial}{\partial d}L & = & 24\,aed+18\,dc^2-72\,d & = \ 0 \\ \frac{\partial}{\partial e}L & = & 12\,a\left(d^2-ce\right)-12\,aec+48\,c-\lambda_2 & = \ 0 \\ \frac{\partial}{\partial y_1}L & = & 2\,\lambda_1 y_1 & = \ 0 \\ \frac{\partial}{\partial y_2}L & = & 2\,\lambda_2 y_2 & = \ 0 \\ \frac{\partial}{\partial y_3}L & = & 2\,\lambda_3 y_3 & = \ 0 \\ \frac{\partial}{\partial \lambda_1}L & = & c+5+{y_1}^2 & = \ 0 \\ \frac{\partial}{\partial \lambda_2}L & = & -e+5+{y_2}^2 & = \ 0 \\ \frac{\partial}{\partial \lambda_3}L & = & -a+1+{y_3}^2 & = \ 0 \end{array} \tag{20}$$

The solution of the above system is $(a,c,d,e) = (1,-5,0,5)$ and $(g(1,-5,0,5) = 300 > 0$ which is a local minimum.

If $-5 < c < 0$ and $0 < e < 5$ we substitute all the combinations to $A$ and $B$ and the we can see that $\sqrt{A}-\sqrt{B} \geq 1$.

If $c = 0$ then $\sqrt{A} = 3|d|$, and we have a rational isolating point. □

### 4.2 Isolating the roots of a cubic

In ([12]) we used a geometric proposition in order to isolate the roots of the cubic. If we use the theorem of isolating polynomials we find the same results.

## 5 Sturm Sequences

Sturm sequences is a well known and useful tool for isolating the roots of any polynomial. In [16] static Sturm sequences were used in order to compare the roots of polynomials of degree 2. Below we give a small introduction to the Sturm sequences. For a more comprehensive view of the definitions and the theorems below see [31] or [**?**].

**Definition 1 (Sturm Sequence)** *Let $P$ and $Q \in \mathbb{R}[x]$ be non-zero polynomials. By a (generalized) Sturm sequence for $P$ and $Q$ we mean any pseudo-remainder sequence (PRS)*

$$\overline{P} = (P_0, P_1, \ldots, P_n), \quad n \geq 1,$$

*such that for all $i = 0, \ldots, n$, we have*

$$\alpha_i P_{i-1} = Q_i P_i + \beta_i P_{i+1}$$

*$(Q_i \in \mathbb{R}[x], \alpha, \beta \in \mathbb{R})$, such that $\alpha_i\beta_i < 0$ and $P_{n+1} = 0$. This PRS is called Sturm sequence or signed pseudo-remainder sequence. We usually write $\overline{P}_{P_0,P_1}$ if we want to denote the first two terms in the sequence.*

**Definition 2** *For a number $\gamma \in \mathbb{R}$ and a Sturm sequence $\overline{P} = (P_0, P_1, \ldots, P_n)$, $V_{\overline{P}}(\gamma)$ will denote the number of sign variations of the sequence of values of the $P_i$ at $\gamma$, $0 \leq i \leq n$.*

Considering the above definitions the following theorems hold

**Theorem 3 (Schwartz-Sharir)** *Let $P, Q \in \mathbb{R}[x]$ be square-free polynomials. If $\alpha < \beta$ are both non-roots of $P$ then*

$$V_{P,Q}[\alpha, \beta] = V_{P,Q}(\alpha) - V_{P,Q}(\beta) = \sum_{\gamma} \operatorname{sign}(P^{'}(\gamma)Q(\gamma))$$

*where $\gamma$ ranges over the roots of $P$ in $[\alpha, \beta]$.*

**Theorem 4 (Sylvester, revisited by Ben-Or, Kozen, Reif)** *Let $\overline{P}$ be a Sturm sequence for $P, P^{'}Q$ where $P$ is square free and $P, Q$ are relative prime. Then for all $\alpha < \beta$ which are non-roots of $P$,*

$$V_{\overline{P}}[\alpha, \beta] = \sum_{\gamma} \operatorname{sign}(Q(\gamma))$$

*where $\gamma$ ranges over the roots of $P$ in $[\alpha, \beta]$.*

It is possible to use the above theorems to order the roots of any pair of polynomials, of any degree, but as a prerequisite we must either find combinations of Sturm sequences that distinguish among all cases or find intervals that contain only one root of every polynomial. The former is not always possible. As regards to isolating intervals we use them in the followings sections and we shall see that they can leads to optimal or to nearly optimal algorithms for root comparison.

The computation of a Sturm sequence is a quite expensive computational task.

Assume that we want to compute the Sturm sequence of two polynomials $P, Q \in \mathbf{Q}[x]$. In order to accelerate the computation we assume that the polynomials are $P, Q \in \mathbf{Q}(a_0, \ldots, a_n, b_0, \ldots, b_n)[x]$, where $a_i$ and $b_j$ are the coefficients of the two polynomials and now are considered as parameters. Next we pre-compute various Sturm sequences (for various $n$ and $m$) and when we want a specific sequence we specialize the parameters. The problem now is that these sequences do not commute with specialization.

In order to take account of all the possible signed remainder sequences that might appear by specializing the parameters we use the definitions and the notation from [1].

**Proposition 2 (Signed pseudo-remainder)** *Let*

$$P = \sum_{i=0}^{n} a_i X^i, \quad Q = \sum_{j=0}^{m} b_i X^i, \quad P, Q \in D[X] \tag{21}$$

*where $D$ is any subring of $\mathbf{C}$. The signed pseudo-remainder is the negative remainder of the euclidean division of $b_m^d P$ by $Q$, where $d$ is the smallest even integer greater than or equal to $n - m + 1$ (we assume $n \geq m$).*

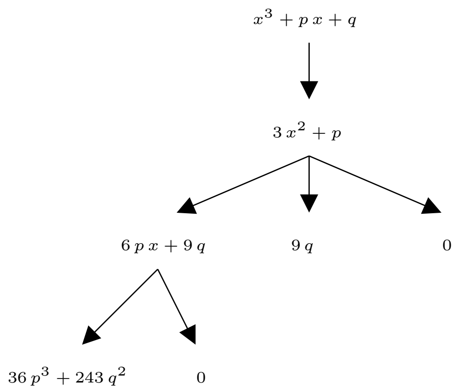


Figure 1: The tree of all possible cases of $\overline{P}_{f,f'}$

**Proposition 3 (Truncation and set of truncations)** *Let*

$$Q = \sum_{j=0}^{m} b_i X^i, \quad Q \in D[X] \tag{22}$$

*We define for $0 \leq i \leq m$ the truncation of $Q$ at $i$ by*

$$\mathrm{TR}_{\,i}(Q) = b_i X^i + \ldots + b_0 \tag{23}$$

*The set of truncations of polynomial $Q \in D[b_m, \ldots, b_0][X]$ is a finite subset of $Q \in D[b_m, \ldots, b_0][X]$ defined by*

$$\mathrm{TR}(Q) = \begin{cases} \{Q\}, & \textit{if } b_m \in D \\ \{Q\} \cup \mathrm{TR}(\mathrm{TR}_{\deg Q - 1}(Q)), & \textit{othewise.} \end{cases} \tag{24}$$

*The tree of all possible signed pseudo-remainder sequences of two polynomials $P, Q \in \mathbf{Q}(a_0, \ldots, a_n, b_0, \ldots, b_n)[X]$, is tree whose root contains $P$. The children of the root contains the elements of the set of truncations of $Q$. Each node $N$ contains a polynomial* $\mathrm{Pol}(N) \in \mathbf{Q}(a_0, \ldots, a_n, b_0, \ldots, b_n)[X]$*. A node $N$ is a leaf if* $\mathrm{Pol}(N) = 0$*. If $N$ is not a leaf then the children of $N$ contain the truncations of* $\mathrm{PRem}\,(Pol(p(N)), Pol(N))$*, where $p(N)$ is the parent of $N$. (You can refer to [1] for details).*

So in order to accelerate the computation of the Sturm sequences we have to pre-compute all the paths from the root of tree to every leaf. Now the specialization commutes with the tree of the possible signed pseudo-remainder sequences. We will come back to this issue when we will talk about the implementation of the comparison of roots of two quartics.

In figure 1 you can see the tree of all possible cases of a PRS with $f_0 = f = x^3 + px + q$ and $f_1 = f^{'}$. The tree enumerates all the possible cases for every specialization of the parameters $p$ and $q$.

# 6 Applications of the Sturm sequence

For a more comprehensive description of the applications of the Sturm sequence the reader can refer to [31] and [12].

## 6.1 The sign of a real algebraic number

Suppose that we want to determine the sign of number $\beta$ in a real number field $\mathbb{Q}(\alpha)$. We assume that $\beta$ is represented by a square-free rational polynomial $B(X) \in \mathbb{Q}[X] : \beta = B(\alpha)$, that is square-free. Assume that $\alpha$ is represented by an isolating interval representation

$$\alpha \cong (A, [a, b])$$

where $A(X) \in \mathbb{Z}[X]$ is a square-free polynomial. By using Corollary 3 we can conclude that

$$\text{sign}(B(\alpha)) = \text{sign}(V_{A,B}[a, b] \cdot A^{'}(\alpha)).$$

If $B(X)$ is not square-free then we can decompose into a product of square-free polynomials. If $B$ has $B_1, B_2 \ldots B_k$ as square-free decomposition then

$$\text{sign}(B(\alpha)) = \prod_{i=1}^{k} \text{sign}(B_i(\alpha)).$$

## 6.2 Comparing two real algebraic numbers

Suppose that we want to compare two algebraic numbers $\gamma_1$ and $\gamma_2$ and that we have an isolating interval representation for them, that is

$$\gamma_1 \cong (P_1(x), I_1), \quad \gamma_2 \cong (P_2(x), I_2)$$

where $I_1 = [a_1, b_1]$ and $I_2 = [a_2, b_2]$, then Algorithm 1 performs the comparison.

In Algorithm 1 let $J$ be the intersection of the two isolating intervals $I_1$ and $I_2$. When $J$ is the empty set, that is the case when the two isolating intervals are distinct, then can easily order the two algebraic number by comparing the first endpoint of their isolating interval. When only $\gamma_1$ or $\gamma_2$ belong to the intersection and the other not then we can treat this case similar to the case were they belong to distinct isolating intervals. In order to decide whether an algebraic number is in an interval we must evaluate its corresponding polynomial to the endpoints of the interval.

As to complete the algorithm it suffices to determine exactly the Sturm Sequence that we use. We use the Sturm sequence in the case where both algebraic numbers are in the intersection of their isolating intervals. Assuming this, we can easily conclude that

$$\gamma_1 \geq \gamma_2 \Leftrightarrow P_2(\gamma_1) \cdot P_2^{'}(\gamma_2) \geq 0$$

We can easily obtain the sign of $P_2^{'}(\gamma_2)$ and from Theorem 3 we can obtain the sign of $P_2(\gamma_1)$. That is

$$P_2(\gamma_1) \cdot P_2^{'}(\gamma_2) \geq 0 \Leftrightarrow (V_{P_1,P_2}[c, d]) \cdot (P_1(d) - P_1(c)) \cdot (P_2(d) - P_1(c)) \geq 0 \tag{25}$$

The last polynomial in Sturm sequence $V_{P_1,P_2}$ is always the resultant of the two polynomials. We have to mention that when $\gamma_1 = \gamma_2 \Leftrightarrow P_2(\gamma_1) = 0$, since both polynomials are square-free.

**Algorithm 1** Compare two algebraic numbers

**Require:** $\gamma_1 \cong (P_1(x), I_1), \quad \gamma_2 \cong (P_2(x), I_2)$

**Ensure:** $\gamma_1 < \gamma_2$ or $\gamma_1 > \gamma_2$

  $J \leftarrow I_1 \cap I_2 = [c, d]$

  **if** $J = \emptyset$ **then**

    RETURN $a_1 < a_2 ? \gamma_1 < \gamma_2 : \gamma_1 > \gamma_2$

  **else if** $(\gamma_1 \in J) \wedge (\gamma_2 \notin J)$ **then**

    {Assume $I_2 - J = [m, n]$}

    RETURN $c < m ? \gamma_1 < \gamma_2 : \gamma_1 > \gamma_2$

  **else if** $(\gamma_1 \notin J) \wedge (\gamma_2 \in J)$ **then**

    {Assume $I_1 - J = [m, n]$}

    RETURN $m < c ? \gamma_1 < \gamma_2 : \gamma_1 > \gamma_2$

  **else**

    {Both numbers lie in $J$}

    Evaluate a Sturm sequence on $J$ to decide

  **end if**

# 7 The Sturm sequence for two quartics

At first we consider the Sturm sequence for a quartic by letting $S_0 = f(X)$ and $S_1 = f'(X)$. From this sequence we can find the multiple roots exactly.

$$\begin{aligned} S_0(X) &= f(X) \\ S_1(X) &= f'(X) \\ S_2(X) &= 3\,\Delta_2 X^2 + 3\,W_1 X - W_3 \\ S_3(X) &= T_1\,X + T_2 \\ S_4(X) &= -\Delta_1 \end{aligned} \tag{26}$$

where

$$\begin{aligned} T_1 &= -W_3\Delta_2 - 3\,W_1^2 + 9\,\Delta_2\Delta_3 \\ T_2 &= AW_1 - 9\,bB \end{aligned} \tag{27}$$

If $\Delta_1 = 0$ then we can compute the multiple root of the quartic either from $S_3(X)$ or from $S_2(X)$.For the rest of the section we assume that the quartics that appear have 4 distinct real roots.

Let the two quartics be

$$f_1(X) = a_1 X^4 - 4b_1 X^3 + 6c_1 X^2 - 4d_1 X + e_1 \tag{28}$$

$$f_2(X) = a_2 X^4 - 4b_2 X^3 + 6c_2 X^2 - 4d_2 X + e_2 \tag{29}$$

We consider the Sturm sequence $S$ with $S_0 = f_1(x)$ and $S_1 = f_2(X)$. The complete Sturm sequence is

$$
\begin{array}{rcl}
S_0(X) & = & f_1(X) \\
S_1(X) & = & f_2(X) \\
S_2(X) & = & -4J\,X^3 + 6G\,X^2 - 4M\,X + M_3 \\
S_3(X) & = & S_{32}\,X^2 + S_{31}\,X + S_{30} \\
S_4(X) & = & S_{41}\,X + S_{40} \\
S_5(X) & = & -8M_5(S_{41} - M_3S_{31}) \\
 & & +32M_4(M_5S_{32} - M_4S_{30}) \\
 & & -12M_6(S_{40} - 2M_3S_{30}) \\
 & & +M_3^2(M_3^2 - 16MM_4 - 16JM_5 + 36GM_6)
\end{array}
\tag{30}
$$

where

$$
\begin{array}{rcl}
S_{32} & = & 2\left[4J(M+6J_1) - 9G^2\right] \\
S_{31} & = & 2\left[6GM - J(16M_1 + M_3)\right] \\
S_{30} & = & 8JM_4 - 3GM_3 \\
S_{41} & = & -4S_{32}(6M_2 + M_4) - 16M_1S_{31} + 8MS_{30} \\
 & & +2\left[-JM_3(16M_1 - M_3) + 16M(M^2 - 6JM_2) - 32J^2M_5\right] \\
S_{40} & = & 6M_6S_{32} - (16M_1 + M_3)S_{30} - 8M(MM_3 - 6JM_6)
\end{array}
\tag{31}
$$

We consider the Sturm sequence $S$ with $S_0 = f_1(X)$ and $S_1 = f_2(X)$ when $J = 0$. The complete Sturm sequence is

$$
\begin{array}{rcl}
S_0(x) & = & f_1(X) \\
S_1(x) & = & f_2(X) \\
S_2(x) & = & 6G\,X^2 - 4M\,X + M_3 \\
S_3(x) & = & G\left[S_{31}\,X + S_{30}\right] \\
S_4(x) & = & -8M_5(S_{41} - M_3S_{31}) \\
 & & +32M_4(M_5S_{32} - M_4S_{30}) \\
 & & -12M_6(S_{40} - 2M_3S_{30}) \\
 & & +M_3^2(M_3^2 - 16MM_4 - 16JM_5 + 36GM_6)
\end{array}
\tag{32}
$$

Of course we must consider two more degenerate Sturm sequence. One for $J = G = 0$, one for $J = G = M = 0$ and the trivial one when $J = G = M = M_3 = 0$.

In order to simplify things we assume present an evaluation scheme for the complete Sturm sequence. We can treat all possible evaluations of the Sturm sequence as a binary tree, which has as nodes the evaluation of a term of the sequence and that branches according to the sign of the computed quantity. We have precomputed all the possible cases and we store the cases where we can decide the result of the sequence evaluation before we reach the bottom of the tree. In Algorithm 2 we can see the algorithm for comparing the two largest roots of the two quartics. We must say that this algorithm is automatically generated and that variable *where* allow us not to use nested *if*'s. The only thing the user must do by hand is to write the functions *COMPUTE*. Additionally the expression $where \hat{=} number$ means $where = where\ XOR\ number$.

The maximum algebraic degree involved in the coefficients of every Sturm sequence is the algebraic degree of the resultant of the two polynomials, which is 8. So in order to decide the maximum algebraic degree involved in the comparison of of the roots of two quartics we must consider the evaluation of the Sturm sequences on the endpoints of the isolating intervals of the roots.

**Theorem 5** *There is an algorithm that compares any two roots of two quartics using Sturm sequences and isolating intervals from Theorem 1 while the algebraic degree of the quantities involved is 14.*

**Proof.** In order to compare any two roots of two quartics we use the algorithms of Section 6. If want to use this algorithm we must provide isolating intervals for the roots of the quartics. For this we use the Theorem of Isolating polynomials and the results from Section 3.

Now the problem is that the endpoints of the isolating intervals are not rational numbers but algebraic ones of degree 2 in the general case, where the coefficients of the representing polynomial are of algebraic degree 2. So we have to evaluate the Sturm sequence on an algebraic number. But we only need the sign of this evaluation, which can be easily obtained as explained in Section 6. Hence there is an algorithm.

As for the maximum algebraic degree involved, we consider the most difficult case, which is to determine the sign of $S_4(X)$ over the algebraic number $\tau_1$ or $\tau_2$. Notice that the degree of the coefficients of $S_4(X)$ is 6. In order to decide the sign we must evaluate the polynomial of Equation 6 over the solution of the equation $S_4(X) = 0$. This evaluation involves algebraic degree 14, which is an upper bound for our algorithm.

Notice that $\deg \mathit{Resultant} = \deg S_5(X) = 8$. □

**Theorem 6 (Algebraic degree of the resultant)** *The resultant of two polynomials $P$ and $Q$ of degree $m$ and $n$ respectively, is a homogeneous polynomial in the coefficients of the polynomials with degree $m + n$. Additionally if the coefficients of $P$ and $Q$ have algebraic degree $p$ and $q$, then the algebraic degree of the resultant is $p\,n + q\,m$ ([?, ?, ?]).*

It is well known that the algebraic degree of the resultant provides a tight lower bound in order to find the common solutions of a system of two equations. In the case of two quartics, assume that one of the quartics has a multiple root and additionally that this root is $\frac{T_1}{T_2}$ (this case happens when the quartic has one double root, see Proposition 1). The algebraic degree of this quantity is 4, but we can express it as a sum of fractions and reduce its algebraic degree to 3. In other words, this rational number is the root of the degree one polynomial $T(X) = T_2\,X - T_1$, whose coefficients have algebraic degree 3.

If we form the resultant of $T(X)$ and a quartic, then by Theorem 6 its algebraic degree is 13. On the other hand the resultant of two quartics has algebraic degree 8. This lead us to the following claim.

**Claim 1 (Minimum condition of comparison)** *In order to compare the roots of two equations the bound on the algebraic degree provided by their resultant is not always tight. Or in other words, resultants are minimum conditions of solvability but* ***resultants are not minimum conditions of comparison****.*

If our claim and the above discussion is correct, then our algorithm for the comparison of the roots of two quartics is nearly optimal, since it has algebraic degree 14, while the lower bound is 13.

**Algorithm 2** Compare the two largest roots of two quartics

**Require:** Isolating interval representation of the two numbers

Find an common isolating interval of the form $[\frac{p}{q}, +\infty)$
Check if both numbers lie in a common interval
COMPUTE $S_2(+\infty)$
**if** $S_2(+\infty) > 0$ **then** where ^= 64
COMPUTE $S_2(\frac{p}{q})$
**if** $\left(S_2(\frac{p}{q}) \geq 0\right)$ **then** where ^= 32
COMPUTE $S_3(+\infty)$
**if** $(S_3(+\infty) > 0)$ **then** where ^= 16
COMPUTE $S_3(\frac{p}{q})$
**if** $\left(S_3(\frac{p}{q}) \geq 0\right)$ **then** where ^= 8
**if** $where \in \{16\}$ **then**
  RETURN SMALLER;
**end if**
**if** $where \in \{112\}$ **then**
  RETURN LARGER;
**end if**
COMPUTE $S_4(+\infty)$
**if** $(S_4(+\infty) > 0)$ **then** where ^= 4
**if** $where \in \{4, 24, 56, 68\}$ **then**
  RETURN SMALLER;
**end if**
**if** $where \in \{32, 92, 96, 124\}$ **then**
  RETURN LARGER;
**end if**
COMPUTE $S_4(\frac{p}{q})$
**if** $\left(S_4(\frac{p}{q}) \geq 0\right)$ **then** where ^= 2
**if** $where \in \{0, 14, 30, 46, 48, 62, 64, 78, 80, 110\}$ **then**
  RETURN SMALLER;
**end if**
**if** $where \in \{8, 38, 40, 54, 72, 86, 88, 102, 104, 120\}$ **then**
  RETURN LARGER;
**end if**
COMPUTE $S_5$
**if** $(S_5 > 0)$ **then** where ^= 1
**if** $where \in \{3, 11, 12, 28, 36, 43, 44, 51, 52, 60, 67, 75, 76, 83, 84, 91, 100, 107, 108, 123\}$ **then**
  RETURN SMALLER;
**end if**
**if** $where \in \{2, 10, 13, 29, 37, 42, 45, 50, 53, 61, 66, 74, 77, 82, 85, 90, 101, 106, 109, 122\}$ **then**
  RETURN LARGER;
**end if**
RETURN EQUAL;

## 8 Quartic-Quadratic

In this section we provide the Sturm sequences that we need in order to compare the roots of a quartic and a quadratic. We assume two equations

$$\begin{aligned} f_1(X) &= a_1X^4 - 4b_1X^3 + 6c_1X^2 - 4d_1X + e_1 \\ f_2(X) &= a_2X^2 - 2b_2X + c_2 \end{aligned}$$

We consider the Sturm sequence $S$ with $S_0 = f_1(x)$ and $S_1 = f_2(X)$. The complete Sturm sequence is

$$\begin{aligned} S_0(X) &= f_1(X) \\ S_1(X) &= f_2(X) \\ S_2(X) &= S_{21}X + S_{20} \\ S_3(X) &= -(\phi^2 - 16\,\Sigma\,\Sigma_p) \end{aligned}$$

In order to reduce the computational cost, we need the quantities

$$\begin{aligned} N_1 &= a_2d_1 - b_2c_1 \\ N_2 &= c_1c_2 - a_2e_1 \\ N_3 &= a_2e_1 - b_2d_1 \\ N_4 &= b_1b_2 - a_2c_1 \\ G &= a_1c_2 - a_2c_1 \\ J_1 &= b_1c_2 - b_2c_1 \\ W_1 &= a_1d_1 - b_1c_1 \\ W_2 &= b_1e_1 - c_1d_1 \\ \Delta_{12} &= b_1^2 - a_1c_1 \\ \Delta_{13} &= c_1^2 - b_1d_1 \\ \Delta_{14} &= d_1^2 - c_1e_1 \\ \Delta_{15} &= c_1^2 - a_1e_1 \\ \Delta_{12} &= b_2^2 - a_2c_2 \end{aligned}$$

and so the involved quantities in the Sturm sequence are:

$$\begin{aligned} \phi &= -4\,b_2J_1 + c_2G + a_2\,(3\,N_2 + 4N_3) \\ \Sigma &= a_2\,(a_2\Delta_{14} + 2\,b_2W_2) \\ &\quad +b_2\,(b_2\Delta_{15} + 2\,c_2W_1) + c_2\,(c_2\Delta_{12} + 2\,a_2\Delta_{13}) \\ \Sigma_p &= \Delta_{22} \\ S_{21} &= -4\,J(b_2^2 + \Delta_{22}) + 4\,a_2(2b_2N_4 + a_2N_1) \\ S_{20} &= c_2(a_1\Delta_{22} + 3\,b_2J) + a_2(-5\,c_2N_4 + a_2N_2). \end{aligned}$$

## 9 Quartic-Cubic

In this section we provide the Sturm sequences that we need in order to compare the roots of a quartic and a cubic. At this point we must say that despite the fact that the factorization of the quantities that appear in the various Sturm sequences seams a very difficult task, there is a way to make this procedure easier. At first we consider the Bezoutian matrix of two polynomials and then we can easily verify that every coefficient in the Sturm sequences is a combination of the elements of the Bezoutian matrix. We consider two equations

$$\begin{aligned} f_1(X) &= a_1X^4 + b_1X^3 + c_1X^2 + d_1X + e_1 \\ f_2(X) &= a_2X^3 + b_2X^2 + c_2X + d_1 \end{aligned}$$

We consider the Sturm sequence $S$ with $S_0 = f_1(x)$ and $S_1 = f_2(X)$. The complete Sturm sequence is

$$\begin{aligned} S_0(X) &= f_1(X) \\ S_1(X) &= f_2(X) \\ S_2(X) &= S_{22}X^2 + S_{21}X + S_{20} \\ S_3(X) &= S_{31}X^2 + S_{30} \\ S_4(X) &= \frac{-S_{31}^2 S_{20} + S_{30}(S_{21}S_{31} - S_{30}S_{22})}{S_{22}^2} \end{aligned}$$

In order to reduce the computational cost, we need the quantities

$$\begin{aligned} S_{22} &= a_2\, G - b_2\, J \\ S_{21} &= a_2\, M - c_2\, J \\ S_{20} &= a_2\, K_1 - b_2\, M \\ S_{31} &= S_{22}\, K_2 - M\,(2\, S_{21} - a_2\, M) + c_2^2\,(b_1\, J - a_1\, G) \\ S_{30} &= -S_{20}\,(J_1 + M) - b_2\, e_1(S_{22} + a_2\, G) - d_2\, G^2 \\ M &= a_1\, d_2 - a_2\, d_1 \\ G &= a_1\, c_2 - a_2\, c_1 \\ J &= a_1\, b_2 - a_2\, b_1 \\ J_1 &= b_1\, c_2 - b_2\, c_1 \\ K_1 &= d_2\, b_1 - e_1\, a_2 - d_1\, b_2 \\ K_2 &= c_1\, c_2 - b_2\, d_1 + b_1\, d_2 - a_2\, e_1 \end{aligned}$$

## 10 Representations for the arrangement of conic arcs

Our representations are those developed for the Curved Kernel, see [21]. We consider conic sections in the general form:

$$f(x, y) = r\, x^2 + s\, y^2 + t\, x\, y + u\, x + v\, y + w. \tag{33}$$

More particularly:

- A conic section is a curve, provided by the Curved Kernel, represented by a polynomial in 2 variables as in (33). We assume that this is polynomial always contains at least one quadratic term. At certain points below, we may consider only the case of ellipses, but generalizing to arbitrary conic sections should be straightforward.
- A conic arc is $x$-monotone, unless it is the input to **make_x_monotone**. It is represented by a supporting conic section and, in the former case, a boolean indicating whether it lies on the upper or lower part of the curve.

- An arc's endpoint is represented as the intersection of 2 conic arcs and its $x, y$ coordinates. These correspond to algebraic numbers of degree 4 expressed by Root-Of-4 structures; this does not preclude the possibility to have rational or quadratic numbers, whenever possible. The ordinate $y$ may be expressed parametrically by a univariate polynomial in $y$, namely $A(x)y + B(x)$, whose coefficients are themselves polynomials in $x$ of degrees 1 and 2 resp. At present, we assume that only $x$ is always given as a Root-Of-4, while $y$ has either parametric or Root-Of-4 expression.

It might be possible to alternate between the 2 representations of $y$, given that of $x$, but for now it does not seem straightforward in all cases. One example of the parametric representation is given at (40) and (41). Remark, that it is not always to obtain a linear polynomial in $y$: for orthogonal ellipses with one tangential intersection, there are 2 double roots for the resultant in $x$. The resultant in $y$ has one double root and two simple roots. For the simple roots, the parametric expression is of the form $Ay^2 + B(x)$, where $A$ is a constant and $B(x)$ is linear.

# 11 Predicates for arrangements of conic arcs

In this section we will use the results from Section 6 in order to derive certain important predicates that we need for the arrangement of conic arcs. Our strategy is to reduce certain predicates to others, so as to reduce the number of predicates that must be implemented from scratch. The paradigm we have in mind is the sweep-algorithm with a vertical sweepline. We plan a CGAL-like implementation of these predicates, aiming at integrating them with the Curved Kernel.

## 11.1 compare_x and compare_y

If the ordinate is represented by $Ay^2 + B(x)$, then we need a comparison of quadratic roots in an extension field.

If we assume that the two ordinates are given by $Ay + B = 0, A'y + B' = 0$, the comparison of 2 ordinates reduces to determining the signs of $A, A'$ and testing the sign of $AB' - A'B$ at the proper real value of $x$. If either of $A, A'$ equals 0, then there are infinite solutions for $y$ and the conics of the geometric problem are overlapping.

Polynomial $AB' - A'B$ is univariate and has degree (at most) 3 in $x$, because $\deg A(x) = 1, \deg B(x) = 2$. Since $x$ is a root of a quartic, the sign computation can be solved by the above methods, in particular those of comparing roots of a quartic and a cubic. If the degrees of $A, A', B, B'$ are smaller than 1 and 2, respectively, then the sign computation is simplified accordingly. The smallest degrees occur when $A$ or $A'$ is constant and $B$ or $B'$ is zero.

The comparison of abscissae reduces to comparing specific roots of polynomials of degree at most 4, which is solved above.

## 11.2 make_x_monotone

In order to cut a conic to monotone arcs we must find its points where the tangent is a vertical line. We take the derivatives with respect to $x$ and $y$. These are:

$$\begin{aligned} f_x &= 2\,r\,x + t\,y + u, \\ f_y &= 2\,s\,y + t\,x + v. \end{aligned}$$

The common points of $f$ and $f_y$ are the points of the conic that have vertical tangents. In order to specify the abscissae of these points we take the resultant of $f$ and $f_y$ by eliminating $y$.

$$R_x = Res_y(f, f_y) = s(A_1 x^2 - 2B_1 x + C_1) \tag{34}$$

where

$$\begin{array}{rcl} A_1 & = & 4\,s\,r - t^2 \\ B_1 & = & v\,t - 2\,s\,u \\ C_1 & = & -v^2 + 4\,s\,w \end{array} \tag{35}$$

We can forget the $s$ factor if we are considering only ellipses, so always $s \neq 0$.

In order to specify the ordinates of the points of interest we consider the resultant of $f$ and $f_x$ by eliminating $x$. This is

$$R_y = Res_x(f, f_x) = r(A_2 y^2 - 2B_2 y + C_2) \tag{36}$$

$$\begin{array}{rcl} A_2 & = & A_1 \\ B_2 & = & -2\,r\,v + t\,u \\ C_2 & = & 4\,r\,w - u^2 \end{array} \tag{37}$$

We might assume that $r \neq 0$ if we restricted attention to ellipses.

From the two resultants, we obtain the abscissae and the ordinates of the tangential points, but we have to find the correspondence between them. This can be done easily if we consider the slope of the line $f_y = 0$. The slope is $-\frac{t}{2\,s}$. There are 3 cases:

- If the slope is negative, i.e. $st > 0$ in the case of ellipses, then the biggest root of $R_x$ corresponds to the biggest root of $R_y$.
- If the slope is positive, i.e. $st < 0$, then the biggest root of $R_x$ corresponds to the smallest root of $R_y$.
- If it is zero, i.e. $t = 0$, then this means that $f_y = 0$ is a horizontal line and that $R_y$ has one double root, which is $-\frac{v}{2\,s}$.

Now that we specified the two tangential points we can split the ellipse to two monotone arcs, the *upper* and the *lower* part.

### 11.3 nearest_intersection_to_right

Given are two conic arcs and a point $\Gamma$. We wish to find the first intersection of the arcs to the right of $\Gamma$. Special cases include that $\Gamma$ be an intersection, in which case we return itself. When the two arcs overlap, we return their common arc, defined by two endpoints. These are identified by straightforward tests, similar to those in the case of circular arcs.

Let us take the general case, and regard curves instead of arcs, for now. We construct all of their intersections by applying **solve**, then apply comparisons between $\Gamma$ and these roots.

We have solved the problem concerning 2 conic curves, but we still need to limit our search among the intersection points of the *arcs*. The constructor of intersections as endpoints may assign the information on whether this point lies on the upper or lower part of the curve, so it would suffice to call **in_x_range**. Otherwise, we apply **is_on_arc**.

*The comparisons of $\gamma_x$ with the abscissae of the intersections shall, internally, optimize execution by using the isolating (also called "separating") points between the roots of the resultant. These are algebraic numbers of degree up to 2 and, in practice, rational points. If we were to do this explicitly by hand, we would apply binary search of $\Gamma$ among the isolating points, where the basic test is a comparison between the x-coordinates of an isolating point and $\gamma_x$. Let the isolating points be $s_0 < s_1 < s_2$, let the real roots be $r_i$, and let us consider the case of 4 real roots, so $i = 0, \ldots, 3$. In summary, we would need two comparisons of $\gamma_x$ with isolating points, and one comparison between $\gamma_x$ and the abscissa of some root of the resultant. Hence two comparisons between algebraic numbers of degree 4 and 2, and one comparison between two algebraic numbers of degree 4. All of these comparisons have been described in the algorithms above.*

**Algorithm 3** Minor subroutine for **nearest_intersection_to_right**: Position $\gamma_x$ among roots using isolating points, where $r_0 < s_0 < r_1 < s_1 < r_2 < s_2 < r_3$.

```
if γx < s1 then
  if γx < s0 then
    if γx < r0 then RETURN r0 else RETURN r1
  else
    if γx < r1 then RETURN r1 else RETURN r2
  end if
else
  if γx < s2 then
    if γx < r2 then RETURN r2 else RETURN r3
  else
    if γx < r3 then RETURN r3 else RETURN No intersection to the right
  end if
end if
```

### 11.4 is_on_arc

Given an arc of curve $f$ and a point on $f$, decide whether the point lies on the arc. First, check **in_x_range**. Then, the problem reduces to deciding whether the point lies on the upper or lower part of $f$, i.e. comparing specific $y$ roots of $f_y(\gamma_x, y) = t\gamma_x + 2sy + v$, and $R_y(y)$, defining the ordinate of $y$. This is a comparison of algebraic numbers of degrees 1 and 4 over $\mathbb{Q}[\gamma_x]$. The heaviest test is a call of **sign_at** on $R_y(-(t\gamma_x/2s) - v/2s)$; which is a polynomial of degree 4 in $\gamma_x$. So we apply **compare** on two polynomials of degree 4, the second one being $R_x(\gamma_x)$ defining $\gamma_x$.

*Alternatively, we may use the representation of intersection points by a pair of a quartic root, for the abscissa, and an expression $Ay + B$ for the ordinate, where $A, B$ are x-polynomials of degree 1 and 2 respectively. The question is to decide, given such a point, whether it lies on the given arc or not. Once we decide whether the point lies on the upper or lower part of the conic, it suffices to test the x-range of the arc. So, our problem is reduced to deciding whether a point of the form [Root-of-4, $Ay + B$] lies on the upper or lower part of a conic with equation $f(x, y)$. Equivalently, we must compare the root of $A(\gamma_x)y + B(\gamma_x)$ against the root of $f_y(\gamma_x, y)$, which is linear in $y$ and $\gamma_x$. If we write $f_y(\gamma_x, y) = Cy + D(\gamma_x)$, we know that $C$ is a constant and $D(\gamma_x)$ is linear. Hence, it suffices to test the sign of $B(\gamma_x)C - A(\gamma_x)D(\gamma_x)$, which is (at most) quadratic in $\gamma_x$, at the proper root of the quartic expressing $\gamma_x$. This is similar to comparing the root of a quartic and a quadratic, presented in section 8.*

### 11.5 compare_y_to_right

Given are two conic arcs on curves $g_1, g_2$, and one of their intersection points $\Gamma$. Point $\Gamma$ may be defined as the intersection of two other curves. The predicate decides which arc is above and which is below, immediately to the right of $\Gamma$. Our method is to test the vertical ordering of the 2 arcs at some convenient test point to the right of the given intersection, see figure 2.

This test point must also be to the left of the next intersection, if any, between the two given *arcs*, so it must lie to the left of the next intersection between the two given conic *curves*. This is equivalent to picking a point between $\Gamma$ and the next real root, to the right, of the resultant $R_x$ of the two conics. In other words, we search for a point whose abscissa isolates the abscissae of $\Gamma$ and the next root of the resultant, provided there is another root to the right.

If such a root does not exist, we have no guarantee that the isolating point will be useful. In this case, we can use one endpoint of some arc and call **compare_y_at_x** on the other arc and this endpoint. In the rest of this discussion, we assume a valid isolating point exists.

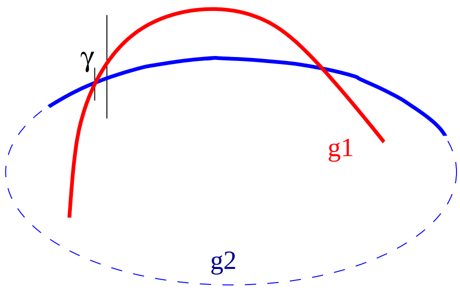


Figure 2: Predicate compare_y_to_right, with the given point marked by $\gamma$ and the test point marked by a longer vertical segment.

Since the resultant is a polynomial in $x$ of degree at most 4, we have described how to compute isolating points in the previous sections. Let $s$ be an adequate isolating point, and $L$ the line $y - s$. It is possible to define the intersection point, call it $q$, of $L$ with one of the curves, say $g_2$, using the algorithms implementing **make_x_monotone**. The abscissa of $q$ is the root of a quadratic or linear polynomial. Assuming that its ordinate is expressed by a linear polynomial in $(\mathbb{Q}[x])[y]$ (ie. with coefficients which are polynomials in $x$), it remains to apply **compare_y_at_x** on $q$ and $g_1$.

An alternative method specializes the equations of $g_1, g_2$ with $x \mapsto s$, thus yielding quadratic polynomials in $y$. Their coefficients are rational, assuming any isolating point is rational. By considering whether the given arcs lie on the upper or lower part of $g_i$, we can focus on the larger or smaller root of the respective quadratic equation. It now suffices to compare these roots of the two quadratics, using some known algorithm, e.g. see [12]. In short, this method requires 2 specializations and a single comparison of quadratic roots.

### 11.6 compare_y_at_x

In this predicate we need to decide whether a given conic arc is above or below a given point, which is defined as a specific intersection of two other conic arcs $g_1, g_2$. Denote the given point by $\Gamma(\gamma_x, \gamma_y)$, where its coordinates can be expressed by roots of quartics.

The supporting conic of the conic arc is of the form of Equation (33). We know in advance if the arc is on the upper or the lower part. Suppose that the arc is on the upper part. See, e.g., figure 3.

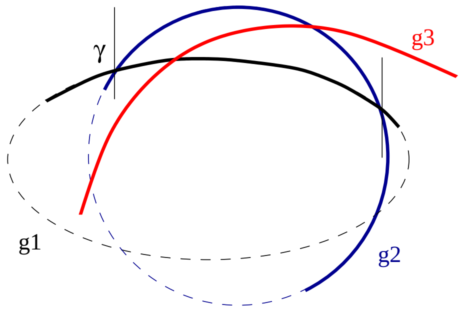


Figure 3: Predicate compare_y_at_x with the given point marked by $\gamma$ and the query (red) arc denoted by $g_3$.

If we set $x = \gamma_x$ to Equation (33), this becomes

$$g(y) = A\,y^2 + B\,y + C \tag{38}$$

where

$$\begin{array}{rcl} A & = & s \\ B & = & t\gamma_x + v \\ C & = & r{\gamma_x}^2 + u\gamma_x + w \end{array}$$

Since we consider only the upper part we can infer that the ordinate of the arc's point with abscissa $\gamma_x$ is the largest root (hence with index 1) of the polynomial $g$. We denote this by

$$y_1 = (g, 1).$$

If the given arc were on the lower part, we would use $(g, 0)$.

Now, there are two possibilities concerning the representation of $\gamma_y$. The first is to use its representation as a root of a quartic. Then, we are led to compare specific roots of a quartic and a quadratic, as in section 8. However, the quadratic polynomial, namely $g$, has coefficients in the extension field containing $\gamma_x$. So, every sign test involving $\gamma_x$ shall reduce to computing the sign of a polynomial over the rationals at the root of the quartic polynomial defining $\gamma_x$. This can be done with the Sturm-based techniques of Section 6.

In what follows, we use the following representation for $\Gamma$: [Root-of-4, $Ay + B$]. Suppose that the intersecting conics are:

$$\begin{array}{rcl} g_1 & = & r_1x^2 + s_1y^2 + t_1xy + u_1x + v_1y + w_1 \\ g_2 & = & r_2x^2 + s_2y^2 + t_2xy + u_2x + v_2y + w_2 \end{array} \tag{39}$$

Then $\gamma_x$ is represented by the resultant of $g_1$ and $g_2$ with respect to $y$, which is a quartic polynomial, and an index that denotes which root of the quartic we need. Assume that the resultant is

$$R_x = Res(g_1, g_2, y) = d_4x^4 + d_3x^3 + d_2x^2 + d_1x + d_0$$

The coefficients of the resultant are of algebraic degree 4. In order to find the ordinate of the intersection point, we let $x = \gamma_x$ in Equations (39) and form the equation

$$\begin{aligned} Q_1(y) &= s_2 g_1(\gamma_x, y) - s_1 g_2(\gamma_x, y) \\ &= A_1 y + B_1 \end{aligned} \tag{40}$$

where

$$\begin{aligned} A_1 &= (s_1 t_2 - s_2 t_1)\,\gamma_x + s_1 v_2 - s_2 v_1 \\ B_1 &= (s_1 r_2 - s_2 r_1)\,{\gamma_x}^2 + (s_1 u_2 - s_2 u_1)\,\gamma_x + s_1 w_2 - s_2 w_1 \end{aligned} \tag{41}$$

In order to find $\gamma_y$ we must solve $Q_1(y) = 0$, which is of degree 1, with respect to $y$. By our notation this is

$$\gamma_y = (Q_1, 0).$$

In order to decide the predicate we compare $y_1$ and $\gamma_y$. The equation that represents $y_1$ is of degree 2 and the equation that represents $\gamma_y$ is of degree 1 (the reader can refer to [12] in order to see the treatment of such a case). The first thing to do is to compare $\gamma_y$ with the apex of the Equation (38), which is $-\frac{B}{2A}$. This is equivalent to testing the sign of the quantity

$$\begin{aligned} J &= A_1\,B - 2\,A\,B_1 \\ &= k_2 \gamma_x^2 + k_1 \gamma_x + k_0 \end{aligned}$$

where

$$\begin{aligned} k_2 &= 2\,s\,s_2\,r_1 + t\,s_1\,t_2 - t\,s_2\,t_1 - 2\,s\,s_1\,r_2, \\ k_1 &= -2\,s\,s_1\,u_2 + 2\,s\,s_2\,u_1 - t\,s_2\,v_1 + v\,s_1\,t_2 - v\,s_2\,t_1 + t\,s_1 v_2, \\ k_0 &= v\,s_2\,v_1 + 2\,s\,s_2\,w_1 - 2\,s\,s_1 w_2 + v\,s_1\,v_2. \end{aligned}$$

In order to test the sign of $J$ we find the sign of the polynomial $J(x)$ over the algebraic number $\gamma_x$. This can be done as explained at Section 6.

If $J \leq 0$ we are done ($y_1$ is ABOVE). If $J > 0$ then we must test the sign of $g(\gamma_y)$. This is equivalent to testing the sign of the quantity

$$F = L_4 \gamma_x^4 + L_3 \gamma_x^3 + L_2 \gamma_x^2 + L_1 \gamma_x + L_0$$

where

$$
\begin{aligned}
L_4 &= ss_1{}^2r_2{}^2 - ts_2{}^2r_1t_1 + rs_1{}^2t_2{}^2 + rs_2{}^2t_1{}^2 - 2\,ss_1r_2s_2r_1 + ss_2{}^2r_1{}^2 \\
&\quad -ts_1{}^2r_2t_2 + ts_1r_2s_2t_1 - 2\,rs_1t_2s_2t_1 + ts_2r_1s_1t_2 \\
L_3 &= ts_2r_1s_1v_2 + 2\,ss_2{}^2r_1u_1 - 2\,ss_2r_1s_1u_2 - ts_2{}^2r_1v_1 \\
&\quad +2\,rs_2{}^2t_1v_1 - ts_2{}^2u_1t_1 + 2\,rs_1{}^2t_2v_2 - 2\,rs_1t_2s_2v_1 \\
&\quad +ts_2u_1s_1t_2 - vs_2{}^2r_1t_1 - ts_1{}^2u_2t_2 + ts_1u_2s_2t_1 \\
&\quad -2\,rs_2t_1s_1v_2 + 2\,ss_1{}^2r_2u_2 + us_1{}^2t_2{}^2 + us_2{}^2t_1{}^2 \\
&\quad +vs_2r_1s_1t_2 + vs_1r_2s_2t_1 - ts_1{}^2r_2v_2 + ts_1r_2s_2v_1 \\
&\quad -2\,ss_1r_2s_2u_1 - 2\,us_1t_2s_2t_1 - vs_1{}^2r_2t_2 \\
L_2 &= 2\,ss_1{}^2r_2w_2 + 2\,ss_2{}^2r_1w_1 - ts_1{}^2u_2v_2 - ts_2{}^2u_1v_1 \\
&\quad -ts_1{}^2w_2t_2 - ts_2{}^2w_1t_1 - vs_1{}^2r_2v_2 + rs_1{}^2v_2{}^2 \\
&\quad +rs_2{}^2v_1{}^2 + ws_1{}^2t_2{}^2 + ws_2{}^2t_1{}^2 + ss_1{}^2u_2{}^2 \\
&\quad +ss_2{}^2u_1{}^2 - 2\,ss_1r_2s_2w_1 - 2\,ss_2r_1s_1w_2 - 2\,ss_1u_2s_2u_1 \\
&\quad -vs_1{}^2u_2t_2 + ts_1u_2s_2v_1 - vs_2{}^2r_1v_1 + ts_2w_1s_1t_2 \\
&\quad +vs_1r_2s_2v_1 + ts_2u_1s_1v_2 + ts_1w_2s_2t_1 + vs_1u_2s_2t_1 \\
&\quad -vs_2{}^2u_1t_1 + vs_2r_1s_1v_2 - 2\,rs_1v_2s_2v_1 - 2\,ws_1t_2s_2t_1 \\
&\quad +2\,us_1{}^2t_2v_2 + 2\,us_2{}^2t_1v_1 + vs_2u_1s_1t_2 - 2\,us_1t_2s_2v_1 \\
&\quad -2\,us_2t_1s_1v_2 \\
L_1 &= -vs_1{}^2u_2v_2 - 2\,ss_1u_2s_2w_1 - 2\,us_1v_2s_2v_1 - 2\,ws_2t_1s_1v_2 \\
&\quad +vs_1u_2s_2v_1 + ts_2w_1s_1v_2 + 2\,ss_1{}^2u_2w_2 + us_2{}^2v_1{}^2 \\
&\quad +2\,ws_2{}^2t_1v_1 - vs_2{}^2w_1t_1 + us_1{}^2v_2{}^2 + ts_1w_2s_2v_1 \\
&\quad -2\,ws_1t_2s_2v_1 + 2\,ss_2{}^2u_1w_1 - 2\,ss_2u_1s_1w_2 - ts_2{}^2w_1v_1 \\
&\quad +vs_1w_2s_2t_1 + 2\,ws_1{}^2t_2v_2 + vs_2w_1s_1t_2 - vs_1{}^2w_2t_2 \\
&\quad -vs_2{}^2u_1v_1 - ts_1{}^2w_2v_2 + vs_2u_1s_1v_2 \\
L_0 &= -vs_1{}^2w_2v_2 - 2\,ss_1w_2s_2w_1 - 2\,ws_1v_2s_2v_1 + ss_2{}^2w_1{}^2 \\
&\quad +vs_1w_2s_2v_1 + ws_2{}^2v_1{}^2 + ws_1{}^2v_2{}^2 + vs_2w_1s_1v_2 \\
&\quad -vs_2{}^2w_1v_1 + ss_1{}^2w_2{}^2.
\end{aligned}
$$

In order to find the sign of $F$ we must find the sign of the quartic $L(x) = L_4x^4 + L_3x^3 + L_2x^2 + L_1x + L_0$ over the algebraic number $\gamma_x$. This can be done as explained at Section 6.

To summarize the results for this predicate, we can decide it using two comparisons of roots, in the most difficult case. Considering the respective polynomials, we have one comparison of a quartic and a quadratic and one comparison of two quartics.

# 12 Functions for solving

## 12.1 solve

Given 2 conics, we wish to express all real intersection points using Root-Of coordinates. It is straightforward to obtain the abscissae and ordinates as Root-Of-4 coordinates, by computing the univariate resultants in $x$ and $y$, respectively. The main problem is *matching* these algebraic numbers.

Currently we are working in an efficient implementation of this function that treats all the cases in a unified way.

This procedure may also decide whether the endpoints lie on the upper or lower part of the curve. This is possible precisely when more than one intersections have the same $x$-root. If so, this information is stored with the endpoint.

Let us denote the different cases by the multiplicity of real roots of the resultants in $x$ and $y$. For instance, the case $(2,1,1;2,2)$ corresponds to one double and two simple $x$-roots, and two double $y$-roots. Notice that multiple roots may be due to an intersection of high multiplicity to simple intersections with the same coordinate, or to both things happening at the same time. A star $(*)$ indicates any valid integer. Since $x$ and $y$ are interchangeable, we discuss only have of the actual cases.

Case of some complex roots. If all roots are complex, there is nothing to be done. Assume there exists a pair of complex roots. The cases $(2;2)$ and $(2;1,1)$ are trivial. In the second case, we also decide the (upper or lower) part of the curve. The case $(1,1;1,1)$ shall be solved by subroutine **solve_simple**. We cannot decide the part of the curve. In the rest, we assume all roots are real.

Case of a root with multiplicity 4, namely $(4;2,2)$: Trivial. It also decides the upper / lower part of the curve.

Case of one triple root, say in the resultant $R(x)$. The case $(3,1;1,1,1,1)$ is infeasible because it implies that the vertical line yielding the triple root of the resultant in $x$ intersects each conic at 3 simple points.

Case $(3,1;3,1)$: This case corresponds to the following two subcases

$$\begin{array}{c|cc} & & \\ 1 & 1 & 0 \\ 3 & 2 & 1 \\ \hline & 1 & 3 \end{array} \qquad \begin{array}{c|cc} & & \\ 1 & 0 & 1 \\ 3 & 3 & 0 \\ \hline & 3 & 1 \end{array} \tag{42}$$

It is sufficient to test the upper right box in order to deduce the intersection points.

$(3,1;2,1,1)$ The case is shown above in the 2nd table in (42). The vertical line at $x = \gamma_3$ intersects each conic at most twice, hence we are certain to have a double and two simple intersections. We have $k \in \{0,1\}$ so that exactly two $k$ values equal 1, either on the diagonal or the anti-diagonal. Since every conic has at most two intersections with the vertical line $x = \gamma_3$, it suffices to test whether the specialization of one of the given conics, say $f(\gamma_3, y)$, vanishes at some candidate simple root. This decision reduces to calling **sign_at** on $f(\gamma_3, y)$ and a fourth-degree algebraic number. [1]

$(3,1;2,2)$ It is shown in the 3rd table in (42). Notice that $f(\gamma_1, y), g(\gamma_1, y)$ have at most one root in each $y$-interval. It suffices to call two times **sign_at**, namely on $f(\gamma_1, y), R(y)$ and on $g(\gamma_1, y), R(y)$, for some particular root of $R(y)$ (either root can be used here) where $R$ is quadratic. [2]

Case of 4 simple $x$-roots. This implies there are 4 simple intersections; there are 3 cases. In none of these cases can we decide the upper / lower part of the curve.

- Cases $(1,1,1,1;1,1,1,1)$ and $(1,1,1,1;2,2)$. Solved by repeated application of **solve_simple** on the two given conics.

[1] Alternatively, placing the $k$ values reduces to calling **solve_simple** on any one of the candidate simple roots.

[2] Alternatively, we decide with one call to **solve_simple** applied to any one of the candidate roots projecting to $\gamma_1$.

- Case $(1,1,1,1;2,1,1)$. One can use the fact that the double $y$-root is rational, call it $\gamma_y$, and apply **sign_at** on $f(x,\gamma_y), R(x)$, then on $g(x,\gamma_y), R(x)$. This test applies to at most three of the candidate roots projecting to $\gamma_y$. Then, one application of **solve_simple** is needed to completely decide the simple roots (like in the cases above where we had to find the two 0 values of $k$).

Case of at least one double $x$-root. When there is exactly one double root and, hence, another two simple roots in $R(x)$, it is possible to express the double root as a rational $\gamma_x$. There are three cases shown below:

$$
\begin{array}{c|ccc}
 & & & \\
2 & 1/2/0 & k/0/1 & k/0/1 \\
2 & 1/0/2 & k/1/0 & k/1/0 \\
\hline
 & 2 & 1 & 1
\end{array}
\qquad
\begin{array}{c|ccc}
2 & 2/0 & 0/1 & 0/1 \\
1 & 0/1 & k/0 & k/0 \\
1 & 0/1 & k/0 & k/0 \\
\hline
 & 2 & 1 & 1
\end{array}
\qquad
\begin{array}{c|cc}
 & & \\
2 & 1/m & 1/m \\
2 & 1/m & 1/m \\
\hline
 & 2 & 2
\end{array}
\tag{43}
$$

$(2,1,1;2,2)$ This case is shown in the first table above. We need to examine at most three candidate roots with the same $y$-coordinate. First, apply **sign_at** on $f(\gamma_x,y), R(y)$ and $g(\gamma_x,y), R(y)$, where $R$ is quadratic. If either is nonzero, then the matching is solved. Repeat with **sign_at** on $f(\gamma_x,y), R(y)$ and $g(\gamma_x,y), R(y)$, using the other $y$-root expressed by $R$. If either application yields nonzero, then the matching is solved.

Otherwise, there are two simple roots projecting to $\gamma_x$. It suffices now to call **solve_simple** on any one candidate simple root. in order to decide the $k$ values, where exactly two $k$ values are 1.

$(2,1,1;2,1,1)$ Exploit the double rational $y$-root, denoted by $\gamma_y$, and test whether $f(\gamma_x,\gamma_y) = g(\gamma_x,\gamma_y) = 0$ on $\mathbb{Q}$. If not, the matching is trivial. Otherwise, apply **solve_simple** to one of the candidate simple roots.

$(2,2;2,2)$ This is the last shown table above. In order to decide whether we are in the case involving $m$ or not, we can use **solve_simple** on one of the four squares. If we find no simple root there, then we consider $m \in \{0,2\}$, where there are precisely two nonzero values either on the diagonal or on the anti-diagonal. To choose between the two possibilities, we may apply a new function

$$\textbf{sign\_at}\,(f(x,y),\alpha,\beta)\,,$$

which is stronger than the function with the same name used above (overloading). It shall be applied on $f, g$, the two conics, using each time the algebraic numbers that represent the $x$- and $y$-coordinates of some candidate point. The candidate match is a root iff both function calls return 0.

In the implementation, we use the fact that all (double) roots of both resultants can be expressed as quadratic algebraic numbers; let them be denoted by $\alpha, \beta$. Moreover, we can use the defining polynomial $R_1(\alpha)$ to substitute $\alpha^2$ by a linear function in $\alpha$ in $f$. Similarly for $\beta$ and, therefore, $f(\alpha,\beta)$ becomes linear in both algebraic numbers.

So, function **sign_at** can be implemented with the available Sturm sequences. The overall computation is equivalent to comparing roots of polynomials of degree 1 and 2, of $f(\alpha,y)$ with a specified root of quadratic polynomial $R_2(y)$. Both quantities in the Sturm sequence lie in $\mathbb{Q}[\alpha]$, so each sign can be computed with the previous, lighter version of **sign_at**$(\phi(x), R_1(x))$, where $R_1(x)$ is a quadratic polynomial. In fact, we encounter a linear $\phi$ and another quadratic $\phi$: since the highest degree in the overall sequence is quadratic in the (algebraic) coefficients of $f$, and they are themselves linear in $\alpha$, then the most expensive sign computation involves polynomials of degrees 2 and 2.

Recall that an alternative approach [**?**, **?**] uses the Jacobi curve to decide this case.

### 12.2 solve_simple

This function, given two bivariate polynomials of total degree at most 2 and a square in $\mathbb{Q}^2$, decides whether these polynomials have a common real root inside the square or not. It is assumed that the given polynomials have at most one real intersection in the box. This intersection can be either simple or double.

Intersections with the square's corners can be ignored. We shall consider the intersections of each polynomial with the boundary of the square, and shall use any sequence of these intersections around the square.

**Lemma 2** *Consider two bivariate polynomials of degree $\leq 2$, with at most one real common root inside a given square. Then, they have a simple common root iff their intersections with the square boundary alternate exactly once, when ordered around the square, starting at any point on the boundary. If there are no alternations, then there is either a double root or no root at all.*

A single alternation means that, if we delete all successive intersections of the same conic with the boundary, then what remains is the pattern $\bullet, \circ, \bullet, \circ$, where the $\bullet$ and $\circ$ stand for the intersections of each conic with the boundary. In the opposite case (double or no root), the intersections have a pattern of the type $\bullet, \circ, \circ, \bullet, \bullet, \circ, \circ, \bullet$.

It is clear that testing the lemma for a given square is straightforward with the tools we have developed before. In particular, we may apply the constructor of each root of quadratic $f(x, \gamma_y) \in \mathbb{Q}[x]$, then keep those lying on the interval $x \in (a, b)$. Alternatively, a Sturm sequence of $f(x, \gamma_y), f'(x, \gamma_y) \in \mathbb{Q}[x]$ on $(a, b)$ yields the intersections on the edge defined by vertices $(a, \gamma_y)$ and $(b, \gamma_y)$ of the square.

For edges containing intersections of both conics, it is enough to apply **stl::sort** on the corresponding algebraic numbers, represented as Root-Of. The reason is that the **compare** function, detailed in previous sections, has enabled the implementation of the inequality operators. Now we have complete information on the intersection points with the 4 edges, which allows us to decide whether there is any alternation or not.

## 13 Sign_at functions

Using the results of Subsection 6.1 we can easily implement function `sign_at(Poly` $f$`, RootOf` $a$`)`, which computes the sign of a function $f$ evaluated over an algebraic number $\alpha$.

Assume that we want to compute the sign of of a univariate polynomial $f(X)$ evaluated over

$$f(X, Y) = r\, X^2 + s\, Y^2 + t\, X\, Y + u\, X + v\, Y + w. \tag{44}$$

evaluated over two algebraic numbers $\gamma_x$ and $\gamma_y$, where at least one of them is of algebraic degree at most 4 and the other of algebraic degree at most 4. Without loss of generality, let $\gamma_x$ be of degree

2 and $\gamma_y$ of degree 4. Further $\gamma_x$ and $\gamma_y$ are represented by polynomials $P_x$ and $P_y$ and isolating intervals $I_x$ and $I_y$ respectively.

We consider the univariate polynomial $F(X) = f(X, \gamma_y)$. So the problem now is to find the sign of the polynomial $F(X)$ evaluated over the algebraic number $\gamma_y$. This can be done similar to the evaluation scheme of `sign_at(Poly` $f$`, RootOf` $a$`)`, by computing the Sturm sequence of $P_x$ and $F(X)$ and evaluate this sequence on the rational endpoints of $I_x$. In order to decide the sign of each such evaluation we must test the sign of polynomials, of degree up to 4, evaluated over the algebraic number $\gamma_y$. This can be done easily by direct calls to

The main difference is that the quantities that we have to test, in order to decide the sign, involve $\gamma_x$. This can be done with direct calls to `sign_at(Poly` $f$`, RootOf` $a$`)` function.

As an example suppose

$$\begin{array}{l} \gamma_x \cong [P_x, I_x] \\ \gamma_y \cong [P_y, I_y] \\ P_x(X) = a_2\, X^2 + a_1\, X + a_0 \\ P_y(Y) = b_4\, Y^4 + b_3\, Y^3 + b_2\, Y^2 + b_1\, Y + b_0 \end{array} \tag{45}$$

The Sturm sequence of $P_x$ and $F$ is

$$\begin{array}{rcl} S_0(X) & = & P_1 \\ S_1(X) & = & F \\ S_2(X) & = & S_{21}\, X + S_{20} \\ S_3(X) & = & -r(S_{34}\, \gamma_y^4 + S_{33}\, \gamma_y^3 + S_{32}\, \gamma_y^2 + S_{31}\, \gamma_y + S_{30}) \end{array} \tag{46}$$

where

$$\begin{array}{rcl} S_{21} & = & r(a_2\, t\, \gamma_y + a_2\, u - r\, a_1) \\ S_{20} & = & r(a_2\, s\, \gamma_y^2 - r\, a_0 + a_2\, v\, \gamma_y + a_2\, w) \\ S_{34} & = & {a_2}^2\, s^2 \\ S_{33} & = & -a_1\, a_2\, t\, s + 2\, a_2^2\, s\, v \\ S_{32} & = & a_0 a_2 t^2 - a_1 a_2 t v - a_1 a_2 u s + r {a_1}^2 s + {a_2}^2 v^2 - 2\, a_2 s r a_0 + 2\, {a_2}^2 s w \\ S_{31} & = & -2\, a_2 r a_0 v - a_1 t r a_0 - a_1 a_2 t w - a_1 a_2 u v + r {a_1}^2 v + 2\, a_0 a_2 t u + 2\, {a_2}^2 v w \\ S_{30} & = & -2\, a_2 r a_0 w - a_1 a_2 u w + r {a_1}^2 w - a_1 u r a_0 + {a_2}^2 w^2 + r^2 {a_0}^2 + a_0 a_2 u^2 \end{array} \tag{47}$$

In order to find the sign of the evaluation of the Sturm sequence evaluated at the endpoints of $I_x$ we must find the sign of polynomials, of degree at most 4, evaluated at the algebraic number $\gamma_y$.

## 14 Preliminary Experiments

We have implemented 4 classes to handle algebraic numbers of respective degrees from 1 to 4. We encode an algebraic number as a root of a polynomial of degree up to four. So, we store the coefficients of the polynomial and an index that denotes which root we are interesting in. For algebraic numbers of degree up to 2 we have implemented addition, subtraction and multiplication. Additionally, we provide methods to compare any two algebraic numbers of degree up to 4.

We did some preliminaries tests, in order to estimate the efficiency of our method. These tests are by no means complete.

In order to find isolating intervals for quartics polynomials we use the *sqrt* function. In a new version of our code, we will not have this restriction since we plan to approximate to any level of accuracy this square root by continued fractions. We tested our algorithm against CORE ([?]. Our implementation uses SYNAPS [?] as a wrapper for the GMP-Float arithmetic type, while CORE uses its own type *BigInt*. The tests were performed on a 2.6 MHz Pentium with 512 MB memory, using g++ 3.2. One can see the results on Table 1, where the column STURM refers to our implementation. We mention that the *rootOf* operator in CORE cannot handle polynomials with multiple roots at present. On the other hand, our algorithm is faster on such degenerate polynomials, since in most of these cases we can find the roots exactly.

| | **Sturm** | **CORE** |
|---|---|---|
| *4-4* | 108 | 7889 |
| *4-3* | 105 | 8499 |
| *4-1* | 99 | 9476 |
| *2-2* | 107 | 8543 |

Table 1: Running times for various comparisons of the roots of two quartics. The left column indicates the indices of the roots of the two polynomials. The timings are in $\mu sec$.

## Acknowledgments

Both authors would like to thank Arno Eigenwilling for pointing out a error in solve function and for productive discussions about the multiplicities of the intersection points.

The second author acknowledges productive discussions and inspirational comments by Professor Nikos Tzanakis about the bounds of continued fractions and Fibbonacci numbers.

Both authors partially supported by INRIA's project "CALAMATA", a bilateral collaboration with National Kapodistrian University of Athens. The first author is partially supported by the IST Programme of the EU as a Shared-cost RTD (FET Open) Project under Contract No IST-2000-26473 (ECG - Effective Computational Geometry for Curves and Surfaces).

## 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, 2003.

[2] E. Berberich, A. Eigenwillig, M. Hemmer, S. Hert, K. Mehlhorn, and E. Schomer. A computational basis for conic arcs and boolean operations on conic polygons. In *European Symposium on Algorithms*, volume 2461 of *Lecture Notes of Computer Science*, pages 174–186. Springer-Verlag, 2002.

[3] P. Bikker and A. Y. Uteshev. On the Bézout construction of the resultant. *J. Symbolic Computation*, 28(1–2):45–88, July/Aug. 1999.

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

[5] J. E. Cremona. Classical invariants and 2-descent on elliptic curves. *J. Symbolic Computation*, 31(1/2):71–87, 2001.

[6] T. Decker and W. Krandick. Parallel real root isolation using the Descartes method. In P. Banerjee, V. Prasanna, and B. Sinha, editors, *Proc. High Performance Computing, Calcutta, India, 1999*, volume 1745 of *Lect. Notes Comp. Science*, pages 261–268. Springer, 1999.

[7] O. Deviller, A. Fronville, B. Mourrain, and M. Teillaud. Algebraic methods and arithmetic filtering for exact predicates on circle arcs. *Comp. Geom: Theory & Appl., Spec. Issue*, 22:119–142, 2002.

[8] G. Dos Reis, B. Mourrain, R. Rouillier, and P. Trébuchet. An environment for symbolic and numeric computation. In *Proc. of the International Conference on Mathematical Software 2002*, World Scientific, pages 239–249, 2002.

[9] 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*, pages 246–255. ACM, Jun 2003.

[10] I. Emiris, A. Kakargias, M. Teillaud, E. Tsigaridas, and S. Pion. Towards an open curved kernel. 2003. Submitted for publication.

[11] I. Z. Emiris and E. P. Tsigaridas. Comparison of fourth-degree algebraic numbers and applications to geometric predicates. Technical Report ECG-TR-302206-03, INRIA Sophia-Antipolis, 2003.

[12] I. Z. Emiris and E. P. Tsigaridas. Methods to compare real roots of polynomials of small degree. Technical Report ECG-TR-242200-01, INRIA Sophia-Antipolis, 2003.

[13] L. Guibas, M. Karavelas, and D. Russel. A computational framework for handling motion. In *Proc. 6th Workshop Algor. Engin. & Experim. (ALENEX)*, Jan. 2004. To appear: www.siam.org/meetings/alenex04/abstacts/lguibas-bin.pdf.

[14] M. Hemmer, E. Schömer, and N. Wolpert. Computing a 3-dimensional cell in an arrangement of quadrics: Exactly and actually! In *Proc. Annual ACM Symp. Comput. Geometry*, pages 264–273, 2001.

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

[16] M. Karavelas and I. Emiris. Root comparison techniques applied to the planar additively weighted Voronoi diagram. In *Proc. Symp. on Discrete Algorithms (SODA-03)*, pages 320–329, Jan. 2003.

[17] J. Keyser, T. Culver, D. Manocha, and S. .Krishnan. MAPC: A library for efficient and exact manipulation of algebraic points and curves. In *Proc. Annual ACM Symp. Comput. Geometry*, pages 360–369, New York, N.Y., June 1999. ACM Press.

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

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

[21] S. Pion and M. Teillaud. Towards a cgal-like kernel for curves. Technical Report ECG-TR-302206-01, MPI Saarbrücken, INRIA Sophia-Antipolis, 2003.

[22] R. Rioboo. Real algebraic closure of an ordered field: implementation in axiom. In *Proc. Annual ACM ISSAC*, pages 206–215. ACM Press, 1992.

[23] R. Rioboo. Towards faster real algebraic numbers. In T. Mora, editor, *Proc. Annual ACM ISSAC*, pages 221–228, New York, NY 10036, USA, 2002. ACM Press.

[24] F. Rouillier and P. Zimmermann. Efficient isolation of a polynomial real roots. Technical Report 4113, INRIA–Lorraine, 2001.

[25] G. Salmon. *Modern Higher Algebra*. G.E. Stechert and Co., New York, 1885. Reprinted 1924.

[26] S. Schmitt. Common subexpression search in leda_reals. Technical Report ECG-TR-243105-01, MPI Saarbrücken, 2003.

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

[28] B. Sturmfels. *Algorithms in Invariant Theory*. RISC Series on Symbolic Computation. Springer Verlag, Vienna, 1993.

[29] V. Weispfenning. Quantifier elimination for real algebra–the cubic case. In *Proc. Annual ACM ISSAC*, pages 258–263. ACM Press, 1994.

[30] L. Yang. Recent advances on determining the number of real roots of parametric polynomials. *J. Symbolic Computation*, 28:225–242, 1999.

[31] C. Yap. *Fundamental Problems of Algorithmic Algebra*. Oxford University Press, New York, 2000.