A numerical absolute primality test for bivariate polynomials

André Galligo*
Laboratoire de Mathématiques
Université de Nice (France)

Stephen Watt *
IBM T.J. Watson Research Center (USA) *and*
INRIA Sophia Antipolis (France)

Abstract

We give a new numerical absolute primality criterion for bivariate polynomials. This test is based on a simple property of the monomials appearing after a generic linear change of coordinates. Our method also provides a probabilistic algorithm for detecting absolute factors. We sketch an implementation and give timings comparing with two other algorithms implemented in Maple.

1 INTRODUCTION

Absolute factorization of polynomials with coefficients in a number field k means factorization over the algebraic closure of k. This is a problem which has attracted much attention in the domain of Computer Algebra, as we see from the work of Duval, Heintz and Sieveking, Kaltofen, Trager, Traverso and others [1] [4] [7] [8] [9] [16] [17] [18] [19].

We restrict ourselves to the case of characteristic zero and, for convenience, more precisely to the cases $k = Q$ or $k = Q(\alpha)$, where α is an algebraic number. We also restrict discussion to the bivariate case because it already contains most of the difficulties, and it is a main step toward the general case, see e.g. [21].

Various algorithms have been proposed to solve this problem. The most widely implemented appears to have been independently discovered by several authors [2] [8] [17] [18]. We will refer to this method as the "Single Extension Method."

Two methods are available in the Maple computer algebra system. The first, provided by the `AFactor` command, implements the Single Extension Method for factorization over an algebraic extension of **Q**. The second, included in the share library, has been done by Jean François Ragot [15] following the work of Dominique Duval; it uses a package, written by Mark Van Hoeij, for computing a basis of the ring of integers following Dedekind-Weber algorithm.

Most cited algorithms have a polynomial time complexity or are conjectured so. In contrast, the method presented in this paper needs an extensive search equivalent to the Knapsack problem, which is NP complete, so the main loop has an exponential complexity. Our method seems useful in practice because the calculations are very straightforward and most of them can be performed using floating point approximations.

*Supported in part by FRISCO, ESPRIT Reactive LTR 21.024.

Reprinted from pp. 297-224 *Proc. International Symposium on Symbolic and Algebraic Computation (ISSAC 97)*, 21–23 July 1997, Kihei Hawaii, USA.

The aim of this paper is principally practical. We wish to extend the range for which absolute irreducibility tests may be applied in real problems.

Jean François Ragot writes in his paper of December 1994 *"for bivariate polynomials of degree higher than 6, the two programs implemented in Maple are in general ineffective."* He provides, however, some special examples of higher degrees that either his program or the Single Extension Method can treat, and then gives the corresponding timings. These show that these two programs are well suited for some special cases. We will use these examples to test our method and to provide a first comparison.

In this paper we look for a heuristic which will work quickly on "most" examples of degree about 20. Our approach is probabilistic as it requires the choices of a generic linear change of coordinates and of a new origin. For the irreducibility test, we bound the maximal number of checks but the bound is huge and usually one check suffices. For the moment, until we further develop these ideas, it may be appropriate to consider our work as a preprocessing for the nice algorithms quoted above.

The starting point of our work is the observation that after a linear change of coordinates, all factors of the polynomial P become monic in y of maximal total degree. This implies degree conditions on the monomial terms.

As usual in the subject, we first reduce to the case of a square-free polynomial P. The zero characteristic assumption implies that the algebraic closure of k is contained in the field of complex numbers **C**, so we can use a geometric point of view on the plane curve $\mathcal{C}$ defined by P and try to deduce qualitative results from computations with usual double precision floating point numbers. This will succeed in "many" cases but it (of course) depends on the length of the coefficients of the given polynomial. We give a numerical analysis bounding the possible errors.

When the test of absolute primality fails, our method allows the computation to continue and to detect candidate absolute factors. We outline a probabilistic approach where the factorization process is divided in three steps. First, we aim to find the partition of the (simple) points of a generic fiber of a projection of $\mathcal{C}$ by the irreducible components.

This also determines the number and degrees of possible factors. Second, we construct polynomials P_i with $i = 1, \ldots, s$ with floating point coefficients such that the factorization obtained is valid to some precision. Third, we construct candidate polynomial factors with exact coefficient in an algebraic extension of $\mathbf{Q}$ and test the validity of the factorization. In this article, we only present timings for our implementation of the first step.

We emphasize that our approach is probabilistic more precisely of Las Vegas type: if it terminates (hopefully quickly) it is correct. We note that our process is easily parallelized, but we don't develop this point of view.

We illustrate our method with a very simple example which can be computed by hand:

$$\begin{aligned} P &= y^4 + 4xy^3 - 6y^3 + 7x^2y^2 - 16xy^2 + 11y^2 + 2x^3y \\ &\quad -16x^2y + 19xy - 6y - 5x^4 - 6x^3 + 5x^2 - 6x \end{aligned}$$

We choose $x_0 = 0$, then

$$\begin{aligned} P(0,y) &= y^4 - 6y^3 + 11y^2 - 6y \\ &= y(y-1)(y-2)(y-3). \end{aligned}$$

is square-free, therefore P is also square-free.

Taking $y_i = i$, we have the following limited expansions of the implicit functions:

$$y = \varphi_i(x) = y_i + a_i x + b_i x^2 + c_i x^3 + O(x^4)$$

$$\begin{aligned} \text{At } (0,0), \quad \varphi_0(x) &= 0 - x - \frac{x^2}{2} + \frac{x^3}{4} + O(x^4) \\ \text{At } (0,1), \quad \varphi_1(x) &= 1 - \frac{x}{2} - \frac{15}{8}x^2 + \frac{15}{8}x^3 + O(x^4) \\ \text{At } (0,2), \quad \varphi_2(x) &= 2 + \frac{x^2}{2} - \frac{x^3}{4} + O(x^4) \\ \text{At } (0,3), \quad \varphi_3(x) &= 3 - \frac{5}{2}x - \frac{15}{8}x^2 - \frac{15}{8}x^3 + O(x^4) \end{aligned}$$

This gives

$$\begin{gathered} a_0 = -1, b_0 = -\tfrac{1}{2}, c_0 = \tfrac{1}{4} \\ a_1 = -\tfrac{1}{2}, b_1 = \tfrac{15}{8}, c_1 = \tfrac{15}{8} \\ a_2 = 0, b_2 = \tfrac{1}{2}, c_2 = -\tfrac{1}{4} \\ a_3 = -\tfrac{5}{2}, b_3 = -\tfrac{15}{8}, c_3 = -\tfrac{15}{8} \end{gathered}$$

Then, we notice that

$$\begin{gathered} b_0 + b_2 = 0,\ c_0 + c_2 = 0 \\ b_1 + b_3 = 0,\ c_1 + c_3 = 0 \\ b_0 + b_1 = \tfrac{11}{8} \neq 0,\ c_0 + c_1 = \tfrac{17}{8} \neq 0 \\ b_2 + b_3 = -\tfrac{11}{8} \neq 0,\ c_2 + c_3 = -\tfrac{17}{8} \neq 0 \end{gathered}$$

Therefore, as we will explain in Sections 4 and 6, we associate 0 and 2, 1 and 3. The validity of this kind of association is discussed in Section 3. Next we form

- $(y - \varphi_0)(y - \varphi_2) = y^2 - 2y + xy - x^2 - 2x + O(x^4)$ and define $P_1 = y^2 + (x-2)y - (x^2 + 2x)$
- $(y - \varphi_1)(y - \varphi_3) = y^2 - 4y + 3xy + 5x^2 - 4x + 3$ and define $P_2 = y^2 + (3x-4)y + 5x^2 - 4x + 3$

We check that $P = P_1 \cdot P_2$ and obtain the absolute factorization.

The remainder of this article is structured as follows: Section 2 describes our basic observation: the presence of factors implies the vanishing of certain sums of coefficients, while Section 3 studies the sufficiency of this condition in the generic case. Section 4 gives a test of absolute primality based on computations with a limited precision; Section 5 lists some useful formulæ and presents an error analysis needed to validate the test. Section 6 outlines the process of detection and calculation of approximated factors. Section 7 presents our implementation and some ideas for (often) reducing the volume of computation, and Section 8 is devoted to the treatment of illustrative examples. Finally, Section 9 proposes a construction of candidate exact factors with coefficients in an algebraic extension of k.

2 PREPARATION

2.1 Square-free reduction

Given a polynomial $P \in k[x,y]$ with $k = Q$ or $k = Q(\alpha)$ where α is an algebraic number, by repeated computation of greater common divisors (gcd) we obtain a square-free decomposition of P. Therefore we can assume that this reduction has been done and that we only consider square-free polynomials. Thus P has no multiple factor over k nor, equivalently, over the integral closure of k. See [20] or [21]. For polynomials with approximate coefficients, gcds may be computed as in [3] or [10].

2.2 A basic lemma

A linear change of coordinates of the form $x := X + hY; y := Y$ transforms a polynomial $P(x,y)$ of total degree n into a new polynomial $Q(X,Y)$ also of total degree n, which can be written

$$Q(X,Y) = A^0(h)Y^n + A^1(h,X)Y^{n-1} + \ldots$$

where A^0 is a non-zero polynomial of degree less than or equal to n. So, if we choose any set of $n+1$ values for h, at least one of them is not a root of A^0. Therefore, after at most $n+1$ checks we find a linear change of coordinates s.t. after dividing by the non-zero leading term, the polynomial Q is monic in Y of degree n. We suppose that we have performed such a linear change of coordinates and write $P(x,y)$ instead of $Q(X,Y)$.

Lemma 1 *If a polynomial $P(x,y)$, monic in y and such that the degree in y (denoted by n) is the total degree, admits a factorization $P = P_1 \ldots P_s$, then each P_i for $i = 1, \ldots, s$ can be taken monic in y and such that its degree in y (denoted by n_i) is its total degree. Hence, it can be written:*

$$P_i = y^{n_i} + A_i^1 y^{n_i - 1} + A_i^2 y^{n_i - 2} + \ldots + A_i^{n_i}$$

where for each $j = 1, \ldots, n_i$, A_i^j is a polynomial in x of degree less than or equal to j.

Proof. Let for each i, n_i and m_i denote the total degree and the degree in y of P_i and $a_i y^{m_i}$ the corresponding term. So $m_i \leq n_i$. Multiplying the P_i we get P, then: $n_1 + \ldots + n_s = n = m_1 + \ldots + m_s$. Hence, for any i, we have $n_i = m_i$

and after dividing by the leading coefficient, we can suppose $a_i = 1$.

The bound on the degrees of the polynomials A_i^j follows immediately.

2.3 A non-vanishing result

As P is a square-free polynomial, for all values $x_0 \in Q$ except at most $n(n-1)$ of them, the roots $y_1, \ldots, y_n$ of $P(x_0, y)$ are distinct and all the corresponding points (x_0, y_j) of the curve $\mathcal{C}$, defined by P, are smooth.

We fix such a point (x_0, y_j) set $x = x_0 + X$ and let $y = \varphi_j(X)$ denote the implicit function defined by P near by (x_0, y_j). Hence, $P(x_0 + X, \varphi_j(X)) = 0$. We can easily compute a Taylor expansion of this function, as in [8]. We require an expansion of order 3 near by $X = 0$:

$$\varphi_j(X) = y_j + a_j X + b_j X^2 + c_j X^3 + O(X^4)$$

this defines the coefficients a_j, b_j, c_j which are rational function of y_j and therefore belong to the algebraic extension of $\mathbf{Q}$ containing y_j, their algebraic expressions are given in Section 5.

Hypothesis 1 *For now on, we assume that P is monic in y such that its degree in y is its total degree and P does not admit a factor of degree 1.*

Indeed detection and discard of degree 1 factors of P is an easier task than absolute factorization. There are several natural efficient ways by exact or approximate computations to perform this task. We will not develop this point in the present paper.

Proposition 1 *Let P be as in Hypothesis 1, then for almost all (rational) values of x_0, the coefficients b_j, c_j with $1 \leq j \leq n$ are non–zero.*

Proof. The function $\varphi_j(X)$ is analytic on $\mathbf{C}^2 - \Delta$ (Δ is the discriminant locus), as are its second and third derivatives $b_i(x)$ and $c_i(x)$. If the second derivative vanished on the rational numbers, it would be the zero function and $\varphi_j(X)$ would be a polynomial of degree less or equal to one and a factor of P. But this possibility has been excluded, therefore for almost all x_0 we have $b_j \neq 0$. Similarly, the vanishing of the third derivative on the rational numbers will imply that $\varphi_j(X) = y_j + a_j X + b_j X^2$ is a factor of P. But this possibility has been excluded (when $b_j \neq 0$) by our previous lemma.

3 GENERICITY

In this section, we analyze the generic behavior of the sums $B_I = b_{i_1} + \ldots + b_{i_m}$ for $I = \{i_1, \ldots, i_m\}$, respectively C_I defined similarly.

3.1 The irreducible case

First, we suppose that the polynomial P satisfies Hypothesis 1 and is irreducible in $C[x, y]$.

The following result is well known (see e.g. [13]).

Theorem 1 *The plane curve $\mathcal{C}$ defined by P in $\mathbf{C}^2$ is a ramified covering of degree n of the x-axis $\mathbf{C}$. Let $\Delta = \{z_1, \ldots, z_N\}$ be the set of abscissas of the ramification points supposed distinct. The first homotopy group $G = \Pi_1(\mathbf{C} - \Delta)$ acts transitively on any smooth fiber of that covering.*

This group action is also called monodromy. The result means that any two roots y_1 and y_2 of $P(x_0, y) = 0$ can be exchanged by continuous deformation obtained by letting x follow a (analytical) loop in the complex plane.

For example, let $P = y^2 - x$, $x_0 = 1$, then $y_1 = 1, y_2 = -1$. We deform the equation by letting $x(t) = e^{2i\pi t}$ i.e. we oblige x to follow the circle $\gamma : t \to e^{2i\pi t}$. Above the circle the two roots of P become $y_1(t) = e^{i\pi t}$ and $y_2(t) = -y_1(t)$. After a round the two roots are exchanged.

We will need a more precise result, less well known, due to J. Harris (cf [5]). We express this for plane curves:

Theorem 2 *Let $\mathcal{C}$ be a projective irreducible plane curve of degree n and $\mathcal{C}^*$ denote its incidence curve in the dual space. Then the first homotopy group $\Pi_1(\mathbf{P^2}(\mathbf{C}) * - \mathcal{C}^*)$ acts as the full symmetric group on any smooth section of $\mathcal{C}$.*

We fix a direction d_0 for the y-axis. For a fixed x_0 outside the discriminant locus Δ, the coefficients b_j can be expressed as a rational function of (x_0, y_j) of degree $\leq 3n$, i.e. quotient of two polynomials of such degree.

Let us fix an integer $m < n$. For any subset $I = \{i_1, \ldots, i_m\}$ of cardinality $card(I) = m$ of $\{1, \ldots, n\}$, the sum $B_I = b_{i_1} + \ldots + b_{i_m}$ is a rational function of x_0 and all the y_j of total degree $\leq 3nm$.

The product $\mathcal{B}_m = \prod_{card(I)=m} B_I$ is a rational function of x_0 and of all the y_j of total degree $\leq 3nm$. Moreover it is a symmetric function of the y_j for $1 \leq j \leq n$. As $P(x_0, y)$ is monic in y of total degree n, any symmetric rational function of the y_j for $1 \leq j \leq n$ can be expressed as a rational function of the same degree in x_0 and d_0. Therefore $\mathcal{B}_m$ is a rational function of x_0 and d_0, and its degree is bounded by $3nm \begin{pmatrix} n \\ m \end{pmatrix}$. Considering all integers $m < n$ we have:

Theorem 3 *For any direction d_0, except at most $3n^2 2^n$ directions, and for any complex (or rational) value x_0, except at most $3n^2 2^n$ values, for any m such that $m < n$ the sums $B_I = b_{i_1} + \ldots + b_{i_m}$ never vanish.*

Proof. Following the preceding discussion, we just have to rule out the possibility that $\mathcal{B}_m$ is identically zero. In that case, in a neighborhood $\mathcal{V}$ of (d_0, x_0), for any $(d, x) \in \mathcal{V}$, we would have $B_I = 0$. Therefore for any other subset of cardinal m of $1, \ldots, n$, applying Theorem 1 (to the corresponding projective situation) we deduce that there exists an (analytic) path in $\mathbf{P^2}(\mathbf{C})^* - \mathcal{C}^*$ exchanging $B_I(x)$ and $B_J(x)$ for any $(d, x) \in \mathcal{V}$. This would imply that for any j and any $(d, x) \in \mathcal{V}$, we have $b_j(x) = 0$ which is impossible since we assumed that P does not have a factor of degree 1.

Corollary 1 *of the proof: A similar result is true for C_I.*

3.2 The reducible case

Now, we suppose that P is a product of s prime absolute factors of degree n_i with $n = n_1 + \ldots + n_s$. Hence the curve $\mathcal{C}$ is the union of s irreducible curves $\mathcal{C}_i$ with $1 \leq i \leq s$. The incidence dual curve $\mathcal{C}^*$ of $\mathcal{C}$ contains the incidence dual curves $\mathcal{C}_i^*$ of $\mathcal{C}_i$. Therefore, the first homotopy group $\Pi_1(\mathbf{P^2}(\mathbf{C})^* - \mathcal{C}^*)$ acts as the full symmetric group on any smooth section of each curve $\mathcal{C}_i$ for $1 \leq i \leq s$. We fix such an (d_0, x_0) and denote the section of $\mathcal{C}_i$ by $\{y_j \ ; \ N_{i-1} < j \leq N_i\}$, where $N_i = n_1 + \ldots + n_i$.

Similarly to the construction in the previous subsection, we consider for each $1 \leq i \leq s$, a strict subset I_i of $\{N_{i-1}+1, \ldots, N_i\}$. Then we let

$$\mathcal{B}_{m_1,\ldots,m_s} = \prod_{card(I_i)=m_i} B_{I_1 \cup \ldots \cup I_s}$$

Similarly, we can prove that this is a rational function of (d_0, x_0) of degree bounded by the same bound as above, and that, if $B_{I_1 \cup \ldots \cup I_s}$ is identically zero, then for any list of subsets $J_i \in \{N_{i-1}+1, \ldots, N_i\}$ s.t. $card(I_i) = J_i$, we have that $B_{J_1 \cup \ldots \cup J_s}$ is also identically zero.

Similarly, we see that this implies that all b_j corresponding to the same irreducible curve are equal, and this can be ruled out by the same kind of argument as above. So we state:

Theorem 4 *For any direction d_0 except at most $3n^2 2^n$ directions and for any complex (or rational)value x_0 except at most $3n^2 2^n$ values, the sum $B_I = b_{i_1} + \ldots + b_{i_m}$, with $m < n$, vanishes only if it corresponds to the union of the roots of a family of factors of P.*

4 A TEST OF ABSOLUTE IRREDUCIBILITY

4.1 Notation

Let $P(x, y)$ be a polynomial of (total) degree n with coefficients in $k = \mathbf{Q}$ or $\mathbf{Q}(\alpha)$, α an algebraic number. Suppose that P has been prepared as in Section 2. We choose an adequate level of precision, which we will quantify this at the end of this section.

The general philosophy is that the result of a (precise) computation with limited precision can certify that some quantity is non–zero, but it can only hint that a quantity might be zero.

In order to certify, after a computation with limited precision, that an algebraic number is indeed zero, it requires prior qualitative knowledge. We will not consider this direction of investigation, however in the last section we will sketch a probabilistic method which aims to "guess" an exact factor and check divisibility.

We denote by $y_1, ..., y_n$ the simple solutions of $P(x_0, y) = 0$, computed approximately, for instance by a Newton process; we also suppose that $\frac{\partial P}{\partial y}(x_0, y_i)$ is clearly non-zero with respect to the level of approximation. We let $y = \varphi_i(x - x_0) = y_i + a_i(x-x_0) + b_i(x-x_0)^2 + c_i(x-x_0)^3 + O(x-x_0)^4$ be a limited expansion of the implicit solution of P at (x_0, y_i), computed approximately with the obtained value of y_i.

We compute the quantity $d_i = y_i c_i + a_i b_i$. We also suppose that for any $i = 1, \ldots, n$, a_i, b_i, c_i and d_i are clearly non-zero, or we try another value x_0.

Summarizing, we have a polynomial P, a value x_0 given exactly, $5n$ complex values given approximately but with enough precision (we will make this more precise in Section 5) y_i, a_i, b_i, c_i, d_i for $i = 1, \ldots, n$.

4.2 Test

If for all strict subsets $I = \{i_1, ..., i_\ell\}$ of $\{1, ..., n\}, \ell < n$, at least one of the three quantities:

$$B_I = b_{i_1} + ... + b_{i_\ell}, \quad C_I = c_{i_1} + ... + c_{i_\ell}, \quad D_I = d_{i_1} + ... + d_{i_\ell}$$

is clearly non-zero, then P is irreducible.

Remark 1 There are 2^n subsets so this test has an exponential complexity. For instance, for $n = 20$, $2^n \approx 10^6$ so it requires $\sim 10^7$ floating point operations.

Remark 2 Obviously we should have $B = \sum_{i=1}^n b_i = 0$, $C = \sum_{i=1}^n c_i = 0$ and $D = \sum_{i=1}^n d_i = 0$, but as we work with approximate data, they are almost zero. A simple probabilistic approach could be to interpret B_I, C_I, D_I "clearly non-zero" by greater than $100\,|B|$, $100\,|C|$, resp. $100\,|D|$, and require the three conditions.

Remark 3 A secure approach is obtained by a numerical analysis of the possible error in order to guarantee the non-vanishing of B_I, C_I, D_I. Note in that case that it is sufficient to guarantee the non-vanishing of one of them (see below).

4.3 Proof

If P is not irreducible, it has a monic factor P_1 of degree $\ell < n$. The solutions of $P(x_0, y)$ are denoted by $y_{i_1}, ..., y_{i_\ell}$ and near by x_0 we have:

$$\begin{aligned}
P_1(x_0+x, y) &= (y - \varphi_{i_1}(x)) \times ... \times (y - \varphi_{i_\ell}(x)) \\
&= y^\ell \\
&- y^{\ell-1} \sum_{j=1}^{\ell} \left((y_{i_j} + a_{i_j} x + b_{i_j} x^2 + c_{i_j} x^3 + O(x^4) \right) \\
&+ y^{\ell-2} \sum_{j \neq k} \left((y_{i_j} + a_{i_j} x + b_{i_j} x^2 + c_{i_j} x^3 + O(x^4) \right) \\
&\qquad \times \left((y_{i_k} + a_{i_k} x + b_{i_k} x^2 + c_{i_k} x^3 + O(x^4) \right) \\
&- \ ...
\end{aligned}$$

As P_1 is monic in y of degree ℓ, the coefficient of $y^{\ell-1}$ is of degree ≤ 1 and the coefficient of $y^{\ell-2}$ is of degree ≤ 2. This implies

$$\sum_{j=1}^{\ell} b_{i_j} = \sum_{j=1}^{\ell} c_{i_j} = 0 \tag{1}$$

and

$$\sum_{j \neq k} y_{i_j} \cdot c_{i_k} + a_{i_j} \cdot b_{i_k} = 0 \tag{2}$$

but this last equality gives

$$\sum_{j=1}^{\ell} \sum_{k=1}^{\ell} (y_{i_j} \cdot c_{i_k} + a_{i_j} \cdot b_{i_k}) - \sum_{j=1}^{\ell} (y_{i_j} \cdot c_{i_j} + a_{i_j} \cdot b_{i_j}) = 0$$

or

$$\begin{aligned}
&\left(\sum_{j=1}^{\ell} y_{i_j} \right) \cdot \left(\sum_{k=1}^{\ell} c_{i_k} \right) + \left(\sum_{j=1}^{\ell} a_{i_j} \right) \cdot \left(\sum_{k=1}^{\ell} b_{i_k} \right) \quad - \\
&\left(\sum_{j=1}^{\ell} y_{i_j} \right) \cdot \left(\sum_{k=1}^{\ell} c_{i_j} \right) + \left(\sum_{j=1}^{\ell} a_{i_j} \right) \cdot \left(\sum_{k=1}^{\ell} b_{i_j} \right) \quad = \quad 0
\end{aligned}$$

Using (1), the equality (2) is equivalent to

$$\sum_{j=1}^{\ell} (y_{i_j} c_{i_j} + a_{i_j} b_{i_j}) = 0 \tag{3}$$

and we are done.

4.4 Error Analysis

- We supposed P square-free, hence $Q(x) = Res_y(P, \frac{\partial P}{\partial y})$ is not the zero polynomial. We make it monic. There exist $A(x,y)$ and $B(x,y)$, two polynomials, such that

$$Q(x) = A(x,y)P(x,y) + B(x,y)\frac{\partial P}{\partial y}(x,y)$$

  We suppose $|x_0| \leq 1$, $Q(x_0) \neq 0$. Therefore the norms of the roots y_i are bounded by some value $\kappa > 0$ and $B(x,y)$ is bounded on the poly-disc $D(0,1) \times D(0,\kappa)$. Hence $\left|\frac{\partial P}{\partial y}(x_0, y_i)\right| \geq \frac{1}{\eta}$ for a fixed value $\eta \geq 1$, which could be computed from the coefficients of P and x_0, or just estimated once y_i is approximately computed.

- In Section 5, we show that the values b_j, c_j, d_j can be computed with enough precision such that we can certify that all digits of their double precision floating point representation are correct so, *e.g.*, $\left|\frac{\Delta b_j}{b_j}\right| \leq 2^{-32}$

- Now we consider the error on $B = B_I = \sum_{j \in I} b_j$. We let $b = \sum_{j=1}^{n} |b_j|$ and suppose $|B| \geq \varepsilon > 0$, if $\left|\frac{\Delta B}{B}\right| \leq \frac{1}{2}$ then we can certify that $B \neq 0$.

  We have: $\left|\frac{\Delta B}{B}\right| \leq \sum \left|\frac{\Delta b_j}{b_j}\right| \times \left|\frac{b_j}{B}\right| \leq 2^{-32} \times \frac{b}{\varepsilon}$ and we are done if $\varepsilon \geq b \times 2^{-31}$.

  The same type of analysis works for c_j or d_j.

5 COMPUTATION OF THE TAYLOR COEFFICIENTS

5.1 Exact formulæ

Given x_0 and y_j we let

$$\alpha_j = \frac{\partial P}{\partial x}(x_0,y_j), \quad \beta_j = \frac{\partial P}{\partial y}(x_0,y_j)$$

$$\gamma_j = \frac{1}{2}\frac{\partial^2 P}{\partial x^2}(x_0,y_j), \quad \delta_j = \frac{1}{2}\frac{\partial^2 P}{\partial y^2}(x_0,y_j), \quad \varepsilon_j = \frac{\partial^2 P}{\partial x \partial y}(x_0,y_j)$$

$$\lambda_j = \frac{1}{6}\frac{\partial^3 P}{\partial x^3}(x_0,y_j), \quad \eta_j = \frac{1}{6}\frac{\partial^3 P}{\partial y^3}(x_0,y_j)$$

$$\mu_j = \frac{1}{2}\frac{\partial^3 P}{\partial x^2 \partial y}(x_0,y_j), \quad \nu_j = \frac{1}{2}\frac{\partial^3 P}{\partial x \partial y^2}(x_0,y_j).$$

For simplicity we set $X = x - x_0$, $Y = y - y_j$ and we forget the indices j. We have

$$\begin{aligned} 0 &= P(x_0 + X, y_j + Y) \\ &= 0 + \alpha X + \beta Y + \gamma X^2 + \delta Y^2 + \varepsilon XY + \lambda X^3 + \\ &\quad \mu X^2 Y + \nu X Y^2 + \eta Y^3 + O(\|X,Y\|^4). \end{aligned}$$

Replacing Y by $aX + bX^2 + cX^3 + O(X^4)$, we get:

$$\begin{aligned} 0 &= (\alpha + ab)X + (\beta b + \gamma + \delta a^2 + \varepsilon a)X^2 + \\ &\quad (\beta c + 2\delta ab + \varepsilon b + \lambda + \mu a + \nu a^2 + \eta a^3)X^3 + O(X^4) \end{aligned}$$

So,

$$\begin{aligned} a &= -\alpha/\beta \\ b &= -1/\beta(\gamma + \delta\frac{\alpha^2}{\beta^2} - \varepsilon\frac{\alpha}{\beta}) \\ c &= -2\delta\frac{\alpha}{\beta^3}(\gamma + \delta\frac{\alpha^2}{\beta^2} - \varepsilon\frac{\alpha}{\beta}) + \frac{\varepsilon}{\beta^2}(\gamma + \delta\frac{\alpha^2}{\beta^2} - \varepsilon\frac{\alpha}{\beta}) \\ &\quad -\lambda\frac{1}{\beta} + \mu\frac{\alpha}{\beta^2} - \nu\frac{\alpha^2}{\beta^3} + \eta\frac{\alpha^3}{\beta^4} \end{aligned}$$

We notice that $1/\beta$ and its powers should be precomputed to a sufficient precision. This explains our requirement that β is "clearly" non-zero.

Remark 4 The derivation, and therefore all these quantities, can be expressed as polynomials in y_j of degrees $< 5n$, and we recall that y_j is a solution of $P(x_0, y) = 0$. Thus we could use a computer algebra system to perform first the division by P and then evaluate the remainder at y_j. This may help to control the precision of the approximate computation.

5.2 Error Analysis

Let $M > 0$ be a bound for $|y_i|$ obtained as a function of the norms of the coefficients of P, and $|x_0| \leq 1$. By Newton's algorithm we can compute y_j and its n powers $y_j^2, ..., y_j^n$ with arbitrary precision, say 2^{-m}.

So the coefficients of the polynomial first, second and third derivatives of P are bounded by n, n^2M, and n^3M.

So the coefficients $\alpha_j, \beta_j, \gamma_j, \delta_j, \varepsilon_j, \lambda_j$ and η_j are known with a precision $\leq n^3M \cdot 2^{-m}$.

We know that $\beta_j \neq 0$.

Let M' be a bound for $\frac{|\alpha_j|}{|\beta_j|}, \frac{|\gamma_j|}{|\beta_j|}, ..., \frac{|\eta_j|}{|\beta_j|}, 1$. Then a_j can be computed with a precision $nMM'2^{-m}$, b_j with a precision $3n^2MM'^3 2^{-m}$, c_j with a precision $10n^3MM'^5 2^{-m}$. So we prove the following proposition.

Proposition 2 *If b_j, c_j, d_j are $\neq 0$ and if we compute y_j (and its powers) with enough precision, then b_j, c_j, d_j can be known with a "certified" double precision* i.e. *all digits are correct.*

6 DETECTION OF FACTORS BY APPROXIMATION

6.1 Finding a partition

In this section, we continue with the previous notation and assume that the irreducibility test has failed. With fixed small tolerance $\epsilon_b, \epsilon_c\epsilon_d$, we suppose that we have kept enough precision in our approximate computations in order:

first, to compute the family $\mathcal{F}$ of all sets I in $\{1, \ldots, n\}$ such that

$$|B_I| < \epsilon_b, |C_I| < \epsilon_c, |D_I| < \epsilon_d$$

second, to discard in $\mathcal{F}$ the sets I which contains a strict subset also belonging to $\mathcal{F}$. We denote by $\mathcal{P}$ this restricted family of sets.

Moreover, we expect that $\mathcal{P}$ is a partition of $\{1, \ldots, n\}$. If it is not so, we increase the precision and re-do the computations only for the sums that we found small, in order to check if some of them have to be discarded. If it is still not so, then the value x_0 was unlucky so we change x_0 for another random value and start from the beginning.

So we suppose that $\mathcal{P}$ is a partition with s elements:

$$\{1, \ldots, n\} = \cup_{p=1}^{s} I_p$$

We let n_p denote the cardinality of I_p, so $n = n_1 + \ldots + n_s$. We expect that each subset I_p will correspond to a factor of P.

6.2 Multiplication of Taylor expansions

For each $p, 1 \leq p \leq s$, and for each $i \in I_p$, we compute a Taylor expansion of $\varphi_i(x)$ of order n_p. To simplify the notation, we suppose that we have translated the origin of the x-axis at x_0.

Then we multiply all $y - \varphi_i(x)$, for $i \in I_p$ modulo x^{n_p+1}. So we get:

$$\prod_{i=1}^{n_p}(y - \varphi_i(x)) = P_p(x, y) \ \ modulo \ \ x^{n_p+1}.$$

The polynomial P_p is then written:

$$P_p(x, y) = y^{n_p} + \sum_{j=1}^{n_p} A_j^p(x) y^{n_p - j}.$$

We expect that, up to the chosen tolerance, we have $deg A_j^p(x) \leq j$ and that P_p divides P.

We will see in the last section that we can alternatively use Hensel lifting to compute the (exact or approximated) factors from the qualitative data given by the partition.

6.3 Error recovery

If the polynomials P_p obtained do not fulfill this condition, this may mean that some sums whose approximations are small are indeed non–zero, and thus there are fewer factors. So we re-do the previous step for any new partition $\mathcal{P}'$ obtained by taking the union of some sets which are elements of $\mathcal{P}$. In the worst case, this search has an exponential complexity.

Remark 5 The genericity result of Section 3 and the error analysis of Section 5, which provides insight on the required order of precision, aim to reduce significantly the probability of the need of such a search.

7 IMPLEMENTATION

Our method aims to factorize polynomials of degree n about 20 . The reconstruction relies on an extensive search for detecting vanishing sums among 2^n possibilities: if $n = 20$, $2^n \approx 10^6$, if $n = 30$, $2^n \approx 10^9$. In order to diminish the combinatorial explosion in "most" cases we propose the following pattern matching method.

7.1 Subdivision

We expose the method for $n = 20$, and $\mathbf{B} = \{b_1, ..., b_{20}\}$.

We divide $\mathbf{B}$ into two subsets $\mathbf{B}' = \{b_1, ..., b_{10}\}$, $\mathbf{B}'' = \{b_{11}, ..., b_{20}\}$ so $card\mathcal{P}(\mathbf{B}') = card\mathcal{P}(\mathbf{B}'') = 2^{10} \sim 10^3$. Each subset of $\mathbf{B}$ is the union of a subset I of $\mathbf{B}'$ and of a subset J of $\mathbf{B}''$, $I = (i_1, ..., i_\ell)$, $J = (j_1, ..., j_m)$, clearly:

$$B_{I \cup J} = (b_{i_1} + ... + b_{i_\ell}) + (b_{j_1} + ... + b_{j_m}) = 0$$

if and only if

$$B'_I = b_{i_1} + ... + b_{i_\ell} = -(b_{j_1} + ... + b_{j_m}) = -B''_J.$$

We compute and keep in a vector the 2^{10} (complex) sums B'_I and in another vector the 2^{10} sums $-B''_J$. We look for (approximate) matching in the complex plane.

To prepare the matching, we order the values of B'_I (resp. $\mathbf{B}''_J$) by the order of magnitude of their real part or/and of the norm of the complex number $|\mathbf{B}'_I|$.

Given a tolerance ε, we divide and order the vector or the 2^{10} sums by pieces where

$$\ell\frac{\varepsilon}{2} < \left|B'_I\right| \leq (\ell + 1)\frac{\varepsilon}{2}$$

idem for $\mathbf{B}''_J$. So we only have to compare adjacent pieces. Generically, the values of $|\mathbf{B}'_I|$ and $|\mathbf{B}''_J|$ are not concentrated, so the number of checks is a small multiple of 2^{10} instead of 2^{20}.

7.2 Approximation

Our treatment relies on good approximations of the n coefficients b_i or c_i. Those are obtained by plugging a good approximation of y_i in the formulæ of Section 5. Therefore it is crucial to compute precisely the n roots of y_i of $P(x_0, y)$. We use the Maple command *Digits:=25* to get the y_i with 25 digits, and so the b_i, and c_i are "certified" double precision floating point complex numbers. In this way, the sums B_I are still rather precise without requiring bigfloat computation.

Remark 6 The input polynomial P could be given in a sparse form or more generally as a (straight line) program. In that case, one can get by automatic differentiation a new (straight line) program which evaluates the derivatives. This is all that we need to compute a good approximation of the y_i by a Newton method, and the coefficients b_i and c_i as in Section 5.

8 EXAMPLES

8.1 Ragot's examples

In order to test a preliminary implementation in Maple of the first step of our method, we have used the following two examples of degree 15 from a list presented in [15]. This implementation uses only the coefficients b_j (the c_j and d_j are needed for higher degrees).

Example 1

$$\begin{aligned} P1(x, y) = \ & y^9 + 3x^5y^6 + 5x^4y^5 + 3x^{10}y^3 \\ & -3x^6y^3 + 5x^9y^2 + x^{15}. \end{aligned}$$

We do the computations with bigfloats of 25 digits, we will check that the bounds M and M' of Section 5.2 are not big so this precision will be enough for our purpose.

We perform a (generic) change of coordinates: $(x = X + .9 * y + .8, y = Y)$ in order to get a polynomial monic in y of total degree 15, we choose $x_0 = 0$.

The timings on a PC Pentium Pro 200 are as follows.

- phase 1: 1.1 second of CPU time to compute the 15 roots of $P(0, y)$, the derivatives of P at these roots of order 1 and 2, compute the b_i and put them in a table.
- phase 2: .48 second of CPU time to create the powerset $\mathcal{P}(7)$, and fill a table with all partial sums indexed by subsets of $\{1, \ldots, 7\}$ and of all the opposite of the sums indexed by subsets of $\{8, \ldots, 14\}$ (it is enough to consider only those elements because we know in advance that the sum over all indices should be zero).

- phase 3: .23 second of CPU time to order the previous table and .1 second to check equalities.

So the total amount is 1.9 second.

The result was that the only possible zero sum $B_I$is indexed by 5 roots $(3, 4, 11, 12, 13)$ which corresponds to a factor of degree 5; as we withdraw number 15, we don't obtain the "complementary" zero sum indexed by $(1, 2, 5, 6, 7, 8, 9, 10, 15)$.

Comparison: To obtain the complete factorization on a IBM RS6000, Ragot relates that it took 21 seconds of CPU time with his program and 885 seconds with `AFactor`, the Maple implementation using the Single Extenstion Method. Note, however, that these computations provide more information than the irreducibility test.

Example 2

$$\begin{aligned} P(x,y) \;:=\; & y^{12} - 12x^2y^9 + 21y^8 + 6y^8x^5 + 9x^4y^6 \\ & -48y^5x^7 - 168y^5x^2 + 147y^4 + 12y^4x^{10} \\ & +84y^4x^5 - 27y^3x^6 + 63y^2x^4 + 18y^2x^9 \\ & -588yx^2 - 336yx^7 - 48yx^{12} + 8x^{15} \\ & +84x^{10} + 294x^5 + 343 \end{aligned}$$

The computation takes almost the same time (less than 2 sec.) as above for the previous polynomial.

Comparison: To obtain the complete factorization on a IBM RS6000, Ragot relates that it took 8725 seconds of CPU time with his program and 347 seconds with `AFactor`.

8.2 Examples with random coefficients

Example 3 We tested in 1 second CPU time that the following polynomial of degree 15 is absolutely prime:

$$\begin{aligned} P_1(x,y) \;:=\; & 48x^2y + 12x^2y^2 + 11x^2 + 6x^3 - 11x^7y^2 \\ & -63x^3y^2 + 54x^2y^3 - 23x^{10}y^3 + 58x^4y^4 \\ & +18x^9y^2 - 42x^5y^5 - 80x^{11}y^2 + 10y^{10}x^4 \\ & +14x^4y + 47y^{15} + 18x^{12}y^2 + 36y^{13} \\ & -23x^3y^{12} - 68x^8y^3 - 41x^4y^7 + 72x^3y^{11} \\ & +22xy^{13} - 76x^2y^{13} - 54y^{10} - 18y^9 \\ & +79x^7y^3 - 49x^7y^4 + 9x^5 - 75x^6y - 46x^3y^5 \\ & -17x^4y^5 + 3x^2y^7 + 21y^5 \end{aligned}$$

Then we multiplied it by a polynomial of degree 10 (also absolutely irreducible)

$$\begin{aligned} P_2(x,y) \;:=\; & -40x^2y^2 + 55xy^2 + 36x^3y - 82x^6y^2 - 73x^9y \\ & -27x^5y^5 + 18x^7y^2 + 31x^2y^3 + 54x^7y^3 \\ & -16x^3y^5 - 96y^4 + 13y^3 + 40x^2 - 99y^2 - 20y^6 \\ & +44x^9 - 65y^{10} + 39x^6 + 66xy^5 - 34x^3y^3 \\ & +99x^6y^4 - 9x^3y^4 - 83x^3y^7 + 76x^4y^3 28x^8 \\ & -89y^8 + 71xy^8 + 38xy^9 - 40x^2y^6 - 12y^5x^2 \end{aligned}$$

Example 4 We computed the product $P = P_1 \times P_2$ (we don't expand it because it is too large) and we submitted P, whose degree is 25, to our program.

We obtained the (correct) answer after 706 seconds. With the notation described in the previous subsection, this time is distributed as follows:

phase 1: 9.3
phase 2: 645.3
phase 3: $45.3 + 5.8$

Most of the time is therefore spent for creating a table of less than 10,000 elements and performing few additions on each entry. The timings should improve if more appropriate data structures were available in Maple.

9 DETECTION OF EXACT FACTORS

In this section we analyze the problem of computing a factor of P with exact numbers in a suitable algebraic extension of $\mathbf{Q}$, helped by the qualitative information deduced from the approximate computations described in the previous sections. More precisely, this information is the degrees n_p of the potential factors P_p and the names of the roots y_i of $P(x_0, y) = 0$ which are the roots of $P_p(x_0, y)$. We can view each name as an alias of a small disc in the complex plane which contains y_i and no other y_j with $j \neq i$.

We might, alternatively, perform the entire search procedure with exact numbers in the splitting field of $P(x_0, y)$. Of course, it would be easily certified but so time consuming that no serious example could be effectively computed. A better strategy would be to mix approximate computations in order to quickly discard most of the subsets whose corresponding sum is clearly non–zero and exact computations in order to certify that the sums appearing sufficiently "small" are indeed equal to zero. But even that would not be effective because in general splitting fields are too big.

In order to perform calculations in not too large an extension of $\mathbf{Q}$, we do Hensel liftings instead of Taylor expansions.

As (x_0, y_1) is a smooth point of the complex curve defined by P, we know (see e.g. [8]) that either P is irreducible or it admits a factor on a subfield of $k(y_1)$. Let us call K this subfield which is the best "not too large extension of $\mathbf{Q}$" that we can expect. We write $P(x, y) = F_1(x, y)F_2(x, y)$ for this factorization and require $F_1(x_0, y_1) = 0$. Our target is to compute explicitly F_1.

By specialization $x = x_0$, the former factorization induces a factorization $P(x_0, y) = F_1(x_0, y)F_2(x_0, y)$. As $P(x_0, y)$ is square-free, $R_1(y) = F_1(x_0, y)$ and $R_2(y) = F_2(x_0, y)$ are relatively prime in $K[y]$, but not necessarily irreducible.

Letting $X = x - x_0$, by $X-$adic Hensel liftings in $K[[X]][y]$ of this last relatively prime factorization in $K[y]$, we can recover uniquely the factorization

$$P(x, y) = F_1(x, y)F_2(x, y)$$

after less than n steps involving only rational calculation in K.

Therefore the task is reduced to a univariate polynomial question:

Let $R(y) = P(x_0, y) \in k[y]$ be a square-free polynomial. Assume that its roots in $\mathbf{C}$ are known with enough precision to put them in disjoint discs and label them $y_1, \ldots, y_n$. Assume also that we know, by an oracle, that for a given strict subset I of the interval $\{1, \ldots, n\}$, the polynomial $R_1 = \prod_{i \in I}(y - y_i)$ belongs to $k(y_1)[y]$.

The problem is to find an expression of R_1 rational in $k(y_1)$ or if possibly in an intermediate extension field between k and $k(y_1)$.

An easy, but costly, solution is to use a univariate factorization subroutine for polynomials with coefficients in a fixed algebraic extension of $\mathbf{Q}$, e.g. $k(y_1)$.

10 CONCLUSION

In this paper, we presented a numerical test of absolute irreducibility for bivariate polynomials. Our new approach proceed from a basic observation and a generic non-vanishing theorem. We have supported the method with a theoretical study including a numerical analysis and an implementation in the computer algebra system Maple, and have applied it to several examples. The reported timings show that the first implementation of our method is already rather robust and fast on polynomials of degrees about 20. This was indeed our principal target when we started this work.

Our method can be used also to detect absolute factors. We have divided the factorization procedure into 3 steps. The two first two can be performed with floating point numbers, while the third requires computation with algebraic numbers on a (possibly large) extension of $\mathbf{Q}$.

As we have said, the running times on our examples show that, for the range of degrees considered, the first step is satisfactorily fast, but we still have to develop further and improve the second and third step. We indicate some ideas to reach this goal, and the entire solution is planned for a future article.

Also, we have not yet tried to improve the complexity bounds. This could be achieved by replacing our simple (but exponential) loop in the first step involving the first (or the two first) coefficients of the Taylor development of an implicit solution by the consideration of a higher order development. However, the numerical computation of these higher derivatives is generally ill conditioned; this will be less robust and might need much more precision. In a future article we plan to implement variations of our method and present compromises using $\leq n$ term expansions and/or small integer simultaneous relation finding algorithms as surveyed in [6] and used in [12].

Acknowledgements

The authors would like to thank Erich Kaltofen for bringing some references to our attention, and the anonymous referees for their constructive remarks.

References

[1] Bajaj, C., Canny, J., Garrity, T., and Warren, J. Factoring rational polynomials over the complexes. In *Proc. ISSAC'89* (1989), ACM Press, pp. 81–90.

[2] Chistov, A. L., and Grigoriev, D. Y. Subexponential time solving of systems of algebraic equations i. Technical Report E-9 83, Steklov Mathematical Institute, Leningrad, 1983.

[3] Corless, R., Gianni, P., Trager, B., and Watt, S. The singular value decomposition for approximate polynomial systems. In *Proc. ISSAC'89* (1989), ACM Press, pp. 195–207.

[4] Duval, D. Absolute factorization of polynomials: a geometric approach. *SIAM J. of Comp. 20* (1991), 1–21.

[5] Harris, J. A bound on the geometric genus of projective varieties. *Ann. Sc. Norm. Sup. Pisa Cl.Sci. IV Ser. 8* (1981), 35–68.

[6] Hastad, J., Just, B., Lagarias, J., and Schnorr, C. Polynomial time algorithms for finding integer relations among real numbers. *SIAM J. of Comp. 18* (1989), 859–881.

[7] Heintz, J., and Sieveking, M. Absolute primality is decidable in random polynomial time in the number of variables. No. 11 in LNCS. Springer Verlag, 1981, pp. 16–28.

[8] Kaltofen, E. Fast parallel absolute irreducibility testing. *J. Symbolic Computation 1* (1985), 57–67.

[9] Kaltofen, E. Effective Noether irreducibility forms and applications. *J. Comput. Syst. Sci. 50*, 2 (1995), 274–295.

[10] Lakshman Y. N., and Karmakar, N. Approximate polynomial greatest common divisors and nearest singular polynomials. In *Proc. ISSAC'96* (1996), ACM Press.

[11] Lenstra, A., Lenstra Jr., H., and Lovacz, L. Factoring polynomials with rational coefficients. *Math. Ann. 261* (1982), 515–534.

[12] Miller, V. S. Factoring polynomials via relation-finding. No. 601 in LNCS. Springer Verlag, 1992, pp. 115–121.

[13] Mumford, D. *Algebraic Geometry I—Complex projective varieties*. Springer Verlag, 1976.

[14] Noether, E. Ein algebraisches Kriterium für absolute Irreduzibilität. *Math. Ann. 85* (1922), 25–33.

[15] Ragot, J. F. A method of absolute factorization of polynomials in two variables over Q — improvement and implementation, Dec. 1994. Preprint, presented as a poster at AAECC'95.

[16] Sasaki, T., Susuki, M., Kolár, M., and Sasaki, M. Approximate factorization of multivariate polynomials and absolute irreducibility testing. *Japan J. Indus. Appl. Math. 8* (1991), 357–375.

[17] Trager, B. *Integeration of algebraic functions*. PhD thesis, MIT, 1984.

[18] Traverso, C. A study on algebraic algorithms: the normalization. In *Rend. C. Sem. Mat. Torino* (1986), pp. 111–130.

[19] von zur Gathen, J. Irreducibility of multivariate polynomials. *J. Comp. System Sci. 31* (1985), 225–264.

[20] Yun, D. Y. Y. On square-free decomposition algorithms. In *Proc. SYMSAC'76* (1976), ACM Press, pp. 26–35.

[21] Zippel, R. *Effective Polynomial Computation*. Kluwer Acad., 1993.