# A Polynomial-Time Reduction from Bivariate to Univariate Integral Polynomial Factorization[*]

by Erich Kaltofen[**]

Department of Computer and Inf. Sciences
University of Delaware
Newark, Delaware 19711

---

[*] This paper was written while the author was a visiting research associate at Kent State University supported by the Department of Energy under grant DE-AS02-ER7602075.

[**] Author's current address: University of Toronto, Dept. of Computer Science, Toronto, Ontario M5S1A7, Canada.

*Abstract*

An algorithm is presented which reduces the problem of finding the irreducible factors of a bivariate polynomial with integer coefficients in polynomial time in the total degree and the coefficient lengths to factoring a univariate integer polynomial. Together with A. Lenstra's, H. Lenstra's and L. Lovasz' polynomial-time factorization algorithm for univariate integer polynomials and the author's multivariate to bivariate reduction the new algorithm implies the following theorem. Factoring a polynomial with a fixed number of variables into irreducibles, except for the constant factors, can be accomplished in time polynomial in the total degree and the size of its coefficients. The new algorithm can be generalized to reducing multivariate factorization directly to univariate factorization and to factoring multivariate polynomials with coefficients in algebraic number fields and finite fields in polynomial time.

## *1. Introduction*

Both the classical Kronecker algorithm [Kronecker 1882] and the modern multivariate Hensel algorithm [Musser 75, Wang 78, Zippel 79] solve the problem of factoring multivariate polynomials with integer coefficients by reduction to factoring univariate polynomials and reconstructing the multivariate factors from the univariate ones. However, the running time of both methods suffers from the fact that, in rare cases, an exponential number of factor candidates obtained from the univariate factorization may have to be tested to determine the true factors (cf. [Kaltofen 82b, Sec. 3.2]). In [Kaltofen 82a] we have shown that factoring multivariate polynomials with integer coefficients in a fixed number of variables is Turing-reducible in time polynomial in the total degree and the coefficient size to factoring bivariate polynomials. This paper provides a polynomial-time Turing reduction from bivariate to univariate integer polynomial factorization. Our algorithm is a bivariate version of an algorithm due to H. Zassenhaus [Zassenhaus 81], which, instead of leading to an integer linear programming problem, as is the case for Zassenhaus' algorithm, leads to a system of linear equations for the coefficients of an irreducible bivariate factor.

Recently, A. Lenstra, H. Lenstra, and L. Lovasz have shown that factoring univariate polynomials is achievable in polynomial time [Lenstra et al. 82]. Their as well as our algorithms exclude the full factorization of a possible common integer content, which all coefficients of the polynomial to be factored might have. Therefore our result implies the following theorem. Factoring a polynomial with a fixed number of variables into irreducibles, except for the constant factors, can be accomplished in time polynomial in the total degree and the size of its coefficients.

We will present our new algorithm for the bivariate case only though it can be generalized to directly reducing multivariate to univariate factorization [Kaltofen 82b, Chap. 3] thus avoiding the algorithm of [Kaltofen 82a]. However, the results in [Kaltofen 82a] also imply a polynomial-time reduction for irreducibility testing, which our new algorithm

does not provide. We like to point out that by using ideas from [Lenstra et al. 82] we were able to reduce the order of the approximation for a root needed to obtain a correct minimal polynomial though this improvement seems to be only applicable in the bivariate case.

The question arises whether our algorithm is of practical importance. Unlike in the univariate case, in the multivariate Hensel algorithm the factors of the reduced univariate polynomial are almost always the true images of the multivariate factors, in which case no exponential running time occurs. This empirical observation can be explained by the Hilbert Irreducibility Theorem but there is no guarantee that one can always avoid bad reductions in polynomial time. In this connection we state an open problem in section 6.

In this paper, we adopt the notation from [Kaltofen 82a]. We also apply the initial transformations discussed there to our input polynomial. We shortly review these matters in section 2. In the next section, we present the algorithm. The correctness of the proposed algorithm is then shown in section 4. In section 5 we analyze its complexity, in particular we show that the size of all intermediately computed integers stays within polynomial bounds. Section 6 concludes this paper with an open problem and a short discussion how our new algorithm can also be applied to coefficient domains other than the integers, such as finite fields and algebraic number fields.

## *2. Notation and Initial Transformations*

By Z we denote the integers (or, when stated, any unique factorization domain) by Q the rationals (or the quotient field of Z); by C we denote the complex numbers. By 1/r Z we denote those elements in Q which, when multiplied by r, lie in Z. D[y,x] denotes the bivariate polynomials in y and x over D; $deg_x(f)$ and $deg_y(f)$ denote the highest degrees in $f \in D[y,x]$ of x and y, resp., by deg(f) we denote the total degree of f. The coefficient of the highest power of x in f is referred to as the leading coefficient of f in x and will be denoted by $ldcf_x(f)$. We call f monic in x if $ldcf_x(f)$ is the unity of D. We note that the highest degree of y in any factor of f is less than or equal to $deg_y(f)$. The

infinity norm of $f \in C[y,x]$, the maximum of the absolute values of the coefficients of f, will be denoted by $|f|$. The square norm of the coefficients of f will be denoted by $|f|_2$. By $\binom{n}{m}$ we denote the binomial coefficient $\frac{n!}{m!\,(n-m)!}$.

First we notice that all arithmetic operations on Z[y,x] including substitutions and greatest common divisor (GCD) computations take polynomial time in the degree and coefficient size of the input polynomials. As we have shown in [Kaltofen 82a, Sec. 3], factoring $\bar{f}(\bar{y},x) \in Z[\bar{y},x]$ can be reduced to factoring a polynomial $f(y,x) \in$ Z[y,x] which is monic in x and for which f(0,x) is squarefree. The process involves the GCD computation of all coefficients $\in Z[\bar{y}]$ of $\bar{f}(x)$, a squarefree decomposition [Wang and Trager 79], a transformation which makes a squarefree factor monic [Knuth 81, Sec. 4.6.2, Exercise 18], and finally a translation of $\bar{y} =$ y+b, $b \in Z$. The translation point b must not be a root of the discriminant $\in Z[\bar{y}]$ of the constructed squarefree and monic polynomial but its degree is easily bounded. We refer the reader to [Kaltofen 82a, Lemma 5] for the details.

### *3. Description of Main Algorithm*

[Input: $f(y,x) \in$ Z[y,x] monic in x such that f(0,x) is squarefree. Z can be any unique factorization domain and Q its quotient field. Output: $g(y,x) \in$ Z[y,x] irreducible which divides f(y,x). The algorithm then can be called again to factor the co-factor of g.]

(I) [Initializations:] $n \leftarrow deg_x(f)$; $d \leftarrow deg_y(f)$.

Rewrite $f(y,x) = \sum_{k\geq 0} f_k(x)\, y^k$. [Since f is monic $deg(f_k) < n$ for $k \geq 1$. Also $f(0,x) = f_0(x)$.]

(F) [Factorization of $f_0(x)$:] Compute an irreducible factor $t(x) \in$ Z[x] of $f_0(x)$; m = deg(t). [Let $\beta$ be a root of t. In the following we will perform computations in $Q(\beta)$ whose elements are polynomials in $Q[\beta]$ of degree m-1.]

(N) [Newton iteration, emulated as Hensel lifting: We construct

$$\alpha_k(y) = \sum_{j=0}^{k} a_j\ y^j,\ a_j\ \varepsilon\ Q(\beta)$$

for k=0,1,... such that

$$f(y, \alpha_k(y)) \equiv 0 \bmod y^{k+1}.]$$

[Set order of approximation:] $K \leftarrow \lceil d(2n-1)/m \rceil$.

$g_0(x) \leftarrow x - \beta$; $h_0(x) \leftarrow f_0(x)/g_0(x)$; $a_0 \leftarrow \beta$.

FOR $k \leftarrow 1,\ldots,K$ DO steps (N1) and (N2)

(N1) IF k=1 THEN $b_k(x) \leftarrow f_k(x)$ ELSE

$$b_k(x) \leftarrow f_k(x) - \sum_{s=1}^{k-1} a_s\ h_{k-s}(x).$$

(N2) [Solve $g_0(x)h_k(x) + h_0(x)a_k = b_k(x)$ with $deg(h_k) \le n-2$.]

$$a_k \leftarrow b_k(\beta)/f_0'(\beta);$$

$$h_k(x) \leftarrow \left( b_k(x) - a_k h_0(x) \right) / g_0(x).$$

(L) [Find minimal polynomial for $\alpha_K$:]

[Compute the powers of $\alpha_K(y)$:]

FOR $i \leftarrow 0,\ldots,n-1$ DO $\alpha_K^{(i)}(y) \leftarrow \alpha_K^i \bmod y^{K+1}$.

FOR $I \leftarrow m,\ldots,n-1$ DO $L \leftarrow \lceil d(n+I)/m \rceil$;

Try to solve the equation

$$\alpha_L^{(I)} + \sum_{i=0}^{I-1} u_i(y)\ \alpha_L^{(i)} \equiv 0 \bmod y^{L+1} \qquad (3.1)$$

for polynomials $u_i(y)\ \varepsilon$ Q[y], $deg(u_i) \le d$. Let $u_i = \sum_{0 \le k \le d} u_{ik} y^k$ and let

$$\alpha_L^{(i)} = \sum_{k=0}^{L} \left( \sum_{j=0}^{m-1} a_{kj}^{(i)} \beta^j \right) y^{k.}$$

Then (3.1) leads to the linear system

$$a_{kj}^{(I)} + \sum_{i=0}^{I-1} \sum_{s=0}^{d} a_{k-s,j}^{(i)}\ u_{is} = 0 \qquad (3.2)$$

for k=0,...,L, j=0,...,m-1 in the variables

$u_{is}$, i=0,...,I-1, s=0,...,d.

IF (3.2) has a solution (which, as we will prove, is then integral and unique) THEN

$$g(y,x) \leftarrow x^I + \sum_{i=0}^{I-1} u_i(y)\, x^i$$

and EXIT. [We will also show that then g(y,x) is an irreducible factor of f(y,x).]

[At this point, the above FOR loop has not produced a solution to (3.1). In this case f is irreducible.] $g \leftarrow f$. □

Notice that L, the order of the approximation needed, grows with I, the possible degree of the minimal polynomial. Hence we could improve our algorithm by increasing the order of the approximation within the loop on I in step L instead of computing the best approximation eventually needed a-priori in step (N). Also, a complete factorization of $f_0$ may exclude certain degrees for g. E.g., if $f_0$ factors into irreducibles of even degree, then g cannot be of odd degree. (Cf. [Knuth 81, Sec. 4.6.2, Exercise 16].)

### *4. Correctness Proof*

We first show that step (N) computes a root $\alpha_K(y)$ of f(y,x) modulo $y^{K+1}$. The numbers $a_k \in Q(\beta)$ and the polynomials $h_k(x) \in Q(\beta)[x]$ must satisfy

$$\left( x - \sum_{k=0}^{\infty} a_k y^k \right) \left( \sum_{k=0}^{\infty} h_k(x) y^k \right) = f(y,x)$$

which leads to

$$g_0(x) h_k(x) + a_k h_0(x) = b_k(x). \qquad (4.1)$$

Noting the fact that $h_0(\beta) = f_0'(\beta)$ we now only need to set $x=\beta$ in (4.1) to obtain the assignments of step (N2). If we choose the n-1 coefficients of $h_k(x)$ and $a_k$ as unknowns then (4.1) becomes a linear system whose coefficient matrix is the resultant of $g_0$ and $h_0$, which in our case happens to be equal to $f_0'(\beta)$.

We now prove that the first solution of (3.2) corresponds to an irreducible factor of f. First, we must prove a simple lemma.

*Lemma 4.1:* Let g(y,x) monic in x divide f(y,x) in Z[y,x] and assume that $g(0,\beta) = 0$ in $Q(\beta)$. Then $g(y,\alpha_k(y)) \equiv 0 \bmod y^{k+1}$ for $k \geq 1$ and $\alpha_k(y)$ as computed in step (N).

*Proof:* Let $f = g\bar{g}$ and assume that j is the first index such that $g(y,\alpha_j(y)) \not\equiv 0 \bmod y^{j+1}$. Because $g(y,\alpha_{j-1}(y)) \equiv 0 \bmod y^j$ we have $g(y,\alpha_j(\beta)) \equiv \gamma y^j \bmod y^{j+1}$ with $\gamma \in Q(\beta)$ and $\gamma \neq 0$. Since $g(0,\beta)=0$ and f(0,x) is squarefree it follows that $\bar{g}(0,\beta) \neq 0$. Therefore

$$g(y,\alpha_j(y))\bar{g}(y,\alpha_j(y)) \equiv \gamma\bar{g}(0,\beta)y^j \not\equiv 0 \bmod y^{j+1},$$

a contradiction to $\alpha_j(y)$ being the j-th order approximation of a root of f. □

*Theorem 4.1:* The first solution of (3.1) in step (L), as I increases, determines a proper factor g of f in Z[y,x]. This factor is also irreducible.

*Proof:* We show that g must divide f provided its coefficients satisfy (3.2). The irreducibility of g then follows immediately from the fact that the minimal polynomial for the root of f(y,x) corresponding to $\alpha_L$ also provides a solution to (3.1) and hence (3.2).

Let D(y,x) = GCD(f,g), $j = deg_x(D)$. We shall prove that the condition j < I = $deg_x(D)$ is impossible. Assume that this condition is satisfied, i.e. $0 \leq j < I$. By $S_j(y,x)$ we denote the j-th subresultant of f and g and write

$$S_j(y,x) = s_{j0}(y)x^j + \ldots + s_{jj}(y)$$

with $s_{jp}(y) \in Z[y]$, $0 \leq p \leq j$ (cf. [Brown and Traub 71, Sec. 5]). Since D divides $S_j$ it follows that $s_{j0}(y) \neq 0$. From lemma 4.1 and the fact that g is the polynomial of smallest degree solving (3.1) we conclude that $D(0,\beta) \neq 0$. This implies that $y^{L+1}$ divides $s_{j0}(y)$. However, we can show that even $y^{mL}$ divides $s_{j0}(y)$.

Let $f(y,x) = x^n + a_{n-1}(y)x^{n-1} + \ldots + a_0(y)$, $g(y,x) = x^I + b_{I-1}(y)x^{I-1} + \ldots + b_0(y)$ and assume that $a_{-i}, b_{-i} = 0$ for -i <

0. Then

$$s_{j0} = \begin{matrix} 1 & a_{n-1} & . & . & . & a_{2j-I+1} \\ & & & & & \\ & & & & & \\ & & & & & \\ 0 & 1 & a_{n-1} & \ldots & & a_j \\ & & & & & \\ 1 & b_{I-1} & . & . & . & b_{2j-n+1} \\ & & & & & \\ & & & & & \\ & & & & & \\ 0 & 1 & b_{I-1} & \cdots & & b_j \end{matrix} \begin{matrix} \\ \\ \\ I-j \\ \\ \\ \\ n-j \\ \\ \\ \end{matrix} \qquad (4.2)$$

We can triangulize the determinant in (4.2) by repeatedly subtracting that multiple of a row of a smaller column entry such that the larger column entry is replaced by the remainder. This process is very similar to the Euclidean algorithm. However, if we interpret the rows in (4.2) as

$$fx^{I-j-1},\ldots,f\ ,gx^{n-j-1},\ldots,g$$

the row operations described above lead to constructing two polynomial $\lambda_i$ and $\mu_i \in Z[y,x]$, i = j ,..., n+I-j-1 such that

$$\lambda_i f + \mu_i g = h_i,\ deg_x(h_i)=i. \qquad (4.3)$$

It follows that

$$s_{j0} = \prod_{i=j}^{n+I-j-1} \mathrm{ldcf}_x(h_i).$$

But we will show below that for $j \le i \le j+m-1$

$$y^L \ divides\ \mathrm{ldcf}_x(h_i). \qquad (4.4)$$

Therefore $y^{mL}$ divides $s_{j0}(y)$ and since $s_{j0} \neq 0$ we conclude that $deg_y(s_{j0}) \ge mL$. However, the degree of each entry in the determinant in (4.2) is bounded by $d = deg_y(f)$ and thus $deg_y(s_{j0}) \le (I+n-1)d$. This contradicts the fact that $mL \ge (I+n)d$.

We finally prove (4.4). By (4.3) it follows that

$h_i(y,\alpha_L(y)) \equiv 0 \bmod y^{L+1}$ and $h_i = D\ \bar{D}$. Since $D(0,\beta) \neq 0$ we conclude that $\bar{D}(y,\alpha_L(y)) \equiv 0 \bmod y^{L+1}$. If $y^L$ did not divide $\mathrm{ldcf}_x(\bar{D})$ then $\bar{D}(0,\beta) = 0$. But $deg(\bar{D}(0,x)) = i-j < m$ and the minimal polynomial of $\beta$ has degree m. □

*Remark:* In the above proof the argument that $y^{mL}$ divides $s_{j0}(y)$ is due to [Lenstra et al. 82]. The author originally

used the bound L = (n+I-1)d which also generalizes to the case in which there are more than two variables. The proof for the tighter bound depends on the fact that Q[y] is a Euclidean domain and seems not to extend to the full multivariate case.

*5. Complexity Analysis*

In order to show that the algorithm is of polynomial-time complexity in deg(f) log(|f|) we first count the number of arithmetic operations in Q and then show that if Q is the set of the rationals, all numerators and the common denominator of fractions computed during our algorithm are bounded in size polynomially in deg(f) log(|f|).

*Step (F):* It is the result by [Lenstra et al 82] that t(x) can be computed in $O(n^{12}+n^9 log(|f|_2)^3)$ binary steps.

*Step (N):* We count the number of additions, subtractions and multiplications in $Q(\beta)$ which we shall call ASM ops. Obviously, the multiplication is the most expensive operation and takes $O(m^2)$ arithmetic operations in Q. The complexity of the initialization step in (N) is clearly dominated by later steps. Step (N1) takes O(kn) ASM ops and this complexity dominates step (N2). Hence, step (N) takes $O(K^2 n)$ ASM ops, i.e.

$$O(m^2 K^2 n)$$

rational operations.

*Step (L):* It is easy to show that all $\alpha_K^{(i)}$ can be computed in $O(K^2 n)$ ASM ops, or $O(m^2 K^2 n)$ rational operations. To solve the linear system (3.2) in p = m(L+1) equations and q = I(d+1) unknowns by Gaussian elimination takes at most $O(pq^2)$ rational operations, hence step (L) takes

$$O(m^2 K^2 n) + O(mKd^2 n^3)$$

rational operations. Since mK ≈ dn the number of rational operations for both steps (N) and (L) is dominated by

$$O(d^3 n^4).$$

A more difficult problem is to bound the size of any occurring rational number. Our proof proceeds in various stages. First we bound $|\alpha_K(y)| \le B_1(f)$ and $|\alpha_K^{(i)}(y)| \le B_2(f)$ for $1 \le i \le n-1$ as polynomials in $Q(\beta)[y]$ with complex coefficients, $\beta$ being any root of h. Then we estimate the common integral denominator $\le B_3(f,m)$ of all occurring coefficients of elements of $Q[\beta]$ calculated throughout the algorithm. For a computed element

$$\gamma(\beta) = \sum_{0 \le i \le m-1} \frac{c_i}{M} \beta^i$$

with M, $c_i \in Z$ we obtain from $|\gamma(\beta)| \le B_2(f)$ and $|M| \le B_3(f,m)$ a bound $B_4(f,m)$ for all occurring numerators $c_i$, i.e. $|c_i| \le B_4(f,m)$. Finally, we consider the Gaussian elimination on (3.2) and give a bound $B_5(f,m)$ for the absolute values of the numerators and denominators which appear in the course of this process. The logarithms of all bounds $B_1(f)$, $B_2(f)$, $B_3(f,m)$, $B_4(f,m)$ and $B_5(f,m)$ will be polynomial in deg(f) log(|f|).

The most difficult of these bounds is $B_1(f)$. We first need to prove a lemma.

*Lemma 5.1:* Let $g_0(x)$ and $h_0(x) \in Z(\beta)[x]$ be as computed in step (N). Furthermore, let A denote the Sylvester matrix of $g_0(x)$ and $h_0(x)$ with entries in $Q(\beta)$.

a) Then $|\beta| \le 2|f_0|$ and $|h_0| \le 2^n |f_0|_2 \le (n+1)^{1/2} 2^n |f_0|$, where the later coefficient bound holds for any factor $\in C[x]$ of $f_0(x)$.

b) Let M be any n-1 by n-1 submatrix of A. Then its determinant is bounded by

$$|det(M)| \le T(f_0) = \left( n \; 2^n \; |f_0| \right)^{n-1}.$$

c) The resultant of $g_0$ and $h_0$, the determinant of A, is bounded by $1/S(f_0) \le |\det(A)| \le 2T(f_0)$ with

$$S(f_0) = (4|f_0|)^{(n-1)(n-2)/2}.$$

*Proof:* a) It is easy to show that if $|\beta| \ge 2|f_0|$ then $|f_0(\beta)| > 0$ which proves the bound on $\beta$. The bound on

$|h_0|$ is the Landau-Mignotte bound translated to maximum norms [Mignotte 74, Knuth 81, Sec. 4.6.2, Exercise 20].

b) By part a) we know a bound for the absolute value of each entry in A. Hadamard's determinant inequality [Knuth 81, Sec. 4.6.1, Exercise 15] then gives the bound for $|\det(M)|$.

c) Let $\beta_2$ ,..., $\beta_n$ be the conjugates of $\beta = \beta_1$, i.e. $h_0(x) = (x-\beta_2) \ldots (x-\beta_n)$. Then $\det(A) = \prod_{2\le i\le n}(\beta-\beta_i)$. The discriminant of $f_0(x)$, $discr(f_0(x)) = \prod_{i\neq j}(\beta_i-\beta_j)$ is an integer not equal 0, because $f_0(x)$ is squarefree [van der Waerden 53, pp. 87-89]. Also $|\beta_i| \le 2|f_0|$ by a) and hence $|\beta_i-\beta_j| \le 4|f_0(x)|$ for $1 \le i < j \le n$. Therefore

$$\begin{aligned} 1 &\le |discr(f_0(x))|^{1/2} \\ &= |det(A)| \prod_{2\le i<j\le n} |\beta_i-\beta_j| \\ &\le |det(A)| \ (4|f_0|)^{(n-1)(n-2)/2}. \ \square \end{aligned}$$

The following theorem is a consequence of what we call the main theorem on the coefficient growth for the Hensel lifting algorithm. We shall not prove this theorem here but refer the reader to Theorem 1 in [Kaltofen 82a] the proof of which can be easily modified to yield our statement. The proof for the complete multivariate version can be found in [Kaltofen 82b, Sec. 3.4].

*Theorem 5.1:* Let $g_0(x)$, $h_0(x)$ and $a_k$ be as computed in step (N). Let $S(f_0)$ and $T(f_0)$ be as defined in lemma 5.1 and let $N = max(n^2, n|f|)$. Furthermore, let

$$\frac{1}{res(g_0,h_0)} = \frac{1}{R}\, r(\beta) \ with \ R \in Z,\ r(\beta) \in Z[\beta].$$

Finally, let $d_k$ denote $\frac{1}{k}\binom{2k-2}{k-1}$ for $k \ge 1$, the k-th Catalan number. Then for all $k \ge 1$

$$R^{2k-1}\, a_k \in Z[\beta]$$

and, independently of which root $\beta$ of $f_0$ we choose,

$$|a_k| \le d_k (N(f)\, S(f_0)\, T(f_0))^{2k-1}. \ \square$$

Therefore we can set

$$B_1(f) = d_K(N(f)\ S(f_0)\ T(f_0))^{2K-1} < (2n\,|\,f\,|\,)^{2Kn^2},$$

assuming that n ≥ 4. It is easy to show by induction that

$$|\,\alpha_K^{(i)}(y)\,| \le (K+1)^{i-1}B_1(f)^i$$

for 2 ≤ i ≤ n-1 which implies that we can choose

$$B_2(f) = ((K+1)B_1(f))^{n-1}.$$

We now demonstrate that for R = $res(t, f_0')$, t the minimal polynomial of β as computed in step (F),

$$\alpha_K^{(i)}\ \varepsilon\ \left(\frac{1}{R^{2K-1}}\ Z[\beta]\right)[y] \qquad (5.1)$$

for i = 1 ,..., n-1. We first show that $res(g_0, h_0)$ ε $1/R\ Z[\beta]$. There exist polynomials λ(x) and μ(x) ε Z[x] such that $\lambda t$ + $\mu f_0'$ = R. Thus $R/f_0'(\beta) = \mu(\beta)$ ε $Z[\beta]$. The remark in section 4 that $res(g_0, h_0) = f_0'(\beta)$ establishes our claim. Theorem 5.1 now implies that $a_k$ ε $1/R^{2k-1}Z[\beta]$ which shows our initial statement for i = 1, the remainder of which can be shown by induction. Using Hadamard's determinant inequality we can derive from (5.1) and lemma 5.1 a) a bound $B_3(f, m)$ for the common denominator in all rational coefficients, namely

$$R^{2K-1} < B_3(f, m) = \left((n+1)2^n\,|\,f_0\,|\right)^{2(n+m)K}.$$

A well-known lemma now gives us a bound for the numerators of all occurring rational coefficients.

*Lemma 5.2:* Let β be a root of $t(x)$ ε Z[x], monic squarefree of degree m. Let A be a real upper bound for the absolute value of any conjugate of β. Assume that

$$\left|\sum_{i=0}^{m-1} c_i \beta^i\right| \le C \ \textit{with}\ c_i\ \varepsilon\ Z.$$

Furthermore, let D be the absolute value of the discriminant of t. Then

$$|\,c_i\,| \le \frac{C\ m!A^{m(m-1)/2}}{D^{1/2}},\ 0 \le i < m.$$

(Cf. [Weinberger and Rothschild 76, Lemma 8.3].) □

In our case we can choose A = $2|f_0|$ by lemma 5.1 a), C = $B_2(f)\,B_3(f,m)$ and D ≥ 1. Therefore, a valid bound is

$$B_4(f,m) = B_2(f)\,B_3(f,m)\,m!\,(2|f_0|)^{m^2/2}.$$

We finally need to investigate the Gaussian elimination process. In order to solve (3.2) we can first remove the common denominator of all rational coefficients. As can be shown with little effort, all intermediate rationals computed during the elimination process are fractions of subdeterminants of the matrix for the linear system [Gantmacher 58, Chap. 2]. It is not necessary to calculate the GCD of the numerator and denominator of a newly obtained rational since, as can also be shown, the denominator of the row used for the elimination in subsequent rows divides the numerators and denominators in these rows after the elimination process. If q = (n-1)(d+1) then the Hadamard bound for numerators and denominators is

$$B_5(f,m) = (q^{1/2}\,B_4(f,m))^q.$$

Using the estimates from above and assuming that d ≥ n we can easily establish that

$$B_5(f,m) < (4n|f|)^{6n^4d^3}$$

which together with the initial operation count shows that the running time of steps (N) and (L) of our algorithm is dominated by

$$O(d^{6+\varepsilon}n^{8+\varepsilon}log(|f|)^{1+\varepsilon})$$

for any ε > 0. Since both the initial transformations and step (F) can be accomplished in polynomial time this concludes the proof that an irreducible factor of any bivariate integral polynomial can be found in time polynomial in its total degree and coefficient size. To find the remaining irreducible factors we reapply our algorithm. The coefficients of any intermediate cofactor can be bounded uniformly (cf. [Gel'fond 60, pp. 135-139] or [Knuth 81, Sec. 4.6.2, Exercise 21].) Hence the complete factorization process takes polynomial time.

## *6. Conclusion*

We have shown how to overcome the extraneous factor problem during the multivariate Hensel algorithm by approximating a root and then determining its minimal polynomial, which leads to solving a system of linear equations. Our main algorithm was formulated for coefficients from a unique factorization domain and hence can be also applied to polynomials over Galois fields or algebraic extensions of the rationals. It can be shown that in both cases the algorithm works in polynomial time.

In the case of algebraic coefficients we need a polynomial-time algorithm for univariate factorization. That this is possible is a consequence of the polynomial-time algorithm for factoring univariate polynomials over the integers [Landau 82]. One usually describes an algebraic extension of the rationals by the minimal polynomial of an algebraic integer generating the field and then reduces the problem to factoring polynomials with coefficients which are algebraic integers. The ring of algebraic integers is in general not a unique factorization domain. Therefore we cannot guarantee that a solution of (3.2) consists of algebraic integers but one can prove that the numbers are algebraic integers within an integral quotient [Weinberger and Rothschild 76, Lemma 7.1].

In the case that the coefficients are elements from a finite field one may not be able to carry out all transformations of section 2.3. It may happen that a good translation point b does not exist within the coefficient field. Then the coefficient domain has to be extended to a larger field and thus the factors returned by our main algorithm may have coefficients which are not in the original coefficient field. A simple trick by taking the norm [Weyl 40, pp. 10-13] can then be used to determine the irreducible factors in the smaller field. This approach together with the Berlekamp algorithm [Knuth 81, Sec. 4.6.2] gives an algorithm which works in time polynomial in the total degree of the input polynomial and

We conclude this paper with the following open problem. the cardinality of the coefficient field.

*Open Problem:* Does there exist an infinite sequence of irreducible polynomials $f(y,x) \in Z[y,x]$, $n = \deg(f)$, such that for no polynomial $p(n)$ any polynomial $f(i,x)$ is irreducible for an integer $i$ with $|i| < p(n)$? This problem asks whether there is an effective version of the Hilbert Irreducibility Theorem.

*References*

[Brown and Traub 71]

Brown, W.S., Traub, J.F.: On Euclid's Algorithm and the Theory of Subresultants. J. ACM **18,** 505-514 (1971).

[Gantmacher 59]

Gantmacher, F. R.: Matrix Theory, vol. 1. New York: Chelsea 1959.

[Gel'fond 60]

Gel'fond, A. O.: Transcendental and Algebraic Numbers. New York: Dover Publ. 1960.

[Kaltofen 82a]

Kaltofen, E.: A Polynomial Reduction from Multivariate to Bivariate Integer Polynomial Factorization. ACM Proc. Symp on Theory Comp. 1982, 261-266.

[Kaltofen 82b]

Kaltofen, E.: On the Complexity of Factoring Polynomials with Integer Coefficients. Ph.D. thesis, Rensselaer Polytechnic Institute August 1982.

[Knuth 81]

Knuth, D. E.: The Art of Computer Programming, vol.2, Seminumerical Algorithms, 2nd ed. Reading, MA: Addison Wesley 1981.

[Kronecker 1882]

Kronecker, L.: Grundzuege einer arithmetischen Theorie der algebraischen Groessen. J. f. d. reine u. angew. Math. *92,* 1-122 (1882).

[Landau 82]

Landau, S.: Factoring Polynomials over Algebraic Number Fields is in Polynomial Time. Private Communications.

[Lenstra et al. 82]

Lenstra, A. K., Lenstra, H. W., Lovasz, L.: Factoring Polynomials with Rational Coefficients. Report 82-05. Amsterdam: Mathematisch Instituut 1982.

[Mignotte 74]

Mignotte, M.: An Inequality about Factors of Polynomials. Math. Comp. *28,* 1153-1157 (1974).

[Musser 76]

Musser, D. R.: Multivariate Polynomial Factorization. J. ACM *22,* 291-308 (1976).

[van der Waerden 53]

van der Waerden, B. L.: Modern Algebra, vol.1. Engl. transl. by F. Blum. New York: Ungar Publ. 1953.

[Wang 78]

Wang, P. S.: An Improved Multivariate Polynomial Factoring Algorithm. Math. Comp. *32,* 1215-1231 (1978).

[Wang and Trager 79]

Wang, P. S., Trager, B. M.: New Algorithms for Polynomial Square-free Decomposition over the Integers. SIAM J. Comp. *8,* 300-305 (1979).

[Weinberger and Rothschild 76]

Weinberger, P. J., Rothschild, L. P.: Factoring Polynomials over Algebraic Number Fields. ACM

Trans. Math. Software *2,* 335-350 (1976).

[Weyl 40]

Weyl, H.: Algebraic Theory of Numbers. Princeton: Princeton Univ. Press 1940.

[Zassenhaus 81]

Zassenhaus, H.: Polynomial Time Factoring of Integral Polynomials. ACM SIGSAM Bulletin *15,* 6-7, (May 1981).

[Zippel 79]

Zippel, R. E.: Probabilistic Algorithms for Sparse Polynomials. Ph.D. thesis, MIT 1979.

*Appendix*

Errata and Remarks to [Kaltofen 82a]

p. 262, Lemma 5: d = max($deg_u(h)$, $deg_v(h)$, $deg_x(h)$).

p. 263, (4): ($\vec{h}_{ij}$, $\vec{g}_{ij}$) Syl($g_{00}$, $h_{00}$) = $\vec{b}_{ij}$.

p. 264, theorem 2, proof: Prof. H. Lenstra points out that the correspondence between the factors of f(u,v,x) and $g_{c,s}(u,x)$ also follows from the uniqueness of the Hensel lifting process (theorem 1), which avoids the introduction of the fractional power series domain.

p. 264, sec. col., line 8: $\bar{b}_{ijk}$.

p. 264, sec. col., middle: A more accurate bound for $\alpha$ is $|\alpha| \leq 2|F/lc(F)|$ and hence one can remove the factor 2sd from the bound in theorem 2.

p. 265, first col., line 3: monotonicly.

p. 265, theorem 3, line 9: deg(h) log(|h|).

p. 265, theorem 3, proof: Prof. H. Lenstra suggests the following much more efficient algorithm for the factorization of g. Choose a point c such that the g(u,cu,x) does not factor into more irreducibles than g(u,v,x). Then lift these factors to a factorization

of g(u,v,x) by performing the coefficient arithmetic in Q(u). Since the initial factorization is correct all computed coefficients must be elements of Z[u].

p. 266, Ref.: Adleman, L. M.