Factorization over Finitely Generated Fields

James H. Davenport*

Emmanuel College
Cambridge
England
CB2 3AP

Barry M. Trager

Mathematical Sciences Department
IBM Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights
NY 10598

Abstract. This paper considers the problem of factoring polynomials over a variety of domains. We first describe the current methods of factoring polynomials over the integers, and extend them to the integers mod p. We then consider the problem of factoring over algebraic domains. Having produced several negative results, showing that, if the domain is not properly specified, then the problem is insoluble, we then show that, for a properly specified finitely generated extension of the rationals or the integers mod p, the problem is soluble. We conclude by discussing the problems of factoring over algebraic closures.

1. Introduction.

The problem of factoring polynomials is one that has received great attention in computer algebra. Not only is factoring of great inherent interest and utility, but it is also required by a great many other algorithms, e.g. integration. In fact, when it comes to algebraic numbers, factorisation is a prerequisite for ensuring unique representations - the only way we can discover how to represent $6^{1/2}$ in terms of $2^{1/2}$ and $3^{1/2}$ is to observe that x^2-6 factors over the extension of the rationals generated by $2^{1/2}$ and $3^{1/2}$. While factoring over the integers, or over algebraic extensions, is the most common requirement, recently factoring over other domains has been required. Indeed the investigations that lead to this paper were prompted by the authors' interests in integration, which leads to requirements to factor over algebraic extensions of polynomial domains, over algebraic extensions of finite fields, and over algebraic closures of such objects.

2. Factoring over polynomial domains.

One of the major advances in the field was the development of effective methods for factoring univariate polynomials with integer coefficients (originally due to Zassenhaus[1969], and implemented, inter alia, by Musser[1971]). This process has subsequently been refined in a variety of ways (see, for example, Wang[1978] and [Moore & Norman,1981], and the recent ideas of Zassenhaus[1981]), but the fundamental principles are the same. We first outline the univariate method since the multivariate process relies on it.

1) The problem is reduced to factoring square-free polynomials. This is done by square-free decomposition, to which there are many references, e.g. [Yun,1977].
2) A 'suitable' prime p is chosen.
3) The coefficients of the polynomial to be factored are reduced mod p.
4) This polynomial is factored (over the integers mod p) by Berlekamp's[1967] algorithm.

---

* Many of the discussions leading to this paper took place while the first author was a post-doctoral fellow at the IBM Thomas J. Watson Research Center.

5) This factorisation is 'lifted' to one mod p^n, for suitably large n;
6) This is examined to yield the factorisstion over the integers. This is not necessarily a trivial process, since one factor over the integers may be represented by several factors mod p^n, and we need some way of combining factors mod p^n into factors over the integers. Collins[1979] discusses the average complexity of this process.

Since Berlekamp's method* will factorise polynomials over the integers mod p, this means that the problem of factoring univariate polynomials over the prime fields (viz. Q and the fields of integers modulo p) was solved.

3. Multivariate Polynomials.

The next problem to be considered is the factorisation of multivariate polynomials. The process (due originally, in the case of charaoteristio 0, to Wang and Rothschild[1975]) is in fact very similar to that for factoring univariate polynomials over the integers, and proceeds as follows:

1) Ensure the polynomial is square-free (doing square-free decompositions over the integers mod p is slightly tricky, since, for example $(x^p-1)' = 0$, but it can be done - see algorithm square-free-decompose in Davenport[1981] for one way).
2) Find values for all but one of the variables such that the result of substituting them in leaves one with a square-free polynomial. Since there are only finitely many values of any variable which do not change the square-free nature of a polynomial, this can always be done over the integers: the rare case when it cannot be done over the integers modulo p is discussed below.
3) Factor the resulting univariate polynomial by the methods discussed above.
4) 'Lift' this factorisation back to a multivariate one by methods analogous to those used in the univariate case. This can be done either variable by variable or by lifting all the variables at once [Wang, 1978], and Zippel[1979] has some interesting remarks on the interaction of this process with the sparsity of the original polynomial.

There is a potentially serious problem here, inasmuch as the factorisation in step (3) may yield many factors, several of which correspond to one multivariate factor. In the worst case, one may need to try all combinations of the results of extending the univariate factors before finding the multivariate factors (or asserting that the multivariate polynomial is irreducible). We shall not discuss this problem, often known as the "combinatorial explosion", further here, except to note that a variety of methods have been proposed, e.g. in [Wang,1978], to minimise the cost.

There remains the possibility that step (2) above cannot be completed over a finite field - for example no value of y leaves $x(x+1)(x+y)$ square-free over the integers modulo 2. The solution to this is to make the ground field larger, by taking an algebraic extension of it, and to admit values of y from this larger ground field. This leaves us with a univariate polynomial over an algebraic extension of the integers modulo p, which can be factored by the methods of Berlekamp[1970]. This can then be 'lifted' to a multivariate factorisation exactly as above, so that we now have a multivariate factorisation of the original polynomial over the larger ground field. This can be converted to one over the original ground field by considering the norm of each factor - more precisely, having lifted our factorisation from $k[\theta][x]$ to $k[\theta][x][y]$, we then consider this as a factorisation over $k[x][y][\theta]$, and take norms with respect to the extension by θ, as described by Trager[1976]. Of course, in practice this case is extremely rare, but nevertheless, as we have seen, it does not pose any theoretical embarrassment, though it is likely to cost a great deal in computer time, since our ground field is now an extension of the

---

* Though, if p is large, an alternative method [Berlekamp,1970] is better.

integers modulo p, rather than being just the integers modulo p.

Hence we can factor multivariate polynomials over any prime field.

Furthermore, since K[x,y] = K[x][y], we can factor over polynomial extensions of prime fields. As a result of Gauss's Lemma [van der Waerden, 1949, p. 73], this extends to rational function extensions, since, once we have cleared denominators, factoring in K[x][y] is the same as factoring in K(x)[y]. Hence we can factor polynomials over any finitely generated, purely transcendental extension of any prime field, and, since any polynomial can only involve a finite number of items, we can drop the restriction "finitely generated" in the above.

4. A Negative Result.

This therefore leaves algebraic fields as the next major problem. However, it is certainly not possible to solve the factorisation problem (i.e. produce an algorithm that will factorise any polynomial) for infinitely generated algebraic extensions of the integers, as is shown by the following example (due to Fröhlich & Shepherdson [1956]).

Let f be a function from the natural numbers into themselves whose image is a recursively* enumerable but not recursive set (such functions exist [Kleene, 1938]). Then let K be the field $Q(p_{f(1)}^{1/2}, p_{f(2)}^{1/2}, \ldots)$, where p_i is the i-th prime. Then consider attempting to factorise the polynomial x^2-p_n. This has one factor if p_n is not a square in K, i.e. n is none of the f(i), and two factors if n is one of the f(i). Hence any algorithm to factor polynomials, even of this very simple kind, over K would enable us to solve the question "is n one of the f(i)", which is known to be insoluble.

In case the above example be thought too abstract, here is a simple illustration of the fact that the ability to factorise polynomials (even quadratics) over infinitely presented fields is deeper than might be thought. Define g(n) to be 1 if 2n is the sum of two primes, and -1 otherwise. g is clearly a computable function. Let L be $Q(g(1)^{1/2}, g(2)^{1/2}, \ldots)$ and consider the factorisation of x^2+1 over L. It has one factor if L = Q, i.e. the Goldbach conjecture (that every even number is the some of two primes) is true, and two factors if the conjecture is false. This is more embarrassing, in some ways, because L is definitely finitely generated (being either Q or Q[i]), so a straight-forward restriction to being finitely-generated will not help here: we must insist, in some way, that the field be explicitly finitely generated.

While the above could be regarded as pedantry ("after all, who would actually state a problem like that"), it has an interesting consequence. Classical algebra texts (e.g. van der Waerden[1949]) show that, in the presence of a descending chain condition on divisors (which is nearly always present), the existence of greatest common divisors is equivalent to unique factorisation. Now it is certainly easy to construct greatest common divisors in the domains discussed above, while we cannot construct factorisations. This is a formalisation of the widely-held belief among computer algebraists that factorisation is "inherently" more complicated than g.c.d. computations.

5. Algebraic Extensions

So let us now assume that we have a field $L = k(t_1, \ldots, t_n, s_1, \ldots, s_m)$, which we shall also write as $K(s_1, \ldots, s_m)$, where the t_i are all transcendental over $k(t_1, \ldots, t_{i-1})$, and the s_i are algebraic, with given minimal polynomial, over $K(s_1, \ldots, s_{i-1})$. This choice of order, placing all the algebraics after all

---

*For the benefit of those not familiar with recursive function theory, this means that we can compute f(n) for any natural number n, but there can be no procedure for deciding if a given natural number m lies in the range of f, i.e. whether or not m=f(n) for some n.

the transcendentals, is not a real limitation, because the only constraint on the position of an algebraic is that it should come after everything in its minimal polynomial. Suppose that we wish to factor a (potentially multivariate) polynomial over L. If L is of characteristic 0 (i.e. if k is the rational numbers), there is no intrinsic problem - Trager's [1976] algorithm sqfr-norm can be used to reduce the problem to factoring a (generally much larger+) polynomial in the same variables over K, which we know how to do. Having factored this polynomial, we can recover the factors of the original polynomial over L, as in his algorithm alg-factor.

Life is not so simple if L has finite characteristic. There are two reasons for this:

a) (only applicable if n=0) There may not be enough elements of L, because sqfr-norm searches through L looking for a substitution which will produce a square-free norm, and it is possible (but only in a finite number of cases) for a substitution not to yield a square-free norm;
b) The whole theory of algebraic field extensions is much more complicated in the case of characteristic p, because an algebraic extension can now be inseparable (see van der Waerden[1949] Section 38). K[θ] is said to be an inseparable extension of K if the minimal polynomial of θ is irreducible over K, but has multiple roots in K[θ]. To see how this can happen, consider K=k[x], where k is the finite field with p elements, and x is transcendental over k, and let θ be defined by $\theta^p=x$. Then the minimal polynomial for θ, viz. y^p-x, is irreducible over K, but over K[θ] it factors into $(y-\theta)^p$.

These two problems require different solutions. Problem (a) is solved by observing that this can only occur if L is finite. This implies that n=0, and that L is an algebraic extension of a field with p elements. In that case we can reduce the problem to a univariate one, as described in section 3 (including, if necessary, the ground field extension process), and then factor the univariate polynomial thus produced by the method of Berlekamp[1970].

Problem (b) requires a rather different approach. We first observe (see van der Waerden[1949] for details) that an element can only be inseparable if its minimal polynomial is a polynomial in x^p. In fact we can split an inseparable extension into several parts - a separable extension followed by one or more purely inseparable extensions, viz. those generated by an element with minimum polynomial of the form y^p-z. It is a fact (often attributed to Krull[1953], but actually proved by Endler[1952]) that one can dispense with the inseparable extension by a change of generating elements. This is not hard to see in any special case*, and Endler provides an algorithm for performing the change of representation. Of course, once one has a separable representation, the problem polynomial can be transformed into that representation, and the factorisation problem solved there, and the resulting factors transformed back.

6. Algebraic Closures.

We have shown how one can factor polynomials over an explicitly finitely generated field, and this often what is required. However, one sometimes wants to factor over algebraic closures. A good example of this is integration, where the ground field has to be considered to be algebraically closed (see Risch[1969], where it is shown that $1/(x^2-2)$ is only integrable if the ground field contains $2^{1/2}$). The previous work is not of any direct help here, since algebraic closures are infinitely generated.

+ Even if the minimal polynomials for the s_i contain none of the t_i, the degree of the polynomial to be factored over the integers has been multiplied by the degrees of all the minimal polynomials.

* For example, if k is a field of p elements, and we consider k[x][y], where x is transcendental over k, and y satisfies $y^p=x$, then this field is isomorphic to k[y], under the mapping $y \to y$ & $x \to y^p$.

This problem was discussed by Risch[1969, p.178], who used the Kronecker trick of mapping $x_i \rightarrow t^{d^{i-1}}$, where d is larger than any integer occurring in the problem, to reduce the problem to a univariate one. Further details, and a cost analysis, are presented by Trager[1981, chapter 3].

So, let us suppose that we wish to factor a polynomial in $x_1, \ldots, x_n$ over the field K, where K is an explicitly finitely generated extension of a prime field*.

1) Ensure the polynomial is square-free, as in step (1) of Section 3.
2) Reduce the problem to a univariate one, as in step (2) of Section 3. Call this univariate polynomial f(x), defined in K[x].
3) Factor this over K[x], as in Sections 3 and 5. So $f(x) = f_1(x)f_2(x)\ldots f_m(x)$.
4) Until f is the product of linear factors, extend K by a root of one of the non-linear factors, and re-factor all the factors over this larger field.
5) Lift this back to a multivariate factorisation in $x_1, \ldots, x_n$, as in step (4) of Section (3). Note that the "combinatorial explosion" mentioned there is quite likely to occur in this case, because we have ensured that our original factorisation is into linear factors, all combinations of which will need to be tried before one can assert that the original polynomial is irreducible over the algebraic closure (if indeed it is).

---

* The reader may object that we need only consider K to be a transcendental extension of a prime field, since the algebraic closure of an algebraic extension of K is the same as the algebraic closure of K. This is perfectly true mathematically, but is a pitfall computationally, since algebraic numbers only have a unique representation after one has chosen a basis. As an example, consider factoring x^2+4 over Q[i]. It factors as (x+2i)(x-2i). But if we try to factor it over Q, we find that is does not factor over Q, so we introduce a new algebraic number θ defined by $\theta^2=-4$, and factor the polynomial as $(x+\theta)(x-\theta)$. If we try to use this as an extension of Q[i] we are in trouble, since i and θ are not independent.

7. Conclusions

We have shown that, assuming the problem is stated in a suitably explicit form, we can factor multivariate polynomials over any finitely generated extension of a prime field, or over the algebraic closure of such a field. In practice, this means that any 'reasonable' factorisation problem is solved in principle.

This is not to say that factorisation is a dead area. Many problems of implementation remain, and there is much that has to be done before all the algorithms described in this paper can be made available to the user.

Also, while these algorithms all work, and while parts of them are in use and proving relatively efficient, other parts are very expensive. In this context one thinks particularly of the "combinatorial explosion" of section 3, and Trager's[1976] algorithm sqfr-norm of Section 5. It is the authors' intuitive feeling that, in general, the exponential nature of sqfr-norm is inherent in the problem, but they have no proof. Indeed, there is remarkably little known about the complexity of factoring as a whole, though Yun[1977] deals with the square-free part of factorisation algorithms, and Collins[1979] studies the "average" complexity of the univariate factor-combining process.

Two particular areas that the authors feel deserve attention are:

1) The ideas of Zassenhaus[1981]. Can they be adapted to eliminate the combinatorial explosion of section 3.
2) The ideas of Wang[1976] and Weinberger and Rothschild[1976]. These deal with the factoring of multivariate polynomials over algebraic number fields (whereas, of course, our section 5 can deal with arbitrary extensions), but they are often more efficient when they apply. It is obviously possible to achieve some compromise between them and sqfr-norm - it would be interesting to investigate the details.

8. References

Berlekamp,1967
Berlekamp,E.R., Factoring Polynomials over Finite Fields. Bell System Tech. J. 46(1967) pp. 1853-1859.

Berlekamp,1970
Berlekamp,E.R., Factoring Polynomials over Large Finite Fields. Math. Comp. 24 (1970) pp. 713-735.

Collins,1979
Collins,G.E., Factoring univariate integral polynomials in polynomial average time. Proc. EUROSAM 79 [Springer Lecture Notes in Computer Science 72, Springer-Verlag, Berlin-Heidelberg-New York, 1979], pp. 317-329.

Davenport,1981
Davenport,J.H., On the Integration of Algebraic Functions. Springer Lecture Notes in Computer Science 102, Springer-Verlag, Berlin-Heidelberg-New York, 1981.

Endler,1952
Endler,O., Konstruktion von allgemeinen Normalkörpen in beschränkt vielen Schritten. Hausdorff-Gedächtnis-Preisarbeit der Math.-Nat.-Fakultät Bonn 1952.

Fröhlich & Shepherdson,1956
Fröhlich,A. & Shepherdson,J.C., Effective Procedures in Field Theory. Phil. Trans. Roy. Soc. Ser. A 248(1955-6) pp. 407-432.

Kleene,1938
Kleene,S.C., On Notation of Ordinal Numbers. Journal of Symbolic Logic 3(1938) pp.150-155.

Krull,1953
Krull,W., Über Polynomzerlegung mit endlich vielen Schritten. Math. Z. 59(1953) pp. 57-60.

Moore & Norman,1981
Moore,P.M.A. & Norman,A.C., Implementing a polynomial factorization and GCD package. Proc. SYMSAC 81.

Musser,1971
Musser,D.R., Algorithms for Polynomial Factorization. Ph.D. Thesis, Computer Science Department, University of Wisconsin (Madison) 1971.

Risch,1969
Risch,R.H., The Problem of Integration in Finite Terms. Trans. A.M.S. 139(1969) pp. 167-189 (MR 38 #5759).

Trager,1976
Trager,B.M., Algebraic Factoring and Rational Function Integration. Proc. SYMSAC 76, pp. 219-226.

Trager,1981
Trager,B.M., Integration of Algebraic Functions. Ph.D. Thesis, M.I.T. Dept. of EE&CS, expected date Sept. 1981.

van der Waerden,1949
van der Waerden,B.L., Modern Algebra, Frederick Ungar, New York, 1949 (Translated from Moderne Algebra 2nd. ed.).

Wang,1976
Wang,P.S., Factoring Multivariate Polynomials over Algebraic Number Fields. Math. Comp. 30(1976) pp. 324-336.

Wang,1978
Wang,P.S., An Improved Multivariable Polynomial Factorising Algorithm. Math. Comp. 32(1978) pp. 1215-1231.

Wang & Rothschild,1975
Wang,P.S. & Rothschild,L.P., Factoring Multi-Variate Polynomials over the Integers. Math. Comp. 29(1975) pp. 935-950.

Weinberger & Rothschild,1976
Weinberger,P.J. & Rothschild,L.P., Factoring Polynomials over Algebraic Number Fields. ACM Transactions on Mathematical Software 2(1976) pp. 335-350.

Yun,1977
Yun,D.Y.Y., On the Equivalence of Polynomial Gcd and Squarefree Factorization Algorithms. Proc. 1977 MACSYMA Users' Conference [NASA publication CP-2012, National Technical Information Service, Springfield, Virginia], pp. 65-70.

Zassenhaus,1969
Zassenhaus,H., On Hensel Factorization. I. J. Number Theory 1(1969) pp. 291-311. MR 39 #4120.

Zassenhaus,1981
Zassenhaus,H., Polynomial time factoring of integer polynomials. Presentation to Symbolic Mathematics Committee of SHARE Europe Association, Antwerp, 15th. Jan 1981. [An earlier version of this, entitled "On a problem of Collins", was presented at the AMS Summer Meeting in Ann Arbor, Aug. 1980.]

Zippel,1979
Zippel,R.E., Probabilistic Algorithms for Sparse Polynomials. Proc. EUROSAM 79 [Springer Lecture Notes in Computer Science 72, Springer-Verlag, Berlin-Heidelberg-New York, 1979], pp. 216-226.