Factoring Rational Polynomials over the Complexes

Chanderjit Bajaj * John Canny † Thomas Garrity ‡ Joe Warren §

Abstract

We give NC algorithms for determining the number and degrees of the absolute factors (factors irreducible over the complex numbers C) of a multivariate polynomial with rational coefficients. NC is the class of functions computable by logspace-uniform boolean circuits of polynomial size and polylogarithmic depth. The measures of size of the input polynomial are its degree d, coefficient length c, number of variables n, and for sparse polynomials, the number of non-zero coefficients s. For the general case, we give a random (Monte-Carlo) NC algorithm in these input measures. If n is fixed, or if the polynomial is dense, we give a deterministic NC algorithm. The algorithm also works in random NC for polynomials represented by straight-line programs, provided the polynomial can be evaluated at integer points in NC. Finally, we discuss a method for obtaining an approximation to the coefficients of each factor whose running time is polynomial in the size of the original (dense) polynomial. These methods rely on the fact that the connected components of a *complex* hypersurface $P(z_1, \ldots, z_n) = 0$ minus its singular points correspond to the absolute factors of P.

*Department of Computer Science, Purdue University. Supported in part by ARO Contract DAAG29-85-C0018 and ONR contract N00014-88-K-0402

†Computer Science Division, Berkeley. Supported in part by a David and Lucille Packard Fellowship

‡Department of Mathematics, Rice University

§Department of Computer Science, Rice University. Supported in part by NSF grant IRI 88-10747

1 Introduction

Factoring polynomials is an important problem in symbolic computation with applications as diverse as theorem proving and computer-aided design. Methods for factoring polynomials with rational coefficients over the rational numbers are well-known. [Lenstra 82,Kaltofen 85b] establish that factoring polynomials in a fixed number of indeterminates over the field of rational numbers Q is in polynomial time.

However, factoring polynomials over C differs from factoring over Q. For example, x^2+2y^2 is irreducible over Q. However, $x^2 + 2y^2 = (x + \sqrt{2}\mathrm{i}y)(x - \sqrt{2}\mathrm{i}y)$ when factored over C. This example illustrates one difficulty in factoring over C. The coefficients in an exact factorization over C must be represented symbolically (possibly by polynomials of high degree).

Work on factoring rational polynomials over C has not been as extensive as that of factorization over Q. [Noether 22,Davenport 81,Heintz 81] each give methods that require time exponential in the degree of the input polynomial. [DiCrescenzo 84,Duval 87] give geometric methods of factorization based on algebraic geometry. [Kaltofen 85a] describes an NC method for testing whether a rational polynomial is irreducible over C. The method involves computing approximate roots and their corresponding minimum polynomials. The first polynomial time algorithm for factoring over C seems to have been [Chistov 83]. However, it has remained an open problem whether computing the number of factors, irreducible over C, of a rational polynomial is in NC.

Given a polynomial P with rational coefficients, the input size is measured by number of variables n, degree d, coefficient size c, and number of non-zero coefficients s. We show that the general problem of computing number and degrees of the factors is in random NC in these measures, in the Monte-Carlo sense (definitely fast, probably correct). If the num-

ber of variables is fixed, or if the polynomial P is dense, we give a deterministic NC solution. Finally, if the polynomial is represented as a straight-line program of length p our algorithm runs in random NC plus the time to evaluate the polynomial at an integer point. By the parallelization result of Valiant et al. [Valiant 83], any straight-line program of size p and degree d can be converted into an equivalent program of polynomial size, and polylogarithmic depth in d and p, which can therefore be evaluated in NC. However, the conversion itself is not in NC, and seems intrinsically sequential, because of constant evaluation which is P-complete. So we cannot run our algorithm in random NC for straight-line program polynomials unless we are given a program of low depth.

Finally, we discuss a method for obtaining an approximation to the coefficients of each factor whose running time is polynomial if the polynomial P is dense. Previous methods of factoring have typically relied on an algebraic approach. We take a geometric approach, relying on the fact, described in section two, that the number of connected components of a *complex* hypersurface $P(z_1, \ldots, z_n) = 0$ minus its singular points is precisely the number of factors, irreducible over $\mathbf{C}$, of $P(z_1, \ldots, z_n)$. In section three, we describe a fast parallel method for reducing the factorization problem for $P(z_1, \ldots, z_n)$ to the bivariate case. In section four, we describe a fast parallel method for determining the number of connected components of $P(z_1, z_2)$ minus its singular points. This computation can be done using the sign sequences associated with various Sturm sequences.

2 Connectivity and Factorization

2.1 Preliminaries

Let $P_i(z_1, \ldots, z_n) \in \mathbf{C}[z_1, \ldots, z_n]$ $i = 1, \ldots, k$ be polynomials with complex coefficients in n variables. Let $V(P_1, \ldots, P_k)$ denote the set of common zeros of these polynomials in $\mathbf{C}^n$

$$V(P_1, \ldots, P_k) = \{z \in \mathbf{C}^n | P_i(z) = 0, \quad i = 1, \ldots, k\}$$

This is an example of an *algebraic* set. For a single polynomial P, the set $S = V(P)$ is called a *hypersurface*. A hypersurface S is said to be *irreducible* if it is the zeros set of a polynomial $P(z_1, \ldots, z_n)$ which is irreducible over $\mathbf{C}$. More generally, an algebraic set is irreducible if it cannot be expressed as a finite union of proper algebraic subsets. An irreducible algebraic set is called a *variety*.

For the rest of the paper, we assume that P is square-free (irreducible factors have multiplicity one). Note that if the original P is not square-free, we may compute the square-free part of P by computing

$$P/GCD(P, \frac{\partial P}{\partial z_1})$$

where P is monic in z_1. This computation may be performed in NC using greatest common divisor algorithm of [Borodin 82].

The key observation of this section is that there is a fundamental relationship between the *singular* points of a complex set and its irreducible components.

Definition Let P be a square-free polynomial, $S = V(P)$ a hypersurface, the set of *singular points* of S, denoted $Sing(S)$ is defined by

$$Sing(S) = S \cap V(\frac{\partial P}{\partial z_1}, \ldots, \frac{\partial P}{\partial z_n}). \qquad (1)$$

For example, an algebraic plane curve has a finite number of singular points. More generally, the singular set can be defined for any algebraic set, but we will not give a definition here. Intuitively, the singular points of an algebraic set are the points where the set is not smooth (smooth points have neighborhoods diffeomorphic to some $\mathbf{C}^k$).

2.2 Topology of Zero Sets of Reducible Polynomials

Removing the singular set from an algebraic set may split it into several connected components. Here connectivity means connectivity in the usual (metric) topology. As the following theorems show, these components correspond exactly to the irreducible components of the curve.

Theorem 1 *The set S is irreducible if and only if $S - Sing(S)$ is connected.*

A proof appears in [Griffiths 78, pp. 21].

Theorem 2 *The irreducible components of set S are exactly the closures of the connected components of $S - Sing(S)$.*

This is a consequence of the next two lemmas.

Lemma 1 *Let the set S have distinct irreducible components $S_1, S_2, \ldots, S_k$. Then for any i and j, $S_i \cap S_j \subseteq Sing(S)$.*

A proof also appears in [Griffiths 78, pp. 21].

Lemma 2 *If S is irreducible, and Y is any proper algebraic subset of S, then $S - Y$ is connected.*

This follows from Corollary (4.16) of [Mumford 1970].

3 Reduction to Bivariate Factorization

The previous theorems held for polynomials in any number of variables. However, we wish to focus our attention on the problem of factoring bivariate polynomials. This section describes a fast parallel method for reducing the problem of factoring a multivariate polynomial to the problem of factoring a bivariate polynomial.

There have been a number of papers giving reductions from multivariate to bivariate factorization. The first appeared in Heintz and Sievking [Heintz 81], and made use of Bertini's theorem. This was a randomized irreducibility test that worked for sparse multivariate polynomials. The idea was extended to factorization in [von zur Gathen 83]. In [Kaltofen 85b] a reduction was given which is in deterministic polynomial time if the number of variables is fixed, or if the polynomials are dense. [Kaltofen 85c] later gave a different randomized reduction for the sparse case. These randomized reductions work for polynomials represented as straight-line programs as well as sparse polynomials. An NC reduction for the dense case was given in [Kaltofen 85a].

For the complex case, we give a new randomized reduction which requires fewer bits per random coefficient $O(\log d)$ than the previous methods $O(d)$ for [Kaltofen 85c] and $O(d^2)$ for [von zur Gathen 83]. A consequence of this is that our reduction also runs in deterministic NC if the number of variables is fixed, or if the polynomials are dense. For sparse polynomials, the reduction is in random NC in the degree d, number of variables n, coefficient size c and number of non-zero terms s. For straight-line program polynomials, the parallel running time is the sum of a polylogarithmic function of measures d, n, c, plus the time to evaluate the polynomial at an integer point.

The irreducibility theorem is an adaption of a well-known result in algebraic geometry. It is stated as Corollary (4.18) in [Mumford 1970]:

Theorem 3 *Given an algebraic variety $X \subset \mathbf{P}^n$ (complex projective n-space) of dimension r, there is a linear subspace $L^{n-r+1} \subset \mathbf{P}^n$ such that $X \cap L$ is an irreducible curve, and X and L meet transversely.*

Since affine varieties have unique closure in projective space, the above theorem also applies to the affine case. The proof of corollary (4.18) in [Mumford 1970] gives a constructive method for finding the space L^{n-r+1}. In our case, $r = n - 1$, and the steps in finding the space L^2 are:

(a) Pick any linear projection $\pi_1 : \mathbf{C}^n \to \mathbf{C}^{n-1}$ such that π_1 restricted to X is almost everywhere a d to 1 covering. Since π_1 is determined by its kernel $v \in \mathbf{C}^n$, this is equivalent to choosing a vector $(0, v) \in \mathbf{P}^n$ not in the projective closure $\overline{X}$ of X.

(b) Let B be the set of branch points of π_1 restricted to X. Now choose any linear projection $\pi_2 : \mathbf{C}^{n-1} \to \mathbf{C}^{n-2}$. This is equivalent to choosing the kernel $u \in \mathbf{C}^{n-1}$ of π_2.

(c) Let B_0 be the set of branch points of π_2 restricted to B. Pick any point a in $\mathbf{C}^{n-2} - B_0$. Then the line $l = \pi_2^{-1}(a)$ is transversal to B, and let L^2 be the plane $\pi_1^{-1}(l)$.

Then by proposition (4.17) of [Mumford 1970], $L^2 \cap X$ is an irreducible curve, and L^2 and X meet transversely, hence the curve has degree d.

The space L^2 is determined by choosing v, u, and a. This is equivalent to picking three vectors b, v and u' in $\mathbf{C}^n$ with $a = \pi_2(\pi_1(b))$, $u = \pi_1(u')$, and letting L^2 be the plane $b + xv + yu'$. Since the map π_2 is arbitrary, we can assume without loss of generality that u' is $(1, 0, \ldots, 0)$, and then that $v_1 = a_1 = 0$. Determining bounds on the number of values of b and v for which this procedure fails gives us our reduction theorem:

Theorem 4 *Let $P(x_1, \ldots, x_n)$ be an irreducible polynomial of degree d. Let $b_2, \ldots, b_n$, $v_2, \ldots, v_n$ be elements chosen randomly from a finite set $E \subset \mathbf{C}$. Then the probability that the bivariate polynomial $Q(x, y) = P(y, b_2 + xv_2, \ldots, b_n + xv_n)$ is reducible is less than $d^4/|E|$, where $|E|$ is the cardinality of E.*

Proof We make use of Schwartz's lemma [Schwartz 80] that the number of points in the set E^n (E a finite subset of $\mathbf{C}$) that lie in an algebraic set $Z \subset \mathbf{C}^n$ of degree d is at most $d|E|^{n-1}$.

First of all, given $X = V(P)$ irreducible, the bad choices for $(0, v)$ are those that are contained in the projective closure $\overline{X}$ of X. This is a projective variety of degree d. By the Schwartz lemma, the probability of such a bad choice is $d/|E|$.

The $|E|^n$ possible values of b give us at least $|E|^{n-2}$ possible values for a (since at most $|E|^2$ lattice points can lie in $\ker(\pi_2 \circ \pi_1)$). These values of a must not lie in the set B_0. To find the degree of B_0, we note that B can be expressed as $\pi_1(V(P, \frac{\partial P}{\partial v}))$ where $\frac{\partial P}{\partial v}$ is the partial derivative of P in the direction of the vector v. By Bezout's theorem, B has degree at most $d(d-1)$. Similarly, the degree of B_0 is $\deg(B)(\deg(B)-1)$ which is less than $d^2(d-1)^2$.

The probability of one of the a's lying in this set is at most $(d^2(d-1)^2)/|E|$. The probability of a bad choice of either b or v is at most $(d+d^2(d-1)^2)/|E|$ which is less than $d^4/|E|$. □

Corollary 1 *Let $P(x_1,\ldots,x_n)$ be a polynomial of degree d with k factors. Let $b_2,\ldots,b_n$, $v_2,\ldots,v_n$ be chosen randomly from E. Then the probability that the polynomial $Q(x,y) = P(y, b_2+xv_2,\ldots,b_n+xv_n)$ does not have k factors with corresponding degrees is less than $d^4/|E|$.*

This follows because b and v can be chosen exactly as in Theorem (4).

So to achieve a probability of failure less than ϵ, we make sure $|E| > d^4/\epsilon$. Choosing integer values for elements of E therefore requires $(4\log d + \log\frac{1}{\epsilon})$ bits. For a deterministic algorithm, we take $|E| = d^4$. Then one of the $|E|^{2n-2}$ choices for b and v will work.

Once values for b and v have been chosen, we construct the polynomial $Q(x,y)$ by evaluating $P(y, b_2+xv_2,\ldots,b_n+xv_n)$ at integer values of x and y and interpolating.

4 Computing Connected Components

Having reduced multivariate factorization to bivariate factorization, we now focus on factoring the polynomial $P(z_1,z_2)$. As seen in the previous sections, if $S = V(P)$, this involves determining the connected components of $S - Sing(S)$. For now, we will describe the mathematical structure of our method for computing the connected components of $S-Sing(S)$. We shall delay the details of how to perform this construction in parallel until later in this section.

4.1 Topology of the Realification of S

Recall that any complex numbers z_1 and z_2 may be written as

$$\begin{aligned} z_1 &= x_1 + y_1\mathbf{i}, \\ z_2 &= x_2 + y_2\mathbf{i}, \end{aligned} \tag{2}$$

where x_1, x_2, y_1, and y_2 are real numbers. Thus the complex plane $\mathbf{C}^2$ can be interpreted as the real four-space $\mathbf{R}^4$. Any set S in $\mathbf{C}^2$ is then a set in $\mathbf{R}^4$, with $\dim_{\mathbf{R}}(S) = 2\dim_{\mathbf{C}}(S)$. In particular, S can be written as the intersection of two real hypersurfaces

$$S = V(P_1, P_2),$$

where $P_1(x_1,x_2,y_1,y_2)$ and $P_2(x_1,x_2,y_1,y_2)$ are the real and imaginary parts of P.

The complex curve S can be thought of as a surface in $\mathbf{R}^4$ which is the kernel of the map $(P_1,P_2) : \mathbf{R}^4 \to \mathbf{R}^2$. We would like to know where this surface is singular, and where the realified projection map fails to have a local inverse. First we need a couple of definitions.

Definition If $F : \mathbf{R}^n \to \mathbf{R}^m$ is a differentiable map, $p \in \mathbf{R}^n$ is a *critical point* of F if the Jacobian dF of F is not surjective at p. The image of a critical point is a *critical value.*

In complex algebraic geometry, the critical points are called *ramification points* and the critical values *branch points.* A point which is not a critical point is called a *regular point*, and the preimage of a *regular value* consists of regular points only.

If $Z = F^{-1}(0)$ for F as above, the set of regular points of F in Z form a manifold of dimension $n-m$. For this reason, they are called *smooth points* of Z. Singular points of Z will be critical points of F.

At singular points of $S = V(P_1,P_2)$, the Jacobian

$$\begin{bmatrix} \frac{\partial P_1}{\partial x_1} & \frac{\partial P_1}{\partial y_1} & \frac{\partial P_1}{\partial x_2} & \frac{\partial P_1}{\partial y_2} \\ \frac{\partial P_2}{\partial x_1} & \frac{\partial P_2}{\partial y_1} & \frac{\partial P_2}{\partial x_2} & \frac{\partial P_2}{\partial y_2} \end{bmatrix} \tag{3}$$

has rank one. For this to happen, both the determinants of

$$\begin{bmatrix} \frac{\partial P_1}{\partial x_1} & \frac{\partial P_1}{\partial y_1} \\ \frac{\partial P_2}{\partial x_1} & \frac{\partial P_2}{\partial y_1} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \frac{\partial P_1}{\partial x_2} & \frac{\partial P_1}{\partial y_2} \\ \frac{\partial P_2}{\partial x_2} & \frac{\partial P_2}{\partial y_2} \end{bmatrix} \tag{4}$$

must be zero. But the partial derivatives, since they arise from an analytic function, are not independent but must satisfy the Cauchy-Riemann equations. For the variable z_1 we have:

$$\frac{\partial P_1}{\partial x_1} = \frac{\partial P_2}{\partial y_1} \qquad \frac{\partial P_1}{\partial y_1} = -\frac{\partial P_2}{\partial x_1}, \tag{5}$$

so that the determinant of the first matrix of (4) is just $(\frac{\partial P_1}{\partial x_1})^2 + (\frac{\partial P_1}{\partial y_1})^2$ and will be zero only when both $\frac{\partial P_1}{\partial x_1}$ and $\frac{\partial P_1}{\partial y_1}$ (and therefore $\frac{\partial P_2}{\partial y_1}$ and $\frac{\partial P_2}{\partial x_1}$) are zero. But this is precisely the condition that the *complex* derivative $\frac{\partial P}{\partial z_1} = 0$.

Similarly, the second determinant in (4) vanishes only when all its elements vanish, which is true if and only if the complex derivative $\frac{\partial P}{\partial z_2} = 0$.

At singular points of S, both the minors in (4) must vanish, which as we have just seen, implies that all the elements in the matrices must vanish. This in turn implies that $\frac{\partial P}{\partial z_1}$ and $\frac{\partial P}{\partial z_2}$ are both zero, which is precisely the condition for a complex singular point. So the realified curve S is a smooth 2-dimensional surface, except at a finite number of singular points,

which are precisely the realification of the singular points of the complex curve S.

Finally, we would like to find out where the realified projection map $\pi : \mathbf{R}^4 \rightarrow \mathbf{R}^2$ taking $(x_1, y_1, x_2, y_2) \mapsto (x_2, y_2)$, fails to be locally invertible. By a change of variables, we can assume that $P(z_1, z_2)$ has a z_1^d term, implying that P does not have any factors univariate in z_2. Thus, π must be finite-to-one everywhere. π cannot be invertible at singular points of S. At a smooth point p, it will fail to be locally invertible if there is a tangent vector to S at p whose image under π is zero. Since the tangent space to S is the set of vectors orthogonal to the rows of (3), this condition is equivalent to the matrix

$$\begin{bmatrix} \frac{\partial P_1}{\partial x_1} & \frac{\partial P_1}{\partial y_1} & \frac{\partial P_1}{\partial x_2} & \frac{\partial P_1}{\partial y_2} \\ \frac{\partial P_2}{\partial x_1} & \frac{\partial P_2}{\partial y_1} & \frac{\partial P_2}{\partial x_2} & \frac{\partial P_2}{\partial y_2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

being singular. It is singular if and only if there is a vector orthogonal to all its rows, and such a vector is a tangent vector whose image under π is zero.

The matrix will be singular if and only if its upper left 2x2 submatrix is singular. By recourse to the Cauchy-Riemann equations, we see that this is equivalent to the complex condition $\frac{\partial P}{\partial z_1} = 0$. Again, this can occur at only a finite number of points, including the singular points of S.

4.2 Reduction to Curve Skeleton

Computing the connected components of $S - Sing(S)$ of a two-dimensional set is difficult. In this section, we will reduce this problem to that of computing the connected components of a one-dimensional subset of $S - Sing(S)$. Central to this reduction is the structure of S around its critical points. As shown in the previous section, the projection of the critical points of S onto the z_2 plane are the zeroes of the polynomial

$$R(z_2) = Res_{z_1}(P, \frac{\partial P}{\partial z_1}), \quad (7)$$

where $Res_{z_1}(P, Q)$ is the resultant of the polynomials of P and Q treated as one variable polynomials in z_1.

As an aid in generating our curve skeleton, we first generate a grid of lines G in the z_2 plane. The intersection of the inverse image of this grid and S will be a one-dimensional set. The edges of G will be parallel to the x_2 and y_2 axes. The vertical edges are the lines $x_2 = v_i$ with the $(v_0 < v_1 < \cdots)$ real constants. The horizontal edges are the lines $y_2 = h_i$ with the $(h_0 < h_1 < \cdots)$ real constants. Specifically, we choose the v_i's such that the open interval (v_i, v_{i+1}) contains

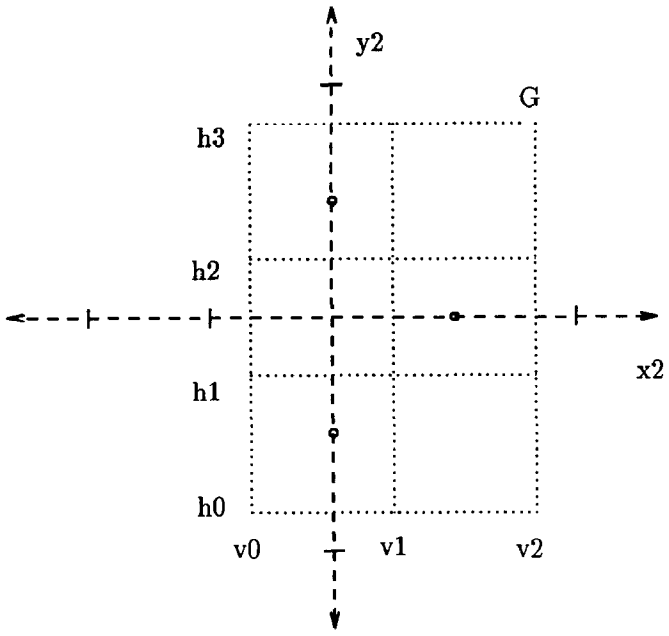

Figure 1: A grid plane whose cells contain at most one critical point.

at most one of the distinct real components of the complex zeroes of $R(z_2)$. Likewise, we choose the the h_i's such that the open interval (h_i, h_{i+1}) contains at most one of the distinct imaginary components of the complex zeroes of $R(z_2)$. Figure 1 illustrates this situation where $R(z_2) = (z_2 - 1)(z_2 + \mathbf{i})(z_2 - \mathbf{i})$.

The lines of G form rectangular cells in the z_2 plane, intersecting in vertices. The key property of this grid is that each cell in the grid contains at most one critical value.

G may now be used to construct a curve skeleton directly on S. The inverse image of G under the projection π onto the z_2 plane is a collection of three-dimensional hyperplanes. Let K be the set $S \cap \pi^{-1}(G)$, this is a curve skeleton that we can represent as a graph, $\hat{K}$. The vertices of $\hat{K}$ represent the points on S lying over each vertex (v_k, h_l) in G. These points are the complex roots of the univariate polynomial $P(z_1, v_k + \mathbf{i}h_l)$. The edges of $\hat{K}$ correspond to algebraic curve segments of K. Figure 2 illustrates three curves segments over two vertices s and t of the grid G, adjacent on a vertical grid line. The curve segments have been projected onto the $x_1 y_2$ plane.

The following two theorems state the relationship between K and the connected components of $S - Sing(S)$.

Theorem 5 *Each connected component of $S - Sing(S)$ contains at least one vertex of the curve skeleton $K = S \cap \pi^{-1}(G)$ over every vertex of G.*

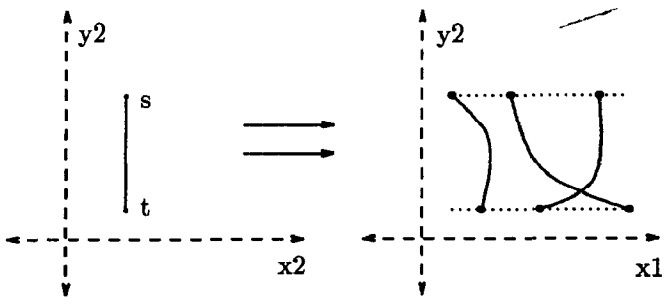

Figure 2: Curve segments on S joining vertices of K

Which follows from the fundamental theorem of algebra.

Theorem 6 *Any path in $S-Sing(S)$ connecting two points in K is homotopic (i.e. can be continuously deformed) to a path in K.*

This follows from the slightly stronger fact that K is a deformation retract of a subset of $S - Sing(S)$. Let B be the set of critical values of $\pi : S \rightarrow \mathbf{R}^2$, B is a finite set. Augment B to B' with a finite set of extra points so that every cell in the grid contains exactly one point of B'. Then K is a deformation retract of $S - \pi^{-1}(B')$.

We sketch the proof for a single grid cell. The cell, minus the point p of B' inside it, can be retracted onto its boundary. This can be done by creating a smooth field of unit vectors radiating from p. The flow defined by this field can be integrated, and defines the retraction. Since there are no critical values of π in this region, this vector field can be continuously lifted onto $S - \pi^{-1}(B')$. This gives us a deformation retraction of this cell of $S - \pi^{-1}(B')$ onto its boundary.

These two theorems assure us that the number of connected components of $S-Sing(S)$ equals the number of connected components in the curve skeleton K. Since K is one-dimensional, its topology may be realized as a graph. To determine the connectivity of K, we need only the adjacency information between points of K, not the actual curve segments. In the next section, we describe a fast parallel method for computing this adjacency information.

4.3 Construction of Curve Skeleton

As defined in the last section, over each vertex (v_k, h_l) in G, there are d vertices in $\hat{K}$. These vertices are the roots of the univariate polynomial $P(z_1, v_k + \mathbf{i}h_l)$. Unfortunately approximating the roots of a univariate polynomial in parallel is a major open problem. The adjacency information for K, though, does not rest on locating the zeroes but only on the relative position of one root to another. This information can be computed using the sign sequences associated with various Sturm sequences.

4.3.1 Sturm Sequences

Sturm sequences are classical. However, their importance in the symbolic manipulation of roots of polynomials is so great that we will review the key ideas. Let $p(x)$ be a one variable polynomial. Consider the following sequence $p_0(x), ..., p_n(x)$ of polynomials:

$$\begin{aligned} p_0 &= p \\ p_1 &= dp(x)/dx \\ &\vdots \quad \vdots \\ p_k &= q_{k-1}p_{k-1} - p_{k-2} \\ &\vdots \\ p_n & \end{aligned} \tag{8}$$

where p_k is simply the negative of the remainder obtained by dividing p_{k-2} by p_{k-1}. Since $p(x)$ is a polynomial, the last term p_n must be a constant. If $p(x)$ is square-free, p_n must be nonzero. Sturm sequences can be computed in NC [Borodin 82].

The importance of Sturm sequences lies in that they provide an easy way of determining how many real roots a polynomial has between two points.

Theorem 7 *Let $p(x)$ be a univariate real polynomial with Sturm sequence $(p_0(x), \ldots, p_n(x))$. Let a and b be real numbers that are not roots of $p(x)$. Then the number of real roots of $p(x)$ between a and b is equal to the number of sign changes in the sequence $(p_0(a), \ldots, p_n(a))$ minus the number of sign changes in the sequence $(p_0(b), \ldots, p_n(b))$.*

The proof can be found in many places, such as [Henrici 88, Chapter 6].

4.3.2 Computing Sign Sequences

Let Σ be a collection of rational polynomials $(p_1, ..., p_m)$ in n variables. Given a specified point $\mathbf{x}$ in $\mathbf{R}^n$, the *sign sequence* of the collection Σ is simply $(sign(p_1(\mathbf{x})), ..., sign(p_m(\mathbf{x}))$.

Theorem 8
Let $p_1(x_1, \ldots, x_n) = 0, \ \ldots, \ p_m(x_1, \ldots, x_n) = 0$ be a system of rational coefficient polynomial equations having a finite number of solution points. Denote the l real solution points not at infinity as $\alpha_j \in \mathbf{R}^n, j = 1, \ldots, l$. Let $q_1(x_1, \ldots, x_n), \ldots, q_k(x_1, \ldots, x_n)$ be a set of polynomials. Then the set of sign sequences of $q_1(\alpha_j), \ldots, q_k(\alpha_j), j = 1, \ldots, l$ can be computed in NC if m is fixed.

This theorem is a corollary of Lemma 2.4 in [Canny 88].

4.3.3 Parallel Adjacency Calculation

We now discuss how to compute the grid G and the adjacency information for $\hat{K}$ in NC with respect to the input size measures. Let $R(z_2)$ be defined by equation (7). Without loss of generality, we assume that $R(z_2)$ is squarefree (if not, make it so). We write $R(z_2)$ in terms of its real and imaginary parts:

$$R(x_2, y_2) = R_1(x_2, y_2) + \mathrm{i}R_2(x_2, y_2).$$

The complex zeroes of R are at the simultaneous zeroes of R_1 and R_2. Let

$$\begin{aligned} U(x_2) &= Res_{y_2}(R_1, R_2), \\ H(y_2) &= Res_{x_2}(R_1, R_2). \end{aligned} \tag{9}$$

The real zeroes of U contain the x_2-coordinates of the critical points and the zeroes of H contain the y_2-coordinates. Again, we ensure that U and H are squarefree.

The solutions v_i to the equation

$$\frac{dU(x_2)}{dx_2} = 0 \tag{10}$$

generate vertical lines that separate the critical points. Likewise the solutions h_i to the equation

$$\frac{dH(y_2)}{dy_2} = 0 \tag{11}$$

generate horizontal lines that separate the critical points. Finally, if A is a constant so that all roots of both U and H are greater than $-A$ and less than A, then the grid G consists of the lines from equations (10) and (11) and

$$x_2 = \pm A$$

$$y_2 = \pm A.$$

We now have a symbolic description of G. We next use this description with Sturm sequences to compute the adjacency information for $\hat{K}$. We describe a method for computing adjacency information in the x_2 direction in G. Let (v_i, h_j) and (v_i, h_{j+1}) be two adjacent vertices in G. These vertices lie on the grid line $x_2 = v_i$. Over these two vertices lie $2d$ points in S. These points form the vertices of $\hat{K}$. The intersection of $x_2 = v_i$ and S define d algebraic curve segments in K. These curve segments form the edges in $\hat{K}$, joining pairs of vertices in $\hat{K}$, each lying over a distinct grid vertex.

We do not attempt to explicitly construct and follow the curve segments. Instead, we symbolically compute the adjacency information. We will project $V(x_2 - v_i) \cap S$ onto the x_1y_2-plane via resultants. As shown in the next section, this projection will introduce only nodal singularities into the curve. To determine adjacency information, we need only locate and detect the relative position of these nodes with respect to the vertices of K. For example, see figure 3

Specifically, consider the three polynomials.

$$\begin{aligned} T(x_1, x_2, y_2) &= Res_{y_1}(P_1, P_2). \\ N(x_2, y_2) &= Res_{x_1}(T, \tfrac{\partial T}{\partial x_1}). \end{aligned} \tag{12}$$

$V(T)$ is the projection of S to x_1, x_2, y_2 space. The second polynomial restricts S to planes parallel to the x_1y_2 plane and through the vertical grid lines, forming an algebraic space curve. $V(N, dU/dx_2)$ consists of lines in the x_1y_2 plane, parallel to the x_1 axis, containing nodes of the projected plane curve (the dotted horizontal lines in figure 3).

Compute the sign sequences of the following polynomials:

- The Sturm sequence of dU/dx_2.
- The Sturm sequence of dH/dy_2.
- The Sturm sequence with respect to y_2 of $N(x_2, y_2)$.
- The Sturm sequence with respect to x_1 of $\frac{\partial T}{\partial x_1}$.

at the common zeros of the system (12). By theorem 8, these sign assignments can be computed in NC with respect to the size of the input polynomials .

To compute adjacencies for $\hat{K}$ we proceed as follows: As y_2 increases, the number of sign alternations of the Sturm sequence for dH/dy_2 increases monotonically. We first sort all the sign assignments according to the number of sign alternations in this Sturm sequence within each sign assignment. This partitions all the zeros of (12) into classes according to y_2 coordinate.

Each of these classes provides adjacency information for a particular slice $y_2 = h_i$. Next we sort within each class according to number of sign alternations of the Sturm sequence of dU/dx_2. This gives us a collection of classes which lie on the same horizontal grid segment between two adjacent vertical grid lines. Let this segment have endpoints s and t as in figure 3. Over s, there are four vertices s_1, s_2, s_3, and s_4 in K . Likewise over t there are four vertices . The projected curve segments link the s_i to the t_j.

Then sort within classes according to number of sign alternations of the Sturm sequence of $N(x_2, y_2)$. This partitions the sign assignments into classes having the same (x_2, y_2) coordinates, which are the coordinates of the node points (the dotted lines in figure 3).

Finally, we sort the sign assignments according to number of alternations of the Sturm sequence of $\partial T/\partial x_1$. This orders the points with the same z_2-coordinates along the dotted line by x_1 coordinate. One of these sign assignments will have a zero assignment to the polynomial $\partial T/\partial x_1$, and this is the sign assignment of the node point itself. From the position of this sign assignment in the ordering, we infer the relative position of the node point along the dotted line and therefore among the branches of the curve in K.

To generate the graph $\hat{K}$, we label the d vertices of $\hat{K}$ over a given grid point with $1, \ldots, d$. These labels come from the x_1 ordering of the corresponding points in K. Each node can be represented as a permutation (an exchange of two adjacent elements) of the indices of the curve branches that cross at the node. To determine the permutation as we move in y_2 past k nodes, we compose the permutations of the nodes. The composition can be done in NC by composing adjacent (in y_2 ordering) permutations, then composing adjacent pairs of these etc. The final permutation gives the change in ordering from one grid point to the next, and provides the d edges joining corresponding vertices of $\hat{K}$.

One may perform similar calculations to compute adjacency information in the horizontal direction.

4.3.4 Projections Introducing only Nodal Singularities

Note from the construction that the space curve $V(x_2 - v_i) \cap S$ has no singularities. However, in projecting this space curve to a plane, we may necessarily introduce singularities. We now show that we may deterministically project the space curve onto the x_1y_2 plane introducing only nodes as singularities. The proof of [Hartshorne 77, Theorem IV.3.10], over the reals, shows that for a generic projection, a space curve is mapped to a plane curve with only nodes for singularities. To deterministically choose a correct projection, we must first characterize those projections that introduce non-nodal singularities. This characterization will take the form of a polynomial condition on the points of the projection that yield such a projection.

By [Hartshorne 77, Theorem IV.3.7] (which, while stated only for algebraically closed fields, can be checked to still apply to the reals), such a point of projection must lie on a multisecant of the curve (i.e. a secant intersecting the curve in more than two places), a tangent of the curve, or a secant with coplanar tangent lines. The space of all lines in three space, adding the lines at infinity, form an algebraic set, called the Grassmanian $G(2, 4)$. The set of all multisecants, tangents and secants with coplanar tangent lines form an algebraic subset B of $G(2, 4)$.

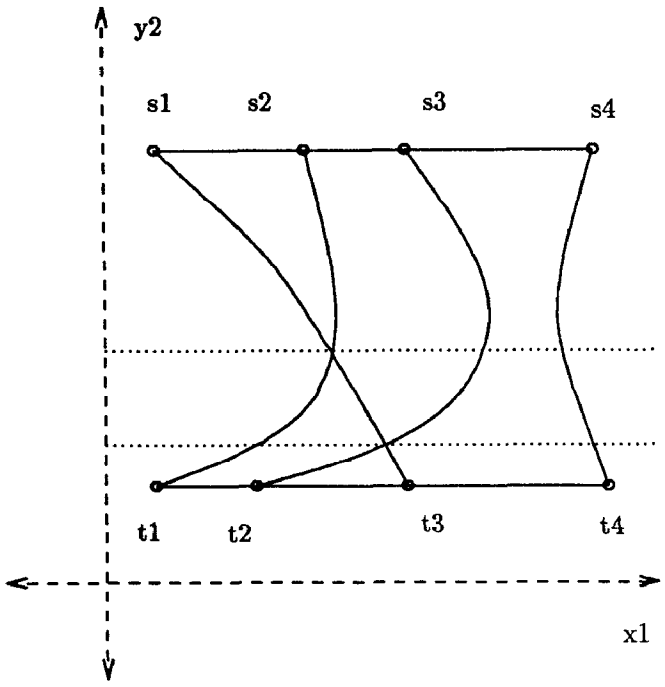

Figure 3: Effect of nodes on adjacency calculations

This fact can be seen as follows. A line in space is given by the intersection of two planes, which provides the local coordinates for $G(2, 4)$. Given the line, we can explicitly give coordinates of points on it as functions of a parameter t. If $P_1(x_1, y_1, y_2)$ and $P_2(x_1, y_1, y_2)$ are the two surfaces defining our space curve, then substituting for points on the line, the intersection of the line with the curve can be determined by examining the order of the roots of the one-variable polynomials $P_1(t)$ and $P_2(t)$. In particular, the set B of multisecants, tangents and secants with coplanar tangent lines can be described by polynomials involving the degrees of $P_1(t)$ and $P_2(t)$. In the space $G(2, 4) \times \mathbf{R}^3$, define the subset $B_L = ((l, p) \in G(2, 4) \times \mathbf{R}^3 : l \in B, p \in l)$. B_L maps, under the projection of $G(2, 4) \times \mathbf{R}^3 \rightarrow \mathbf{R}^3$, to the set of bad points of projection, which is thus described by a polynomial whose degree depends polynomially on the degree of the space curve. Choosing a point not on this set using [Schwartz 80] will guarantee that the projected plane curve has only nodes for singularities.

5 Factorization Information

Once we have computed the adjacency information for the curve skeleton K it is a simple task to recover the number of irreducible factors. By theorem 2, the number of irreducible factors of P equals the number of connected components of $S - Sing(S)$. By theorems 5 and 6, this must also equal the number of connected components of K, and hence $\hat{K}$.

If the polynomial P of degree d has the absolute factorization

$$P = \prod P_i \quad i = 1, \ldots, k$$

where P_i has degree d_i, each vertex in the grid G must have exactly d vertices of K over it. By Bezout's theorem, each factor P_i must generate exactly d_i of these vertices in K. Having identified the connected components in K, we need only count the number of vertices of $\hat{K}$ over any vertex of G that lie in the same connected component of $\hat{K}$. All of these calculations can be performed in NC with respect to the input size measures.

To construct an approximation to the P_i's, we must construct approximations to the points in K. This task entails approximating the roots of univariate polynomial. The best algorithms require time that is polynomial in d ([Pan 85]). We can compute approximations for roughly $d^2/2$ vertices of K lying in the same connected component, and interpolate to recover the factor itself.

Note the only step in computing this approximate factorization that does not run in NC is the univariate factorization step (root approximation). An NC algorithm for univariate factorization would lead directly to an NC algorithm for bivariate factorization, as observed in [Kaltofen 85a].

Acknowledgements

We would like to thank Jim Renegar for his comments and discussion of this work.

References

[Borodin 82] Borodin, A., von zur Gathen, J. and Hopcroft, J. (1982), "Fast parallel matrix and GCD computations," *Inf. and Contr.*, Vol. 52, pp. 241-256.

[Canny 87] Canny, J. (1987), "A New Algebraic Method for Robot Motion Planning and Real Geometry," *28'th Symposium on Foundations of Computer Science*, pp. 39-48.

[Canny 88] Canny, J. (1988), "Some Algebraic and Geometric Computations in PSPACE," *20'th Symposium on Theory of Computing*, pp. 460-467.

[Chistov 83] Chistov, A.L., and Grigoryev, D.Y. (1983), "Subexponential-Time Solving Systems of Algebraic Equations I.," Steklov Institute, LOMI preprint E-9-83.

[Davenport 81] Davenport, J. and Trager, B. (1981), "Factorization over finitely generated fields," *1981 ACM Symposium on Symbolic Algebraic Computation*, pp. 200-205.

[DiCrescenzo 84] DiCrescenzo, C., and Duval, D., (1984) "Computations on Curves", *Eurosam'84*, LICS 174, pp. 100 - 107.

[Duval 87] Duval,D. (1987), *Diverses questions relatives au Calcul Formel Avec Des Nombres Algebriques*, Thèse, L'Université Scientifique, Technologique et Médicale de Grenoble.

[Dvornicich 87] Dvornicich , R., and Traverso, C., (1987) "Newton Symmetric Functions and the Arithmetic of Algebraically Closed Fields", Univ. of Pisa, Manuscript.

[von zur Gathen 83] von zur Gathen, J. (1983), "Factoring Sparse Multivariate Polynomials," *Proc. IEEE Symp. FOCS*, pp. 172-179.

[Griffiths 78] Griffiths, P. and Harris, J. (1978), *Principles of Algebraic Geometry*, John Wiley and Sons

[Hartshorne 77] Hartshorne, R. (1977), *Algebraic Geometry*, Springer-Verlag.

[Heintz 81] Heintz, J. and Sieveking, M. (1981), "Absolute primality of polynomials is decidable in random polynomial time in the number of variables," Proc. 1981 Internat. Conf. Automata, Languages, Prog., *Springer Lec. Notes Comp. Sci.*, Vol. 115, pp. 16-28.

[Henrici 88] Henrici, P. (1988) *Applied and Computational Complex Analysis*, John Wiley and Sons

[Kaltofen 85a] Kaltofen, E. (1985), "Fast Parallel Absolute Irreducibility Testing," *J. Symbolic Computation*, Vol. 1, pp. 57-67.

[Kaltofen 85b] Kaltofen, E. (1985), "Polynomial-Time Reductions from Multivariate to Bi- and Univariate Integral Polynomial Factorization," *SIAM J. Computing*, Vol. 14, pp. 469-489.

[Kaltofen 85c] Kaltofen, E. (1985), "Effective Hilbert Irreducibility," *Inf. and Contr.*, Vol. 66, No. 3, pp. 123-137.

[Lenstra 82] Lenstra, A., Lenstra, H. and Lovasz, L. (1982), "Factoring Polynomials with rational coefficients," *Math. Ann.*, Vol. 261, pp. 515-534.

[Mumford 1970] Mumford, D. (1970), *Algebraic Geometry I: Complex Projective Varieties*, Springer-Verlag.

[Noether 22] Noether, E. (1922), "Ein algebraisches Kriterium fur absolute Irreduzibilitat," *Math. Ann.*, Vol. 85, pp. 26-33.

[Pan 85] Pan, V. (1985), "Fast and Efficient Algorithms for Sequential Evaluation of Polynomial Zeroes and of Matrix Polynomials," *26th IEEE Symposium on Foundations of Computer Science.*

[Schwartz 80] Schwartz, J.T. (1980), "Fast Probabilistic Algorithms for Verification of Polynomial Identities," *Jour. ACM*, Vol. 27, No. 4, pp. 701-717.

[Shafarevich 74] Shafarevich, I. (1974), *Basic Algebraic Geometry*, Springer Verlag.

[Valiant 83] Valiant, L. G., Skyum, S., Berkowitz, S., and Rackoff, C., (1983), "Fast Parallel Computation of Polynomials Using Few Processors," *SIAM J. Comp.*, Vol. 12, No. 4, pp. 641-644.