

Approximate Factorization

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

- 1 The Ruppert Matrix
a criterion for irreducibility
- 2 an Open Problem
polynomial time in symbolic-numeric computing
- 3 The Kernel of the Ruppert Matrix
relation between rank and greatest common divisor
- 4 SVD and Approximate GCD
symbolic-numeric algorithm for approximate factorization

MCS 563 Lecture 28
Analytic Symbolic Computation
Jan Verschelde, 30 March 2011

Approximate Factorization

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

- 1 The Ruppert Matrix
a criterion for irreducibility
- 2 an Open Problem
polynomial time in symbolic-numeric computing
- 3 The Kernel of the Ruppert Matrix
relation between rank and greatest common divisor
- 4 SVD and Approximate GCD
symbolic-numeric algorithm for approximate factorization

taking derivatives

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

Suppose $f = f(x, y)$ is reducible: $f = f_1 f_2$.

Applying the product rule for derivatives gives

$$\frac{\partial f}{\partial x} = \frac{\partial f_1}{\partial x} f_2 + f_1 \frac{\partial f_2}{\partial x} = g_1 + g_2$$

and

$$\frac{\partial f}{\partial y} = \frac{\partial f_1}{\partial y} f_2 + f_1 \frac{\partial f_2}{\partial y} = h_1 + h_2.$$

defining $g_1 = \frac{\partial f_1}{\partial x} f_2$, $g_2 = f_1 \frac{\partial f_2}{\partial x}$, $h_1 = \frac{\partial f_2}{\partial y} f_2$, and $h_2 = f_1 \frac{\partial f_2}{\partial y}$.

Then we write the derivatives of $\log(f_1)$ as

$$\frac{\partial}{\partial x} (\log(f_1)) = \frac{1}{f_1} \frac{\partial f_1}{\partial x} = \frac{g_1}{f} \quad \text{and} \quad \frac{\partial}{\partial y} (\log(f_1)) = \frac{1}{f_1} \frac{\partial f_1}{\partial y} = \frac{h_1}{f}.$$

a partial differential equation

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

For any p with continuous derivatives, the identity

$$\frac{\partial}{\partial \mathbf{x}} \left(\frac{\partial}{\partial \mathbf{y}} p \right) = \frac{\partial}{\partial \mathbf{y}} \left(\frac{\partial}{\partial \mathbf{x}} p \right)$$

holds and we apply it to $p = \log(f_1)$ and $\log(f_2)$ to find

$$\frac{\partial}{\partial \mathbf{x}} \left(\frac{h_1}{f} \right) = \frac{\partial}{\partial \mathbf{y}} \left(\frac{g_1}{f} \right) \quad \text{and} \quad \frac{\partial}{\partial \mathbf{x}} \left(\frac{h_2}{f} \right) = \frac{\partial}{\partial \mathbf{y}} \left(\frac{g_2}{f} \right).$$

The partial differential equation

$$\frac{\partial}{\partial \mathbf{y}} \left(\frac{g}{f} \right) = \frac{\partial}{\partial \mathbf{x}} \left(\frac{h}{f} \right)$$

has nonzero solutions $\Leftrightarrow f$ is reducible.

a criterion for irreducibility

Denote $\deg_x(f)$ (respectively $\deg_y(f)$) as the degree of f when viewed as a polynomial only in x (respectively y).

Theorem (Ruppert's criterion)

A polynomial $f(x, y) \in \mathbb{C}[x, y]$ is irreducible if and only if

$$\frac{\partial}{\partial y} \left(\frac{g}{f} \right) = \frac{\partial}{\partial x} \left(\frac{h}{f} \right)$$

has no nonzero solutions for all polynomial $g, h \in \mathbb{C}[x, y]$, with $\deg_x(g) \leq \deg_x(f) - 1$, $\deg_y(g) \leq \deg_y(f)$, and $\deg_x(h) \leq \deg_x(f)$, $\deg_y(h) \leq \deg_y(f) - 2$.

If the condition $\deg_y(h) \leq \deg_y(f) - 2$ on h is changed into $\deg_y(h) \leq \deg_y(f) - 1$, then $g = f_x$ and $h = f_y$ is a solution to the PDE regardless whether f is irreducible or not.

Degree bounds on g_1, g_2, h_1, h_2 give the Ruppert matrix.

an example

Consider for example $f(x, y) = x^2 + y^2 - 1$. Then
 $g(x, y) = a_{00} + a_{10}x + a_{01}y + a_{11}xy + a_{02}y^2 + a_{12}xy^2$
 and $h(x, y) = b_{00} + b_{10}x + b_{20}x^2$ are
 the general forms of the polynomials to satisfy the PDE.

The sequence of commands in Maple generates the Ruppert matrix for f :

```
[> f := x^2 + y^2 - 1;
[> g := sum(sum(a[i, j]*x^i*y^j,
               i=0..degree(f, x)-1),
             j=0..degree(f, y));
[> h := sum(sum(b[i, j]*x^i*y^j,
               i=0..degree(f, x)),
             j=0..degree(f, y)-2);
```

The Ruppert
Matrix

a criterion for
irreducibility

an Open
Problem

polynomial time in
symbolic-numeric
computing

The Kernel of
the Ruppert
Matrix

relation between
rank and greatest
common divisor

SVD and
Approximate
GCD

symbolic-numeric
algorithm for
approximate
factorization

Maple session continued

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

We setup the Ruppert matrix from the PDE:

```
[> eq := diff(g/f,y) - diff(h/f,x);
[> nq := normal(eq);
[> p := numer(nq);
[> s := coeffs(p,[x,y]);
[> sys := {seq(s[i]=0,i=1..nops([s]))};
[> var := indets(sys);
[> R := LinearAlgebra[GenerateMatrix]
      (sys,var)[1];
[> LinearAlgebra[Rank](R);
```

The rank of the matrix R equals 9, which equals the number of columns, so f is indeed irreducible.

Approximate Factorization

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

- 1 The Ruppert Matrix
a criterion for irreducibility
- 2 an Open Problem
polynomial time in symbolic-numeric computing
- 3 The Kernel of the Ruppert Matrix
relation between rank and greatest common divisor
- 4 SVD and Approximate GCD
symbolic-numeric algorithm for approximate factorization

an open problem

The Ruppert
Matrixa criterion for
irreducibilityan Open
Problempolynomial time in
symbolic-numeric
computingThe Kernel of
the Ruppert
Matrixrelation between
rank and greatest
common divisorSVD and
Approximate
GCDsymbolic-numeric
algorithm for
approximate
factorization

Polynomial factorization is important in computer algebra.

One open problem in symbolic-numeric computing is

“Given is a polynomial $f(x, y) \in \mathbb{Q}[x, y]$ and $\epsilon \in \mathbb{Q}$.

Decide in polynomial time

- *in the degree*
- *and coefficient size*

*if there is a factorizable $\bar{f}(x, y) \in \mathbb{C}[x, y]$ with $\|f - \bar{f}\| \leq \epsilon$,
for a reasonable coefficient vector norm $\|\cdot\|$.”*

E. Kaltofen: Challenges of symbolic computation: my favorite open problems. *J. Symbolic Computation*, 29(6):891–919, 2000.

30 Mar 2011

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

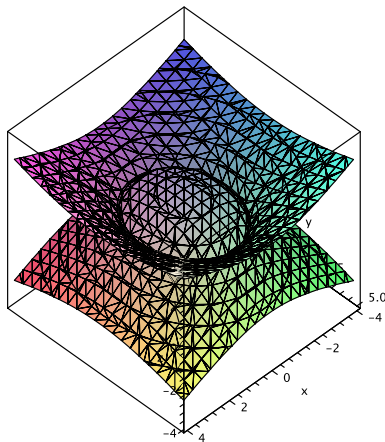
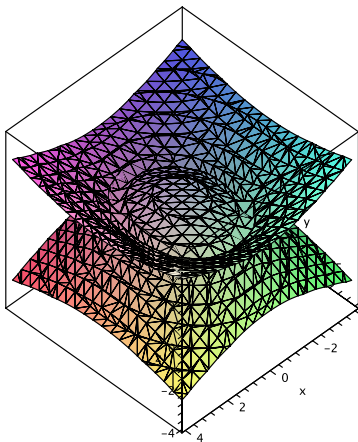
relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

an example

In Maple, we apply `implicitplot3d` on `p` directly and execute `factor(p, sqrt(2))` before plotting the factors.



$$p = (9x^2 + 4y^2 + 18\sqrt{2}z^2 - 36)(9x^2 + 4y^2 - 18\sqrt{2}z^2 - 36).$$

Approximate Factorization

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

- 1 The Ruppert Matrix
a criterion for irreducibility
- 2 an Open Problem
polynomial time in symbolic-numeric computing
- 3 The Kernel of the Ruppert Matrix
relation between rank and greatest common divisor
- 4 SVD and Approximate GCD
symbolic-numeric algorithm for approximate factorization

relaxation of the PDE

We assume the polynomial f we have to factor is square free, i.e.: $\text{GCD}(f, f_x) = 1$.

In the PDE, the condition on h is relaxed to $\deg_y(h) \leq \deg_y(f) - 1$ and the PDE is rewritten into

$$f \cdot \left(\frac{\partial g}{\partial y} - \frac{\partial h}{\partial x} \right) + h \cdot \frac{\partial f}{\partial x} - g \cdot \frac{\partial f}{\partial y} = 0.$$

The relaxation of the condition on h implies that the system of linear equations will have at least one solution; one in the case f is irreducible.

Moreover, the dimension of the solution space equals the number of irreducible factors of f .

Abusing notation, $R(f)$ will still be called the Ruppert matrix, for the matrix resulting of the relaxation on the degree of h .

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

using the GCD

The Ruppert
Matrixa criterion for
irreducibilityan Open
Problempolynomial time in
symbolic-numeric
computingThe Kernel of
the Ruppert
Matrixrelation between
rank and greatest
common divisorSVD and
Approximate
GCDsymbolic-numeric
algorithm for
approximate
factorization

Given a basis of the null space of the Ruppert matrix, how do we recover the irreducible factors?

Proposition 1

Let $f = f(x, y)$, $f = f_1 f_2 \cdots f_s$, and $g_i = \frac{\partial f_i}{\partial \mathbf{x}} \frac{f}{f_i}$, $i = 1, 2, \dots, s$.

$$\mathbf{v} = \sum_{i=1}^s \gamma_i \mathbf{g}_i, \quad \gamma_i \neq \gamma_j, i \neq j$$

$$\Rightarrow f_i = \text{GCD} \left(f, \mathbf{v} - \gamma_i \frac{\partial f}{\partial \mathbf{x}} \right), \quad i = 1, 2, \dots, s.$$

30 Mar 2011

proof of Proposition 1

Proof. To avoid dot dot dots, we assume $s = 3$.

The g_i 's are defined as $g_1 + g_2 + g_3 = \frac{\partial f}{\partial \mathbf{x}}$. Then:

$$\begin{aligned} v - \gamma_1 \frac{\partial f}{\partial \mathbf{x}} &= \gamma_1 g_1 + \gamma_2 g_2 + \gamma_3 g_3 - \gamma_1 (g_1 + g_2 + g_3) \\ &= (\gamma_2 - \gamma_1) g_2 + (\gamma_3 - \gamma_1) g_3 \\ &= (\gamma_2 - \gamma_1) \frac{\partial f_2}{\partial \mathbf{x}} f_1 f_3 + (\gamma_3 - \gamma_1) \frac{\partial f_3}{\partial \mathbf{x}} f_1 f_2. \end{aligned}$$

Because $\gamma_2 \neq \gamma_1$ and $\gamma_3 \neq \gamma_1$ we find

$$\begin{aligned} \text{GCD} \left(f, v - \gamma_1 \frac{\partial f}{\partial \mathbf{x}} \right) &= \text{GCD} \left(f_1 f_2 f_3, \right. \\ &\quad \left. (\gamma_2 - \gamma_1) \frac{\partial f_2}{\partial \mathbf{x}} f_1 f_3 + (\gamma_3 - \gamma_1) \frac{\partial f_3}{\partial \mathbf{x}} f_1 f_2 \right) \\ &= f_1. \end{aligned}$$

The derivations are similar for f_2 and f_3 .

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization



form of the basis

The form of the basis elements of the kernel of the Ruppert matrix is described next:

Proposition 2

Consider $f = f(x, y)$ and $f = f_1 f_2 \cdots f_s$. Denote by $R(f)$ the Ruppert matrix of the relaxed system linear in the coefficient vectors of the polynomials g and h with $\deg(g) \leq (\deg_x(f) - 1, \deg_y(f))$ and $\deg(h) \leq (\deg_x(f), \deg_y(f) - 1)$. Let $\mathbf{u} : R(f)\mathbf{u} = \mathbf{0}$, then $\mathbf{u} = (\mathbf{v}, \mathbf{w})$, where \mathbf{v} and \mathbf{w} are coefficient vectors of the respective polynomials g and h . Identifying the coefficient vector \mathbf{v} with the polynomial $v(x, y)$ we have:

$$v(x, y) = \sum_{i=1}^s \gamma_i g_i(x, y), \quad g_i = \frac{\partial f_i}{\partial \mathbf{x}} \frac{f}{f_i}, \quad i = 1, 2, \dots, s,$$

for some constants $\gamma_i \in \mathbb{C}$.

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

proof of Proposition 2

Proof. Assuming f is monic, we write f as a function of y , expressing the values for the x -coordinates of f as $x_i(y)$, $i = 1, 2, \dots, d$, where $d = \deg_x(f)$:

$$f(x(y), y) = \prod_{i=1}^d (x - x_i(y)).$$

Since $\deg_x(v) < \deg_x(f)$, we have a partial fraction decomposition

$$\frac{v}{f} = \sum_{i=1}^d \frac{a_i(y)}{x - x_i(y)}, \quad a_i(y) = \frac{v(x_i(y), y)}{\prod_{j \neq i} (x - x_j(y))} = \frac{v(x_i(y), y)}{\frac{\partial f}{\partial x}(x_i(y), y)}.$$

We obtain the expression for $a_i(y)$ by equating numerators in the partial fraction decomposition identity for v/f .

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

partial fraction decompositions

For the polynomial $w(x, y)$ with coefficient vector \mathbf{w} , we also set up a partial fraction decomposition:

$$\frac{w}{f} = \sum_{i=1}^d \frac{b_i(y)}{x - x_i(y)} + b_0, \quad b_0 \in \mathbb{C}.$$

Because $\mathbf{u} = (\mathbf{v}, \mathbf{w}) \in \text{kernel}(R(f))$: $\frac{\partial}{\partial y} \left(\frac{\mathbf{v}}{f} \right) = \frac{\partial}{\partial x} \left(\frac{\mathbf{w}}{f} \right)$.

Applying this property to the partial fraction decompositions:

$$\frac{\partial}{\partial y} \left(\frac{w}{f} \right) = \sum_{i=1}^d \frac{-b_i^2}{(x - x_i(y))^2}$$

$$\frac{\partial}{\partial x} \left(\frac{\mathbf{v}}{f} \right) = \sum_{i=1}^d \frac{1}{x - x_i(y)} \frac{\partial a_i}{\partial y} + \sum_{i=1}^d \frac{a_i}{(x - x_i(y))^2} \left(-\frac{\partial x_i}{\partial y} \right).$$

So we find that $\frac{\partial}{\partial y} \left(\frac{\mathbf{v}}{f} \right) = \frac{\partial}{\partial x} \left(\frac{\mathbf{w}}{f} \right)$ implies $\frac{\partial a_i}{\partial y} = 0$.

conclusion of the proof

The Ruppert Matrix

a criterion for
irreducibility

an Open Problem

polynomial time in
symbolic-numeric
computing

The Kernel of the Ruppert Matrix

relation between
rank and greatest
common divisor

SVD and Approximate GCD

symbolic-numeric
algorithm for
approximate
factorization

The constant coefficients a_i belonging to the same factor f_k of f are all conjugated and are all equal, say to γ_k .

So we may write

$$\frac{v}{f} = \sum_{k=1}^s \frac{\gamma_k}{\prod_j (x - x_j(y))} = \sum_{k=1}^s \gamma_k \frac{\partial f_k}{\partial x} \frac{1}{f_k}.$$

Therefore $v = \sum_{k=1}^s \gamma_k \frac{\partial f_k}{\partial x} \frac{f}{f_k} = \sum_{k=1}^s \gamma_k g_k.$



an eigenvalue problem

To recover the g_i 's from the linear combinations:

Proposition 3

Let the matrix $V = [\mathbf{v}_1 \mathbf{v}_2 \cdots \mathbf{v}_s]$ collect the components of the basis vectors of the kernel of the Ruppert matrix $R(f)$, i.e. $R(f)\mathbf{u} = \mathbf{0}$, $\mathbf{u} = (\mathbf{v}, \mathbf{w})$, \mathbf{v} contains the coefficient vectors of the polynomials g of Ruppert's criterion. For any \mathbf{v} in the span of V , there is a unique matrix $A \in \mathbb{C}^{s \times s}$:

$$\mathbf{v}\mathbf{v}_i = \sum_{j=1}^s a_{ij} \mathbf{v}_j \frac{\partial f}{\partial \mathbf{x}} \pmod{f}.$$

Moreover:

$$f = \prod_{\lambda \in \mathbb{C}} \text{GCD} \left(f, \mathbf{v} - \lambda \frac{\partial f}{\partial \mathbf{x}} \right),$$

$$\det(A - \lambda I) = 0$$

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

Proof of Proposition 3

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

Because $\mathbf{v} = \gamma_1 \mathbf{g}_1 + \gamma_2 \mathbf{g}_2 + \cdots + \gamma_s \mathbf{g}_s$, with $g_i = \frac{f}{f_i} \frac{\partial f}{\partial \mathbf{x}}$, the second statement of the proposition follows immediately if $\gamma_i = \lambda_i$, as f_i divides $\mathbf{v} - \lambda_i \frac{\partial f}{\partial \mathbf{x}}$.

Since V is a basis for the null space of the Ruppert matrix, there exists an s -by- s matrix B such that

$$\begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_s \end{bmatrix} = B \begin{bmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \\ \vdots \\ \mathbf{g}_s \end{bmatrix}.$$

30 Mar 2011

computing mod f

We have that $g_i g_j = \left(\frac{\partial f_j}{\partial \mathbf{x}} \prod_{\substack{k=1 \\ \neq i}}^s f_k \right) \left(\frac{\partial f_i}{\partial \mathbf{x}} \prod_{\substack{k=1 \\ \neq j}}^s f_k \right)$ is a multiple of f for $i \neq j$, so $g_i g_j \equiv 0 \pmod{f}$. Then we can write

$$\mathbf{v} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_s \end{bmatrix} \equiv B \begin{bmatrix} \mathbf{v}g_1 \\ \mathbf{v}g_2 \\ \vdots \\ \mathbf{v}g_s \end{bmatrix} \equiv B \begin{bmatrix} \lambda_1 g_1^2 \\ \lambda_2 g_2^2 \\ \vdots \\ \lambda_s g_s^2 \end{bmatrix} \equiv B \Lambda \begin{bmatrix} g_1^2 \\ g_2^2 \\ \vdots \\ g_s^2 \end{bmatrix} \pmod{f}.$$

$$\text{with } \Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_s \end{bmatrix}$$

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

computing mod f continued

The multiplication of $\frac{\partial f}{\partial \mathbf{x}}$ with V leads to

$$\frac{\partial f}{\partial \mathbf{x}} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_s \end{bmatrix} \equiv B \begin{bmatrix} \frac{\partial f}{\partial \mathbf{x}} g_1 \\ \frac{\partial f}{\partial \mathbf{x}} g_2 \\ \vdots \\ \frac{\partial f}{\partial \mathbf{x}} g_s \end{bmatrix} \equiv B \begin{bmatrix} g_1^2 \\ g_2^2 \\ \vdots \\ g_s^2 \end{bmatrix} \pmod{f}, \text{ as } \frac{\partial f}{\partial \mathbf{x}} = \sum_{i=1}^s g_i.$$

So we substitute

$$\begin{bmatrix} g_1^2 \\ g_2^2 \\ \vdots \\ g_s^2 \end{bmatrix} \equiv B^{-1} \frac{\partial f}{\partial \mathbf{x}} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_s \end{bmatrix} \pmod{f}$$

into the previous derivation for $\mathbf{v}\mathbf{v}_i$ and find that $A = B\Lambda B^{-1}$ has eigenvalues $\lambda_i, i = 1, 2, \dots, s$. □

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

Approximate Factorization

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

- 1 The Ruppert Matrix
a criterion for irreducibility
- 2 an Open Problem
polynomial time in symbolic-numeric computing
- 3 The Kernel of the Ruppert Matrix
relation between rank and greatest common divisor
- 4 SVD and Approximate GCD
symbolic-numeric algorithm for approximate factorization

outline of the method

The Ruppert
Matrix

a criterion for
irreducibility

an Open
Problem

polynomial time in
symbolic-numeric
computing

The Kernel of
the Ruppert
Matrix

relation between
rank and greatest
common divisor

SVD and
Approximate
GCD

symbolic-numeric
algorithm for
approximate
factorization

- 1 remove multiple factors with gcd
- 2 compute $s := \text{Rank}(\text{Null}(R(f)))$
and a basis for $\text{Null}(R(f))$
return if $s = 1$
- 3 compute matrix A and its eigenvalues λ_i
- 4 $f_i := \text{GCD} \left(f, \mathbf{v} - \lambda_i \frac{\partial f}{\partial \mathbf{x}} \right)$ for $i = 1, 2, \dots, s$
and for any $\mathbf{v} \in \text{Null}(R(f))$
- 5 apply Gauss-Newton on $f - f_1 f_2 \cdots f_s$

Approximate Bivariate Factorization

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

Input: $f \in \mathbb{C}[x, y]$, $\text{GCD}(f, \frac{\partial f}{\partial x}) = 1$,
 f has no approximate factors in $\mathbb{C}[y]$;
 $S \subset \mathbb{C}$ and $\#S \geq \deg_x(f) \times \deg_y(f)$.

Output: list of approximate factors of f .

Stage 1: form the Ruppert matrix $R(f)$;
 find the last $\deg(f) + 1$ singular values σ_i of $R(f)$,
 $\sigma_n \geq \sigma_{n-1} \geq \dots \geq \sigma_2 \geq \sigma_1$;
 let s be the index so σ_{s+1}/σ_s is maximal;
 if $s = 1$, then return f ;
 form a basis $\mathbf{v}_1, \mathbf{v}_1, \dots, \mathbf{v}_s$ from the last
 s right singular vectors of $R(f)$;

stages 2 and 3

The Ruppert
Matrixa criterion for
irreducibilityan Open
Problempolynomial time in
symbolic-numeric
computingThe Kernel of
the Ruppert
Matrixrelation between
rank and greatest
common divisorSVD and
Approximate
GCDsymbolic-numeric
algorithm for
approximate
factorization

Stage 2: $\mathbf{v} := \sum_{s_i \in S} s_i \mathbf{v}_i$, with coefficients s_i

selected uniformly and independently;

for $y = \alpha$, compute a_{ij} that minimize

$$\left\| \text{remainder} \left(\mathbf{v} \mathbf{v}_i - \sum_{j=1}^s a_{ij} \mathbf{v}_j \frac{\partial f}{\partial \mathbf{x}}, f \right) \right\|_2;$$

compute the eigenvalues λ_i of $A = [a_{ij}]$;

Stage 3: $f_i := \text{GCD} \left(f, \mathbf{v} - \lambda_i \frac{\partial f}{\partial \mathbf{x}} \right)$, for $i = 1, 2, \dots, s$,
where GCD is an approximate GCD.

Summary + Exercises

The Ruppert Matrix

a criterion for irreducibility

an Open Problem

polynomial time in symbolic-numeric computing

The Kernel of the Ruppert Matrix

relation between rank and greatest common divisor

SVD and Approximate GCD

symbolic-numeric algorithm for approximate factorization

Numerical rank via SVD, least squares, eigenvalues, and approximate GCD are key to a symbolic-numeric algorithm for approximate bivariate factorization.

Exercises:

- 1 Find a general formula for the size of the Ruppert matrix, in terms of the degrees $\deg_x(f)$ and $\deg_y(f)$.
- 2 Show that for $f = f(x, y)$, $f = f_1 f_2$, $g_1 = \frac{\partial f_1}{\partial x} f_2$, $g_2 = f_1 \frac{\partial f_2}{\partial x}$, $h_1 = \frac{\partial f_2}{\partial y} f_2$, and $h_2 = f_1 \frac{\partial f_2}{\partial y}$:

$$f \left(\frac{\partial g_1}{\partial y} - \frac{\partial h_1}{\partial x} \right) + h_1 \frac{\partial f}{\partial x} - g_1 \frac{\partial f}{\partial y} = 0.$$

3 Download the Maple code at

http://www4.ncsu.edu/~kaltofen/software/appfac/issac04_mws/multifac_1.3.mpl and use it to factor

$$f(x, y) = 9 + 23y^2 + 13yx^2 + 6y + 7y^3 + 13y^2x^2 + x^4 + 6yx^4 + x^6.$$

4 Download ApaTools available via the homepage of Zhonggang Zeng and use it to factor f from the previous exercise.

- 5** Consider f from above, but now add some random errors to the coefficients, of magnitude 10^{-k} , for k ranging from 1 to 14. For $k = 1$, f is irreducible, while for $k = 14$, the numerical algorithm should return the same factorization as in the exact case. For which k is f no longer irreducible?