When are two polynomials numerically relatively prime?

George Labahn

University of Waterloo
Waterloo, Canada

Joint work with Bernhard Beckermann
Université des Sciences et Technologies de Lille

EXAMPLE

(1)

$$\begin{array}{rclcl} a(z) = & (z - \frac{1}{3}) \cdot (z - \frac{5}{3}) & \longrightarrow & z^2 - 2z + .56 \\ b(z) = & z - \frac{1}{3} & \longrightarrow & z - .33 \end{array}$$

(2)

$$\begin{array}{rclcl} a(z) = & 50 - 7z & \longrightarrow & 50 - 7z \\ b(z) = & z - \frac{1}{7} & \longrightarrow & z - .14 \end{array}$$

$\mathbf{GCD}(a(z), b(z)) = ?.$

$\mathbf{NORMAL}(\frac{a(z)}{b(z)}) = ?$

Answer: Error? convert/exact? Euclid Algorithm?

Example: Euclids Algorithm

$a(z) = a(z) = z^4 + z^3 + (1+\epsilon)\, z^2 + \epsilon z + 1$

$b(z) = z^3 - z^2 + 3\, z - 2$

$c(z) = \mathrm{rem}(a, b) = \epsilon z^2 + (\epsilon - 4) z + 5.$

$d(z) = \mathrm{rem}(b, c) = (5 - \frac{17}{\epsilon} + \frac{16}{\epsilon^2}) z + (-2 + \frac{10}{\epsilon} - \frac{20}{\epsilon^2})$

$e(z) = \mathrm{rem}(c, d) = \frac{\epsilon^2\left(-39\,\epsilon^2 - 32\,\epsilon + 137 + 14\,\epsilon^3\right)}{\left(5\,\epsilon^2 - 17\,\epsilon + 16\right)^2}$

Hence $\mathbf{GCD}(a, b) = 1$

Problem: What happens if ϵ is small numeric?

ϵ less than tolerance - good

ϵ a little over tolerance - horrible errors

Our Problem

Given $a, b \in \mathbb{C}[z]$.

When are a and b relatively prime even after "small" perturbations of their coefficients?

- Common case when computing numerical gcd.
- Want *fast* and *numerically correct* method of determining relative primeness.
- In our case numerically correct means a weakly stable algorithm.

Outline

- Preliminaries: Exact Computation
- Numerically relatively prime.
  - Basic concepts.
  - Inverse Formula for Sylvester Matrix
  - Parameter for "relative prime" measure
  - Bounds
  - Computation: weakly stable algorithm

Exact Computation of GCD

Exact Computation

Solving problems in symbolic environments:

- **Basic coefficient domain:**
  - Quotient field: $\mathbb{Q}(\alpha_1, \ldots, \alpha_k)$
  - symbols are first class objects in CA environments.
- **Basic fact I:**
  - Polynomial arithmetic easier than arithmetic with rational functions

$$\frac{a(x)}{b(x)} + \frac{c(x)}{d(x)} = \frac{a(x) \cdot d(x) + b(x) \cdot c(x)}{b(x) \cdot d(x)}$$

  - Need to normalize out gcd's at every step if want to recognize 0 at later computations.
- **Basic fact II:**
  - To work with polynomial arithmetic in integral domain (e.g. in $\mathbb{Q}[\alpha_1, \ldots, \alpha_k]$) rather than in quotient field.
- **Method I:**
  Do our arithmetic **fraction-free** but at the same time to minimize growth of intermediate computation.

Example: Gaussian Elimination

$$A = \begin{bmatrix} a & b & c & \cdots & \cdots \\ d & e & f & \cdots & \cdots \\ g & h & i & \cdots & \cdots \\ \vdots & \vdots & \vdots & & \end{bmatrix} \approx \begin{bmatrix} a & b & c & \cdots & \cdots \\ 0 & \tilde{e} & \tilde{f} & \cdots & \cdots \\ 0 & \tilde{h} & \tilde{i} & \cdots & \cdots \\ \vdots & \vdots & \vdots & & \end{bmatrix}$$

- Only using cross multiplication results in exponential growth of coefficients: $\mathrm{O}(2^n \cdot N^2)$ in $n \times n$ case (since coefficients of matrix double in size at each elimination row - here N is bound for size of entries).

- Fraction-free Gaussian elimination (FFGE) : Bareiss (1968), observes a common divisor after 2 steps

$$A \approx \begin{bmatrix} a & b & c & \cdots & \cdots \\ 0 & \tilde{e} & \tilde{f} & \cdots & \cdots \\ 0 & 0 & a(..) & \cdots & a(...) \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & a(..) & \cdots & a(...) \end{bmatrix}.$$

Simple observation allows for linear growth of coefficient size. Cost $\mathcal{O}(n^5 \cdot N^2)$.

Example: Euclidean Algorithms

$a(x), b(x) \in \mathbf{Z}[x]$ given by (Knuth)

$$\begin{aligned} a(x) &= x^8 + x^6 - 3x^4 - 3x^3 + 8x^2 + 2x - 5 \\ b(x) &= 3x^6 + 5x^4 - 4x^2 - 9x + 21 \end{aligned}$$

- Euclidean Algorithm over $\mathbb{Q}[x]$ (monic)
$$\begin{aligned} r_1(x) &= x^4 - \tfrac{1}{5}x^2 + \tfrac{3}{5} \\ r_2(x) &= x^2 + \tfrac{25}{13}x - \tfrac{49}{13} \\ r_3(x) &= x - \tfrac{6150}{4663} \\ r_4(x) &= 1 \end{aligned}$$

- Euclidean-like Algorithm over $\mathbf{Z}[x]$ (Cross multiplication)
$$\begin{aligned} r_1(x) &= 15\,x^4 - 3\,x^2 + 9 \\ r_2(x) &= 15795\,x^2 + 30375\,x - 59535 \\ r_3(x) &= 1254542875143750\,x - 1654608338437500 \\ r_4(x) &= 12593338795500743100931141992187500 \end{aligned}$$

- Fraction-free Algorithm over $\mathbf{Z}[x]$ (Removing contents)
$$\begin{aligned} r_1(x) &= 5\,x^4 - x^2 + 3 \\ r_2(x) &= 13\,x^2 + 25\,x - 49 \\ r_3(x) &= 4663\,x - 6150 \\ r_4(x) &= 1 \end{aligned}$$

- Fraction-free Algorithm over $\mathbf{Z}[x]$ (Subresultants)
$$\begin{aligned} r_1(x) &= 15\,x^4 - 3\,x^2 + 9 \\ r_2(x) &= 65\,x^2 + 125\,x - 245 \\ r_3(x) &= 9326\,x - 12300 \\ r_4(x) &= 260708 \end{aligned}$$

Observations

- Gcd Examples:

$$R_0(x) = a(x), R_1(x) = b(x)$$

$$\alpha_i \cdot R_{i-1}(x) = Q_i(x) \cdot R_i(x) + \beta_i \cdot R_{i+1}(x)$$

Let $r_i = lcoeff\ R_i(x)$ and $\delta_i = \deg\ R_{i-1}(x) - \deg\ R_i(x)$.

  - Cross multiplication: $\alpha_i = r_i^{\delta_i+1}$, $\beta_i = 1$.
  - Content removal:
$$\begin{array}{rcl} \alpha_i & = & r_i^{\delta_i+1} \\ \beta_i & = & \text{content prem}(R_{i-1}(x), R_i(x)) \end{array} .$$
  - Reduced: $\alpha_i = r_i^{\delta_i+1}$, $\beta_i = \alpha_{i-1}$.
  - Subresultant: $\alpha_i = r_i^{\delta_i+1}$, $\beta_1 = (-1)^{\delta_1+1}, \beta_i = -r_{i-1} \cdot \phi_i^{\delta_i}$

    where $\phi_1 = -1, \phi_i = (-r_{i-1})^{\delta_{i-1}} \cdot \phi_{i-1}^{1-\delta_{i-1}}$.

- Computations kept in domain of coefficients
- Determines a common factor without any work!
- Some references:
  - Fraction-free Gaussian : Bareiss (1968)
  - Fraction-free GCD : Habicht (1948), Brown (1971), Collins (1967)

Aside: GCD in Maple

- Modular method:
  - $\text{GCD}(a(x), b(x)) = 1$ in $\mathbf{Z}_{23}[x]$

    $\Longrightarrow \text{GCD}(a(x), b(x)) = 1$ in $\mathbf{Z}[x]$

- Hensel lifting
- Things are not always as expected in Symbolic Computation.

  E.g. GCD Heuristic (Char, Geddes, Gonnet '89)

$$\begin{aligned} a(x) &= 6x^4 + 21x^3 + 35x^2 + 27x + 7 \\ b(x) &= 12x^4 - 3x^3 - 17x^2 - 45x + 21 \end{aligned}$$

  - Evaluate at point:

    E.g. $a(100) = 621352707, \;\; b(100) = 1196825521$

  - Integer GCD $= 30607 = 3 \cdot 100^2 + 6 \cdot 100 + 7$
  - Set $c(x) = 3x^2 + 6x + 7$.
    * $c(x) \mid a(x)$ and $c(x) \mid b(x)$
    * Implies that $\text{GCD}(a(x), b(x)) = c(x)$.
  - Often works.

Numeric Computation of some GCD's

Preliminaries: NORMS:

$c \in \mathbb{C}[z]$, $c(z) = c_0 + .. + c_n z^n$ set $\vec{c} = (c_0, .., c_n)^T$

Norm for $\mathbb{C}[z]$: $||c|| = ||\vec{c}||_1 = \Sigma_j \, |c_j|$

Norm for $\mathbb{C}[z]^{r \times s}$:

$$||(c_{j,k})|| = ||(||c_{j,k}||)||_1 = \max_k \sum_j ||c_{j,k}||.$$

NOTE:

$$||c \cdot d|| \leq ||c|| \cdot ||d||$$

DEFINITION: ϵ-prime:

We are interested in computing:

$$\begin{aligned}\epsilon(a,b) := \inf\{||(a-a^*, b-b^*)|| : &\\ (a^*,b^*) \text{ have common root}, &\\ \partial a^* \le m, \partial b^* \le n\}, &\end{aligned}$$

Thus any polynomials a^*, b^* satisfying

$$||(a-a^*, b-b^*)|| \le \epsilon < \epsilon(a,b)$$

and the above degree restrictions are coprime.

We will then refer to a, b as being ϵ–**prime**.

Problem: How to compute $\epsilon(a,b)$?

Answer: Use linear algebra to approximate $\epsilon(a,b)$.

Alternate Form for $\epsilon(a,b)$

$$\epsilon(a,b) = \inf_{\zeta\in\hat{\mathbb{C}}} ||(\frac{a(\zeta)}{||(1,\zeta^m)||}, \frac{b(\zeta)}{||(1,\zeta^n)||})||$$

where $\hat{\mathbb{C}} := \mathbb{C} \cup \{\infty\}$. Infimum attained at ζ^*, "CCR".

Problem: How to get approximate to $\epsilon(a,b)$

(which is better than previous estimates) ?

Better?

Find a τ such that

$$\tau_{previous} \leq \tau \leq \epsilon(a,b).$$

where τ is numerically computable.

Sylvester Matrix

Sylvester Matrix for polynomials $a(z)$, $b(z)$
$\deg(a) = m$, $\deg(b) = n$:

$$S(a,b) = \underbrace{\left[\begin{array}{cccc} a_0 & 0 & \cdots & 0 \\ a_1 & a_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ a_m & & \ddots & a_0 \\ 0 & a_m & & a_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & a_m \end{array}\right.}_{n} \underbrace{\left.\begin{array}{cccc} b_0 & 0 & \cdots & 0 \\ b_1 & b_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ b_n & & \ddots & b_0 \\ 0 & b_n & & b_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & b_n \end{array}\right]}_{m} \in F^{(m+n)\times(m+n)}.$$

Well known:

- $\det(S(a,b)) \neq 0 \Leftrightarrow \mathrm{Gcd}(a,b) = 1$.
- $\det(S(a,b)) \neq 0 \Leftrightarrow$ can solve $a \cdot v + b \cdot u = 1$.

Lower Bound for $\epsilon(a, b)$

For any two polynomials a and b we have

$$\epsilon(a, b) \geq \frac{1}{||S(a, b)^{-1}||_1}.$$

Proof:
$||S(a, b)||_1 = ||(a, b)||$ where $||B||_1 = \max_{x \neq 0} ||Bx||_1 / ||x||$

$$\begin{aligned} \epsilon(a, b) &= \inf\{||S(a, b) - S(a^*, b^*)||_1 : S(a^*, b^*) \text{ singular}\} \\ &\geq \min\{||S(a, b) - B||_1 : \ B \text{ singular}\} = \frac{1}{||S(a, b)^{-1}||_1} \end{aligned}$$

$\square$

- **Note**: In the case of the 2-norm, we have

  $$\sigma_j = \min\{||S(a, b) - B||_2 : \ \text{rank}(B) = j - 1\},$$

  with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{m+n}$ being the singular values of $S(a, b)$. Allows one to define the ϵ–rank and ϵ–GCD c.f. Corless, Gianni, Trager and Watt.

- **Problem**: SVD is expensive.

- Also would like to improve on lower bound

Inverse of Sylvester Matrix

- $\det(S(a,b)) \neq 0 \Leftrightarrow$ can solve $a \cdot v + b \cdot u = 1$
- $\det(S(a,b)) \neq 0 \Leftrightarrow$ can solve
$$S(a,b) \cdot [v_0, \cdots, v_{n-1}, u_0, \cdots, u_{m-1}]^T = [1, 0, \cdots, 0]^T.$$
- Only need to know that a single column of inverse exists!
- If
$$f(z) = f_{-1}z^{-1} + \ldots + f_{1-m-n}z^{1-m-n} = \frac{u(z)}{a(z)} + \mathcal{O}(z^{-m-n})_{z\to\infty}$$
then $S(a,b)^{-1}$ is:

$$\begin{bmatrix} v_0 & 0 & \cdots & \cdots & \cdots & 0 & b_0 & 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & & \vdots & \vdots & \ddots & \ddots & & & \vdots \\ v_{n-1} & \cdots & v_0 & 0 & \cdots & 0 & b_{n-1} & \cdots & b_0 & 0 & \cdots & 0 \\ u_0 & 0 & \cdots & \cdots & \cdots & 0 & -a_0 & 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & & \vdots & \vdots & \ddots & \ddots & & & \vdots \\ u_{m-1} & \cdots & u_0 & 0 & \cdots & 0 & -a_{m-1} & \cdots & -a_0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \\ 0 & f_{-1} & \cdots & f_{1-m-n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & f_{-1} \end{bmatrix}$$

Proof: Cofactor equation gives:
$$u(z)/a(z) + v(z)/b(z) = 1/[a(z) \cdot b(z)] = \mathcal{O}(z^{-m-n})_{z\to\infty}$$
and so
$$f(z) = -v(z)/b(z) + \mathcal{O}(z^{-m-n})_{z\to\infty}.$$
Then just multiply $S(a,b)$ by above to get identity. $\square$

Approximating $|||S(a,b)^{-1}|||$

- From inverse formula we get

$$\left\| \begin{pmatrix} v \\ u \end{pmatrix} \right\| \leq ||S(a,b)^{-1}||_1 \leq \left\| \begin{pmatrix} v \\ u \end{pmatrix} \right\| + 2 \cdot ||f|| \cdot ||(a,b)||.$$

- **Problem:** how large is $||f||$?

- Solve

$$a(z) \cdot \underline{v}(z) + b(z) \cdot \underline{u}(z) = z^{m+n-1}$$

$$\text{i.e. } S(a,b) \cdot \begin{pmatrix} \underline{\vec{v}} \\ \underline{\vec{u}} \end{pmatrix} = (0, \ldots, 0, 1)^T.$$

- Let

$$\kappa := \left\| \begin{pmatrix} v & \underline{v} \\ u & \underline{u} \end{pmatrix} \right\| = \max\{ \left\| \begin{pmatrix} v \\ u \end{pmatrix} \right\|, \left\| \begin{pmatrix} \underline{v} \\ \underline{u} \end{pmatrix} \right\| \}.$$

Answer:

$$\kappa \leq ||S(a,b)^{-1}||_1 \leq \kappa + 2 \cdot ||f|| \cdot ||(a,b)||$$

$$||f|| = ||\underline{v} \cdot u - \underline{u} \cdot v|| \leq \kappa^2$$

Improved Bounds for $\epsilon(a,b)$

- Know:

$$\min\{\epsilon(a,b), \frac{1}{\kappa}\} \geq \frac{1}{||S(a,b)^{-1}||_1} \geq \frac{1}{\kappa + 2 \cdot ||f|| \cdot ||(a,b)||}$$

- Recall

$$\epsilon(a,b) = \inf_{\zeta \in \hat{\mathbb{C}}} ||(\frac{a(\zeta)}{||(1,\zeta^m)||}, \frac{b(\zeta)}{||(1,\zeta^n)||})||$$

where $\hat{\mathbb{C}} := \mathbb{C} \cup \{\infty\}$. Infimum attained at ζ^*, "CCR".

- If $y(\zeta) = (1, \zeta, ..., \zeta^{m+n-1})$ then

$$\begin{aligned} \epsilon(a,b) &= \min_{\zeta \in \hat{\mathbb{C}}} \frac{||y(\zeta) \cdot S(a,b)||_1}{||y(\zeta)||_1} \\ &\overbrace{\geq}^{?} \min\{\frac{1}{||S(a,b)^{-1} \cdot e_1||_1}, \frac{1}{||S(a,b)^{-1} \cdot e_{m+n}||_1}\}? \end{aligned}$$

- Yes! Implies that $\epsilon(a,b) \geq 1/\kappa$. Better bound.
- CCR in unit disk gives even better bound

$$\epsilon(a,b) \geq \frac{1}{\left|\left|\begin{pmatrix} v(z) \\ u(z) \end{pmatrix}\right|\right|} \geq \frac{1}{||S(a,b)^{-1}||}.$$

Example 1:

$a(z) = z^m - 1$, $|b| = 1$, deg $b = m$.

For any m-th root of unity ω let $B = \min|b(\omega^j)|$. Then

$$B \geq \epsilon(a, b) \geq \frac{1}{\|S(a, b)^{-1}\|} \geq \frac{B}{2\sqrt{m}}.$$

Example 2:

$a(z) = z^m$, $|b| = \frac{1-2z}{3}$

Then

$$\epsilon(a, b) = b(\frac{1}{3}) = \frac{1}{3^m}$$

$$\frac{1}{\|S(a, b)^{-1}\|} \approx \frac{1}{\|(u, v)^T\|} \approx \frac{\sqrt{\pi m}}{8^m}.$$

Computation of κ

How to compute the cofactors?

Fast methods: $O(m^2)$ or even $O(m \cdot \log^2 m)$

We want a (weakly) stable numeric algorithm

Choices:

- Euclidean Algorithm
- Padé Approximation of $-b/a$
- Hermite-Padé Approximation
- Structured (Hankel, Toeplitz, Sylvester) linear systems

Noda and Saski (91)

Bini, Pan (94)

Heinig and Bojanczyk (95)

Cabay and Meleshko (91, 93)

Cabay, Jones and Labahn (96)

Beckermann (96)

Numerical Euclidean Algorithm?

- $r_0 \leftarrow a, r_1 \leftarrow b, j \leftarrow 1$

  WHILE $r_j \neq 0$

$$\begin{aligned} q_j &\leftarrow \frac{r_{j-1}}{r_j} + \mathcal{O}(z^{-1})_{z\to\infty}, \\ r_{j+1} &\leftarrow r_{j-1} - q_j r_j, \\ j &\leftarrow j+1. \end{aligned}$$

  OUTPUT : $GCD(a,b) = r_{j-1}$

- Cofactors calculated via extended Euclid algorithm.

- **EXAMPLE:** (using 2 digits precision)

$$\begin{array}{rlcl} a(z) = r_0(z) = & \frac{1}{3}z^3 + \frac{2}{3}z & \longrightarrow & .33z^3 + .67z \\ b(z) = r_1(z) = & \frac{1}{7}z^2 - \frac{2}{7} & \longrightarrow & .14z^2 + .29 \end{array}$$

$$r_2(z) \qquad\qquad \longleftarrow -0.1z^3 + 1.4z??$$

How to truncate? Here $\epsilon(a,b) = \frac{3}{10}$.

Why Not Euclids Algorithm?

$a(z) = 1 - z + (1+\epsilon)\, z^2 - (2+\epsilon)\, z^3 + z^4$

$b(z) = 1 - 2\,z + 3\,z^2 - 2\,z^3$

$$S(a,b) = \begin{bmatrix} 1 & 0 & 0 & -2 & 0 & 0 & 0 \\ -2-\epsilon & 1 & 0 & 3 & -2 & 0 & 0 \\ 1+\epsilon & -2-\epsilon & 1 & -2 & 3 & -2 & 0 \\ -1 & 1+\epsilon & -2-\epsilon & 1 & -2 & 3 & -2 \\ 1 & -1 & 1+\epsilon & 0 & 1 & -2 & 3 \\ 0 & 1 & -1 & 0 & 0 & 1 & -2 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Euclids algorithm solves linear subsystems.

$$\begin{bmatrix} 1 & 1 & 0 \\ -1 & -2 & 1 \\ 1+\epsilon & 3 & -2 \end{bmatrix} \qquad \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ -1 & 1 & -2 & 1 & 0 \\ 1+\epsilon & -1 & 3 & -2 & 1 \\ -2-\epsilon & 1+\epsilon & -2 & 3 & -2 \\ 1 & -2-\epsilon & 0 & -2 & 3 \end{bmatrix}$$

Problem: What happens when subsystems ill-conditioned?

Why Not Euclids Algorithm?

$a(z) = 1 + \epsilon z + (1+\epsilon)\, z^2 + z^3 + z^4$

$b(z) = -2 + 3\, z - z^2 + z^3$

$$S(a,b) = \begin{bmatrix} 1 & 0 & 0 & -2 & 0 & 0 & 0 \\ \epsilon & 1 & 0 & 3 & -2 & 0 & 0 \\ 1+\epsilon & \epsilon & 1 & -1 & 3 & -2 & 0 \\ 1 & 1+\epsilon & \epsilon & 1 & -1 & 3 & -2 \\ 1 & 1 & 1+\epsilon & 0 & 1 & -1 & 3 \\ 0 & 1 & 1 & 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Euclids algorithm solves linear subsystems.

$$\begin{bmatrix} 1+\epsilon & -1 & 3 \\ 1 & 1 & -1 \\ 1 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} \epsilon & 1 & 3 & -2 & 0 \\ 1+\epsilon & \epsilon & -1 & 3 & -2 \\ 1 & 1+\epsilon & 1 & -1 & 3 \\ 1 & 1 & 0 & 1 & -1 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Problem: What happens when subsystems ill-conditioned?

Unimodular Reduction

- $\partial a = m > \partial b$. Euclid's algorithm can be given as:

  $$(r_{j-1}, r_j)\cdot\begin{pmatrix} 0 & 1 \\ 1 & -q_j \end{pmatrix} = (r_j, r_{j+1}), \quad \partial q_j = \partial r_{j-1} - \partial r_j > 0.$$

- Accumulating gives:

  $$(a, b) \cdot U = (\tilde{a}, \tilde{b}), \quad m > \partial\tilde{a} := k = m - \tilde{m} > \partial\tilde{b}$$

  with degree bounds

  $$\partial U \le \begin{pmatrix} k-2 & k-1 \\ k-1 & k \end{pmatrix}.$$

- If $\det U \neq 0$ then $< a, b > = < \tilde{a}, \tilde{b} >$.

  Proof:

  U is invertible as a matrix polynomial: If $U = \begin{pmatrix} u_1 & u_2 \\ u_3 & u_4 \end{pmatrix}$ then

  $$\underbrace{\begin{pmatrix} a & b \\ 0 & 1 \end{pmatrix}}_{\deg\det(\cdot)=m} \cdot U = \underbrace{\begin{pmatrix} \tilde{a} & \tilde{b} \\ u_3 & u_4 \end{pmatrix}}_{\deg\det(\cdot)\le\tilde{m}+(m-\tilde{m})}.$$

  Therefore $\deg\det U \le 0$, and $\det U = \det U(0) = \frac{lc(u_4)\cdot lc(\tilde{a})}{lc(a)}$. $\square$

- Called **unimodular reduction (UR)**.

Why Unimodular Reduction?

- When it exists a UR $U^{(k)}$ of order k is unique (up to normalization)

- One obtains the cofactors at last step.

$$U^{(m)} = \begin{bmatrix} v & -b \\ u & a \end{bmatrix}$$

- Can obtain inverse formulae for Trudi submatrices. This allows us to estimate condition numbers of Trudi submatrices. Estimate uses UR of order k.

- Bound for $\epsilon(a, b)$ a function of $\epsilon(a^{(k)}, b^{(k)})$

- Recursive compute: $U^{(k+s)}$ from a UR of order s for $(a^{(k)}, b^{(k)})$

$$U^{(k+s)} = U^{(k)} \cdot \bar{U}^{(s)}$$

- Numerically stable version of Euclids algorithm. Stability achieved by *jumping* over ill-conditioned subsystems.

- Can do an error analysis

Algorithm

Algorithm:

Method:	Iteratively construct unimodular reductions (for $\tilde{m} = m-1, m-2, ..., 1, 0$).
Single Step:	New pair $(\tilde{a}, \tilde{b})$ from old with unimodular reduction Method: Solve linear system from equality of coeffs.
Exit:	When $\tilde{m} = 0$ or when $\|\|\tilde{b}\|\|$ is “small” (and hence $\epsilon(\tilde{a}, \tilde{b})$ is small.)
Scaling:	$\|\|U\|\| = 1$ for accummulative matrix.
Num. degree:	When $deg\,\tilde{b} =: \tilde{n} = \tilde{m} - 1$.
Continuation:	Via ‘Look–ahead’, when: $\|\det U(0)\|$ not ‘too small’ (note: $\|\det U(0)\| < 1$ because of scaling).
Complexity:	In most cases: $\mathcal{O}(m^2)$.

What happens if two polynomials are **not** relatively prime?

Conjecture 1 : Last well-conditioned reduction gives the degree of the numerical gcd.

Conjecture 2 : Last well-conditioned reduction gives the numerical gcd.

k	q_k	r_k
-1		$z^4+z^3+z^2+1$
0	$z+2$	$z^3-z^2+3\,z-2$
1	$-\frac{z^2}{4}-\frac{z}{16}-\frac{53}{64}$	$-4\,z+5$
2	$-\frac{256\,z}{137}+\frac{320}{137}$	$\frac{137}{64}$
3		0
-1		$z^4+z^3+(1+\epsilon)\,z^2+\epsilon z+1$
0	$z+2$	$z^3-z^2+3\,z-2$
1	$\frac{4}{\epsilon^2}+\frac{z-2}{\epsilon}$	$z^2\epsilon+(\epsilon-4)\,z+5$
2	$-\frac{\epsilon^2}{4}+\left(-1/8+\frac{z}{16}\right)\epsilon^3$	$\left(\frac{16}{\epsilon^2}-\frac{17}{\epsilon}\right)z-\frac{20}{\epsilon^2}+\frac{10}{\epsilon}$
3		$\frac{17\,\epsilon^2 z^2}{16}-\frac{11\,\epsilon^2 z}{4}+\frac{5\,\epsilon^2}{4}$
-1		$z^4+z^3+(1+\epsilon)\,z^2+\epsilon z+1$
0	$z+2$	$z^3-z^2+3\,z-2$
1	$\frac{4}{\epsilon^2}+\frac{z-2}{\epsilon}$	$z^2\epsilon+(\epsilon-4)\,z+5$
2	$-\frac{\epsilon^2}{4}+\left(-1/8+\frac{z}{16}\right)\epsilon^3+\left(-\frac{11}{1024}+\frac{17\,z}{256}\right)\epsilon^4$	$\left(-\frac{17}{\epsilon}+5+\frac{16}{\epsilon^2}\right)z-\frac{20}{\epsilon^2}-2+\frac{10}{\epsilon}$
3		$\frac{209\,\epsilon^3 z^2}{256}-\frac{99\,\epsilon^3 z}{1024}+\frac{137\,\epsilon^2}{256}-\frac{73\,\epsilon^3}{512}$

For

$$a(z)=z^4+z^3+(1+\epsilon)\,z^2+\epsilon z+1$$
$$b(z)=z^3-z^2+3\,z-2.$$

with tolerances:

1: $\epsilon < tol < 1$,
2: $\epsilon^2 < tol < \epsilon$,
3: $\epsilon^3 < tol < \epsilon^2$