Modular methods for polynomial and matrix arithmetic

Marc Moreno Maza

CS 9652, October 4, 2017

Plan

Plan

Euclidean domains: definition

Definition

An integral domain R endowed with a function $d : R \longmapsto \mathbb{N} \cup \{-\infty\}$ is an *Euclidean domain* if the following two conditions hold

- for all $a, b \in R$ with $a \neq 0$ and $b \neq 0$ we have $d(a\,b) \geq d(a)$,
- for all $a, b \in R$ with $b \neq 0$ there exist $q, r \in R$ such that

$$a = bq + r \quad \text{and} \quad d(r) < d(b). \tag{1}$$

The elements q and r are called the *quotient* and the *remainder* of a w.r.t. b (although q and r may not be unique). The function d is called the *Euclidean size*.

Euclidean domains: examples (1/3)

- $R = \mathbb{Z}$ with $d(a) = |a|$ for $a \in \mathbb{Z}$. Here the quotient q and the remainder r of a w.r.t. b (with $b \neq 0$) can be made unique by requiring $r \geq 0$ (hence we have $0 \leq r < b$).
- $R = \mathbf{k}[x]$ where $\mathbf{k}$ is a field with $d(a) = \deg(a)$ the degree of a for $a \in R, a \neq 0$ and $d(0) = -\infty$. Uniqueness of the quotient and the remainder is easy to show in that case. Indeed

$$a \; = \; b\,q_1 + r_1 \; = \; b\,q_2 + r_2 \;\; \text{with} \;\; \deg(r_1) < \deg(b) \;\; \text{and} \;\; \deg(r_2) < \deg(b) \tag{2}$$

implies

$$r_1 - r_2 \; = \; b\,(q_1 - q_2) \;\; \text{with} \;\; \deg(r_1 - r_2) < \deg(b) \tag{3}$$

Hence we must have $q_1 - q_2 = 0$ and thus $r_1 - r_2 = 0$.
- $R = \mathbf{k}$ is a field with $d(a) = 1$ for $a \in \mathbf{k}, a \neq 0$ and $d(0) = 0$. In this case the quotient q and the remainder r of a w.r.t. b are a/b and 0 respectively.

Euclidean domains: examples (2/3)

- Let R be the ring of the complex numbers whose real and imaginary parts are integer numbers. Hence

$$R = \{x + \imath y \mid x, y \in \mathbb{Z}\} \tag{4}$$

- Consider as a map d from R to $\mathbb{N} \cup \{-\infty\}$ the norm of an element. Hence $d(x + \imath y) = x^2 + y^2$ with $x, y \in \mathbb{Z}$.
- It is easy to check that for every $a, b \in R$ with $a, b \neq 0$ we have $d(a\,b) \geq d(a)$. Indeed for $x, y, z, t \in \mathbb{Z}$ we have

$$\begin{array}{rcl} d((x + \imath y)(z + \imath t)) & = & d(x\ z - t\ y + (y\ z + t\ x)\ i) \\ & = & (x\,z - t\,y)^2 + (y\,z + t\,x)^2 \\ & = & x^2\,z^2 + t^2\,y^2 - 2x\,z\,t\,y + y^2\,z^2 + t^2\,x^2 + 2x\,z\,t\,y \\ & = & x^2\,(z^2 + t^2) + y^2\,(z^2 + t^2) \\ & = & \left(y^2 + x^2\right)\left(z^2 + t^2\right) \\ & = & d(x + iy)\,d(z + it) \end{array} \tag{5}$$

Euclidean domains: examples (3/3)

- Moreover for every $a \neq 0$ we have $d(a) \geq 1$. Therefore we have proved that $d(a\,b) \geq d(a)$ holds for every $a, b \in R$ with $a, b \neq 0$.
- Now given $a, b \in R$ with $b \neq 0$ we are looking for a quotient and a remainder of a w.r.t. b. Hence we are looking for q such that $d(a - b\,q) < d(b)$.
- Such a $q = x + \imath y$ can be constructed as follows. Let q' be such that $a - q'\,b = 0$ that is $q' = a/b = a\overline{b}/d(b)$ where $\overline{b}$ is the conjugate of b. Hence q' writes $x' + \imath y'$ with $x', y' \in \mathbb{Q}$.
- Let $x, y \in \mathbb{Z}$ be such that $\mid x - x' \mid \leq 1/2$ and $\mid y - y' \mid \leq 1/2$. Then

$$\begin{array}{rcl} d(a - b\,q) & = & d(a - b\,q + b\,q' - b\,q') \\ & = & d(b(q' - q)) \\ & = & d(b)\left(\mid x - x' \mid^2 + \mid y - y' \mid^2\right) \\ & \leq & d(b)/2 \\ & < & d(b). \end{array} \qquad (6)$$

- It turns out that several q can be chosen. For instance with $a = 1 + \imath$ and $b = 2 - 2\,\imath$ we have $a - b\,q = -1 - \imath$ with $q = \imath$ and $a - b\,q = 1 + \imath$ with $q = 0$. In both cases $d(a - b\,q) = 2 < 8 = d(b)$.
- Finally this shows that a quotient and a remainder of a w.r.t. b may not be uniquely defined in R

Plan

Preliminary remark

Let R be an Euclidean domain. Let $a, b \in R$ with $b \neq 0$.

Remark

Let r be the remainder of a w.r.t. b. Let $c \in R$. It is easy to see that

$$\left\{ \begin{array}{c|c} c & a \\ c & b \end{array} \right. \iff \left\{ \begin{array}{c|c} c & b \\ c & r \end{array} \right. \tag{7}$$

where $x \mid z$ means that x divides z, that is there exists y such that $xy = z$.

Definition

We say that $g \in R$ is a GCD (greatest common divisor) of a, b whenever the following conditions hold:

1. g divides both a and b,
2. any common divisor of a and b divides g as well.

The algorithm: statement

Input: $a, b \in R$.

Output: $g \in R$ a gcd of a and b.

$r_0 := a$
$r_1 := b$
$i := 2$
while $r_{i-1} \neq 0$ **repeat**
  $r_i := r_{i-2} \operatorname{rem} r_{i-1}$
  $i := i + 1$
return r_{i-2}

The algorithm: proof

Let k be the greatest value of i in the algorithm such that $r_i \neq 0$. From the preliminary remark, we have

$$\begin{cases} c \mid a \\ c \mid b \end{cases} \iff \begin{cases} c \mid b \\ c \mid r_2 \end{cases} \iff \cdots \iff \begin{cases} c \mid r_{k-1} \\ c \mid r_k \end{cases} \iff \begin{cases} c \mid r_k \\ c \mid 0 \end{cases} \tag{8}$$

Hence the following properties hold:

- every divisor of a and b divides r_k,
- r_k divides a and b.

Therefore, the algorithm computes a gcd of a and b.

The algorithm: exmaple

However this gcd may not be the nicest one.

```
(37) -> a:= (4*x-1/2) * (x+2) * (5*x+1) * (1/20*x+1)

           4   883  3   333  2   49
   (37)   x  + --- x  + --- x  + -- x - 1
                40        8       20

(38) -> b := (4*x-1/2) * (x+4) * (5*x-1) * (1/20*x+1)

           4   947  3   2889  2   127
   (38)   x  + --- x  + ---- x  - --- x + 2
                40        40        5

(39) -> r2 := a rem b

            8  3    153  2   557
   (39)   - - x  - --- x  + --- x - 3
            5        5        20

(40) -> r3 := b rem r2

          209  2   33231     209
   (40)   --- x  + ----- x - ---
           80       640       32

(41) -> r4 := r2 rem r3

   (41)   0
```

Normal form of a GCD

Definition

Let R be an Euclidean domain such for every $a \in R$ we can choose a *canonical associate* denoted by $\mathrm{normal}(a)$ and called the *normal form* of a. Because of the polynomial case, the unit u such that $a = u\,\mathrm{normal}(a)$ is denoted $\mathrm{lc}(a)$ and called the *leading coefficient* of a. Then $\mathrm{normal}(r_k)$ where r_k is the result of the EA can be called *the* gcd of a and b.

The algorithm with normal form

```
(2) -> a: P := (4*x-1/2) * (x+2) * (5*x+1) * (1/20*x+1)

         4   883  3   333  2   49
   (2)  x  + --- x  + --- x  + -- x - 1
             40        8       20

(3) -> b: P := (4*x-1/2) * (x+4) * (5*x-1) * (1/20*x+1)

         4   947  3   2889  2   127
   (3)  x  + --- x  + ---- x  - --- x + 2
             40        40        5

(4) -> r2 := unitCanonical(a rem b)

         3   153  2   557     15
   (4)  x  + --- x  - --- x + --
              8        32      8

(5) -> r3 := unitCanonical(b rem r2)

         2   159     5
   (5)  x  + --- x - -
              8      2

(6) -> r4 := unitCanonical(r2 rem r3)

   (6)  0
```

Plan

Bézout coefficients (1/2)

- Let $r_0 = a, r_1 = b, r_2 = r_0 \text{ rem } r_1, \ldots, r_i = r_{i-2} \text{ rem } r_{i-1}, \ldots,$ $\gcd(a, b) = r_k = r_{k-2} \text{ rem } r_{k-1}$ be as before.
- For $i = 2 \cdots k$ let q_i be the quotient of r_{i-2} w.r.t. r_{i-1}, that is,

$$r_{i-2} = q_i \, r_{i-1} + r_i. \tag{9}$$

- Hence we have

$$\begin{array}{rcl} r_2 & = & r_0 - q_2 \, r_1 \\ r_3 & = & r_1 - q_3 \, r_2 \\ \vdots & \vdots & \vdots \\ r_i & = & r_{i-2} - q_i \, r_{i-1} \\ \vdots & \vdots & \vdots \\ r_k & = & r_{k-2} - q_k \, r_{k-1} \end{array} \tag{10}$$

Bézout coefficients (2/2)

Observe that each r_i writes $s_i\,a + t_i\,b$. Indeed we have

$$
\begin{array}{llll}
r_0 = s_0\,a + t_0\,b & \text{with} \quad s_0 = 1 & \text{and} & t_0 = 0 \\
r_1 = s_1\,a + t_1\,b & \text{with} \quad s_1 = 0 & \text{and} & t_1 = 1 \\
r_2 = s_2\,a + t_2\,b & \text{with} \quad s_2 = s_0 - q_2\,s_1 & \text{and} & t_2 = t_0 - q_2\,t_1 \\
r_3 = s_3\,a + t_3\,b & \text{with} \quad s_3 = s_1 - q_3\,s_2 & \text{and} & t_3 = t_1 - q_2\,t_2 \\
\vdots & \quad \vdots \qquad \vdots & \vdots & \vdots \\
r_i = s_i\,a + t_i\,b & \text{with} \quad s_i = s_{i-2} - q_i\,s_{i-1} & \text{and} & t_i = t_{i-2} - q_i\,t_{i-1} \\
\vdots & \quad \vdots \qquad \vdots & \vdots & \vdots \\
r_k = s_k\,a + t_k\,b & \text{with} \quad s_k = s_{k-2} - q_k\,s_{k-1} & \text{and} & t_k = t_{k-2} - q_k\,t_{k-1}
\end{array}
\tag{11}
$$

- The elements s_k and t_k are called the *Bézout coefficients* of $\gcd(a, b)$.
- In order to compute **a gcd** together with its Bézout coefficients one needs to enhance the previous algorithm into the so-called *Extended Euclidean Algorithm (EEA)*.

The extended algorithm

Input: $a, b \in R$.

Output: $g \in R$ a gcd of a and b together with $s, t \in R$ such that $g = s\,a + t\,b$.

$r_0 := a$; $s_0 := 1$; $t_0 := 0$
$r_1 := b$; $s_1 := 0$; $t_1 := 1$
$i := 2$
while $r_{i-1} \neq 0$ **repeat**
  $q_i := r_{i-2} \text{ quo } r_{i-1}$
  $r_i := r_{i-2} \text{ rem } r_{i-1}$
  $s_i := s_{i-2} - q_i\, s_{i-1}$
  $t_i := t_{i-2} - q_i\, t_{i-1}$
  $i := i + 1$
return$(r_{i-2}, s_{i-2}, t_{i-2})$

The extended algorithm with normalization

Input: $a, b \in R$.

Output: $g \in R$ the gcd of a and b together with $s, t \in R$ such that $g = s\,a + t\,b$.

$u_0 := \mathrm{lc}(a)$; $r_0 := \mathrm{normal}(a)$; $s_0 := u_0^{-1}$; $t_0 := 0$
$u_1 := \mathrm{lc}(b)$; $r_1 := \mathrm{normal}(b)$; $s_1 := 0$; $t_1 := u_1^{-1}$
$i := 2$
while $r_{i-1} \neq 0$ **repeat**
  $q_i := r_{i-2} \text{ quo } r_{i-1}$
  $r_i := r_{i-2} \text{ rem } r_{i-1}$
  $u_i := \mathrm{lc}(r_i)$
  $r_i := \mathrm{normal}(r_i)$
  $s_i := (s_{i-2} - q_i\, s_{i-1})/u_i$
  $t_i := (t_{i-2} - q_i\, t_{i-1})/u_i$
  $i := i + 1$
return$(r_{i-2}, s_{i-2}, t_{i-2})$

EEA: analyis (1/5)

In order to analyze the extended algorithms, we introduce the following matrices

$$R_0 = \begin{pmatrix} s_0 & t_0 \\ s_1 & t_1 \end{pmatrix} \text{ and } Q_i = \begin{pmatrix} 0 & 1 \\ u_{i+1}^{-1} & -q_{i+1} u_{i+1}^{-1} \end{pmatrix} \text{ for } 1 \leq i \leq k \quad (12)$$

with coefficients in R. Then, we define

$$R_i = Q_i \cdots Q_1 R_0 \text{ for } 1 \leq i \leq k. \quad (13)$$

The following proposition collects some invariants of the Extended Euclidean Algorithm.

EEA: analyis (2/5)

Proposition

With the convention that $r_{k+1} = 0$ and $u_{k+1} = 1$, for $0 \le i \le k$ we have

(*i*) $R_i \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} r_i \\ r_{i+1} \end{pmatrix}$,

(*ii*) $R_i = \begin{pmatrix} s_i & t_i \\ s_{i+1} & t_{i+1} \end{pmatrix}$,

(*iii*) $\gcd(a, b) = \gcd(r_i, r_{i+1}) = r_k$,

(*iv*) $s_i a + t_i b = r_i$ *and* $s_{k+1} a + t_{k+1} b = 0$,

(*v*) $s_i t_{i+1} - t_i s_{i+1} = (-1)^i (u_0 \cdots u_{i+1})^{-1}$,

(*vi*) $\gcd(s_i, t_i) = 1$,

(*vii*) $\gcd(r_i, t_i) = \gcd(a, t_i)$,

(*viii*) *the matrices* R_i *and* Q_i *are invertible;* $Q_i^{-1} = \begin{pmatrix} q_{i+1} & u_{i+1} \\ 1 & 0 \end{pmatrix}$ *and*

$$R_i^{-1} = (-1)^i (u_0 \cdots u_{i+1}) \begin{pmatrix} t_{i+1} & -t_i \\ -s_{i+1} & s_i \end{pmatrix},$$

(*ix*) $a = (-1)^i (u_0 \cdots u_{i+1})(t_{i+1} r_i - t_i r_{i+1})$,

(*x*) $b = (-1)^{i+1} (u_0 \cdots u_{i+1})(s_{i+1} r_i - s_i r_{i+1})$.

EEA: analyis (3/5)

Proof (1/3)

We prove (i) and (ii) by induction on i. The case $i = 0$ follows immediately from the definitions of s_0, r_0, s_1, r_1 and R_0. We assume that (i) and (ii) hold for $0 \leq i < k$. By induction hypothesis, we have

$$\begin{array}{rcl} R_{i+1}\left(\begin{array}{c} a \\ b \end{array}\right) & = & Q_{i+1} R_i \left(\begin{array}{c} a \\ b \end{array}\right) \\ & = & Q_{i+1} \left(\begin{array}{c} r_i \\ r_{i+1} \end{array}\right) \\ & = & \left(\begin{array}{cc} 0 & 1 \\ u_{i+2}^{-1} & -q_{i+2} u_{i+2}^{-1} \end{array}\right) \left(\begin{array}{c} r_i \\ r_{i+1} \end{array}\right) \\ & = & \left(\begin{array}{c} r_{i+1} \\ u_{i+2}^{-1}(r_i - q_{i+2} r_{i+1}) \end{array}\right) \\ & = & \left(\begin{array}{c} r_{i+1} \\ r_{i+2} \end{array}\right). \end{array} \quad (14)$$

EEA: analyis (4/5)

Proof (2/3)

Similarly, we have

$$\begin{array}{rcl} R_{i+1} & = & Q_{i+1} R_i \\ & = & Q_{i+1} \left(\begin{array}{cc} s_i & t_i \\ s_{i+1} & t_{i+1} \end{array} \right) \\ & = & \left(\begin{array}{cc} 0 & 1 \\ u_{i+2}^{-1} & -q_{i+2} u_{i+2}^{-1} \end{array} \right) \left(\begin{array}{cc} s_i & t_i \\ s_{i+1} & t_{i+1} \end{array} \right) \\ & = & \left(\begin{array}{cc} s_{i+1} & t_{i+1} \\ s_{i+2} & t_{i+2} \end{array} \right). \end{array} \quad (15)$$

Property (iii) follows.
Claim (iv) follows from (i) and (ii).
Taking the determinant of each side of (ii) we prove (v) as follows:

$$\begin{array}{rcl} s_i t_{i+1} - t_i s_{i+1} & = & \det \left(\begin{array}{cc} s_i & t_i \\ s_{i+1} & t_{i+1} \end{array} \right) \\ & = & \det R_i \\ & = & \det Q_i \cdots \det Q_1 \det \left(\begin{array}{cc} s_0 & t_0 \\ s_1 & t_1 \end{array} \right) \\ & = & (-1)^i (u_{i+1} \cdots u_2)^{-1} (u_0^{-1} u_1^{-1} - 0). \end{array} \quad (16)$$

EEA: analyis (5/5)

Proof (3/3)

- Now, we prove (vi). If s_i and t_i would have a non-invertible common factor, then it would divide $s_i t_{i+1} - t_i s_{i+1}$. This contradicts (v) and proves (vi).
- We prove (vii). Let $p \in R$ be a divisor of t_i. If $p \mid a$, then $p \mid r_i$ holds since we have $r_i = s_i a + t_i b$ from (i). If $p \mid r_i$, then $p \mid s_i a$ and, thus, $p \mid a$ since t_i and s_i are relatively prime, from (vi).
- We prove $(viii)$. From (v), we deduce that Q_i is invertible. Then, the invertibility of R_i follows easily from that of Q_i. It is routine to check that the proposed inverses are correct.
- Finally, claims (ix) and (x) are derived from (i) by multiplying each side with the inverse of R_i given in $(viii)$.

Remark

When $R = \mathbf{k}[x]$ and $\mathbf{k}$ is a field, the following proposition shows that the degrees of the Bézout coefficients of the EEA grow *linearly*. The second following proposition shows that the Bézout coefficients are essentially unique, provided that their degrees are small enough.

EEA: case of $R = \mathbf{k}[x]$ (1/7)

Proposition

With the same notations as in the previous proposition, we assume that $R = \mathbf{k}[x]$ where $\mathbf{k}$ is a field. Then, for $2 \le i \le k+1$, we have

$$\deg(s_i) = \sum_{2 \le j < i} \deg(q_j) = n_1 - n_{i-1} \tag{17}$$

and, for $1 \le i \le k+1$, we have

$$\deg(t_i) = \sum_{1 \le j < i} \deg(q_j) = n_0 - n_{i-1} \tag{18}$$

where $n_i = \deg r_i$ for $0 \le i \le k$.

EEA: case of $R = \mathbf{k}[x]$ (2/7)

Proof (1/2)

We only prove the first equality since the second one can be verified in a similar way. In fact, we prove this first equality together with

$$\deg(s_{i-1}) < \deg(s_i) \tag{19}$$

by induction on $2 \leq i \leq k+1$. For $i = 2$, the first equality holds since we have

$$\deg(s_i) = \deg(s_0 - q_2 s_1) = \deg(1 - 0\, q_2) = 0 = n_1 - n_{i-1} \tag{20}$$

and the inequality holds since we have

$$-\infty = \deg(s_1) < \deg(s_2) = 0. \tag{21}$$

EEA: case of $R = \mathbf{k}[x]$ (3/7)

Proof (2/2)

Now we consider $i \geq 2$ and we assume that both properties hold for $2 \leq j \leq i$. Then, by induction hypothesis, we have

$$\deg(s_{i-1}) < \deg(s_i) < n_{i-1} - n_i + \deg(s_i) = \deg(q_{i+1}) + \deg(s_i) = \deg(q_{i+1} s_i) \quad (22)$$

which implies

$$\deg(s_{i+1}) = \deg(s_{i-1} - q_{i+1} s_i) = \deg(q_{i+1} s_i) > \deg(s_i) \quad (23)$$

and

$$\deg(s_{i+1}) = \deg(q_{i+1}) + \deg(s_i) = \deg(q_{i+1}) + \sum_{2 \leq j \leq i} \deg(q_j) = \sum_{2 \leq j \leq i+1} \deg(q_j) \quad (24)$$

where we used the induction hypothesis also.

EEA: case of $R = \mathbf{k}[x]$ (4/7)

Proposition

With the same notations as in the previous proposition, we assume that $R = \mathbf{k}[x]$ where $\mathbf{k}$ is a field. We recall $n = \deg a$. Let $r, s, t \in \mathbf{k}[x]$, with $t \neq 0$, be polynomials such that

$$r = sa + tb \quad \text{and} \quad \deg r + \deg t < \deg a. \tag{25}$$

Let $j \in \{1, \ldots, k+1\}$ be such that

$$\deg r_j \leq \deg r < \deg r_{j-1}. \tag{26}$$

Then, there exists a non-zero $\alpha \in \mathbf{k}[x]$ such that we have

$$r = \alpha r_j, s = \alpha s_j \quad \text{and} \quad t = \alpha t_j. \tag{27}$$

EEA: case of $R = \mathbf{k}[x]$ (5/7)

Proof (1/3)

First, we observe that the index j exists and is unique. Indeed, we have $-\infty < \deg r < n$ and,

$$-\infty = \deg r_{k+1} < \deg r_k < \cdots < \deg r_{j+1} < \deg r_j < \cdots < \deg a = n. \qquad (28)$$

Second, we claim that

$$s_j t = s t_j \qquad (29)$$

holds. Suppose that the claim is false and consider the following linear system over R with $\begin{pmatrix} f \\ g \end{pmatrix}$ as unknown:

$$\begin{pmatrix} s_j & t_j \\ s & t \end{pmatrix} \begin{pmatrix} f \\ g \end{pmatrix} = \begin{pmatrix} r_j \\ r \end{pmatrix} \qquad (30)$$

EEA: case of $R = \mathbf{k}[x]$ (6/7)

Proof (2/3)

Since the matrix of this linear system is non-singular, we can solve for f over the field of fractions of R. Moreover, we know that $\begin{pmatrix} f \\ g \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}$ is the solution. Hence, using Cramer's rule we obtain:

$$a = \frac{\det \begin{pmatrix} r_j & t_j \\ r & t \end{pmatrix}}{\det \begin{pmatrix} s_j & t_j \\ s & t \end{pmatrix}}. \tag{31}$$

The degree of the left hand side is n while the degree of the right hand side is equal or less than:

$$\begin{array}{rcl} \deg(r_j t - r t_j) & \leq & \max(\deg r_j + \deg t, \deg r + \deg t_j) \\ & \leq & \max(\deg r + \deg t, \deg r + n - \deg r_{j-1}) \\ & < & \max(n, \deg r_{j-1} + n - \deg r_{j-1}) = n. \end{array} \tag{32}$$

by virtue of the definition of j, Relation (25) and the previous proposition. This leads to a contradiction.

EEA: case of $R = \mathbf{k}[x]$ (7/7)

Proof (3/3)

Hence, we have $s_j t = s t_j$. This implies that t_j divides $t s_j$. Since s_j and t_j are relatively prime (Point (vi) of the second previous proposition we deduce that t_j divides t. So let $\alpha \in \mathbf{k}[x]$ such that we have

$$t = \alpha t_j. \qquad (33)$$

Hence we obtain $s_j \alpha t_j = s t_j$. Since $t \neq 0$ holds, we have $t_j \neq 0$, leading to

$$s = s_j \alpha. \qquad (34)$$

Finally, plugging Equation (33) and Equation (34) in Equation (25), we obtain $r = \alpha r_j$, as claimed.

EA and EEA: complexity estimates

Proposition

Let $a, b \in \mathbf{k}[x]$ where $\mathbf{k}$ is a field. Assume $\deg(a) = n \geq \deg(b) = m$.

- *the EEA requires at most $m + 2$ inversions and $13/2m\,n + \mathcal{O}(n)$ additions and multiplications in $\mathbf{k}$.*
- *If we do not compute the coefficients s_i, t_i then EEA requires at most $m + 2$ inversions and $5/2m\,n + \mathcal{O}(n)$ additions and multiplications in $\mathbf{k}$.*

Proposition

Let $a, b \in \mathbb{Z}$ be multi-precision integers written with m and m words. Then, the EEA can be performed within $\mathcal{O}(m\,n)$ word operations.

Plan

Evaluation

Notatins

Let $\mathbf{k}$ be a field and let $u = (u_0, \ldots, u_{n-1})$ be a sequence of pairwise distinct elements of $\mathbf{k}$.

Horner's rule

- A polynomial in $\mathbf{k}[x]$ with degree $n - 1$, say

$$f = f_0 + f_1 x + \cdots + f_{n-1} x^{n-1} \tag{35}$$

  can be evaluated at $x = x_0$ using *Horner's rule*

$$f(x_0) = (\cdots(f_{n-1} x_0 + f_{n-2}) x_0 + \cdots + f_1) x_0 + f_0 \tag{36}$$

  with $n - 1$ additions and $n - 1$ multiplications leading to $2n - 2$ operations in the base field $\mathbf{k}$.
- The proof is easy by induction on $n \geq 1$.

Lagrange interpolant (1/2)

Definition

For $i = 0 \cdots n-1$ the i-th *Lagrange interpolant* is the polynomial

$$L_i(u, x) = \prod_{\substack{0 \leq j < n \\ j \neq i}} \frac{x - u_j}{u_i - u_j} \tag{37}$$

with the property that

$$L_i(u, u_j) = \begin{cases} 0 & \text{if} \quad i \neq j \\ 1 & \text{otherwise} \end{cases} \tag{38}$$

Lagrange interpolant (2/2)

Proposition

Let $v_0, \ldots, v_{n-1}$ *be in* $\mathbf{k}$*. There is a* **unique** *polynomial* $f \in \mathbf{k}[x]$ *with degree less than* n *and such that*

$$f(u_i) = v_i \quad \text{for} \quad i = 0 \cdots n-1. \tag{39}$$

Moreover this polynomial is given by

$$f(x) \;=\; \sum_{0 \leq i < n} v_i \; L_i(u, x). \tag{40}$$

Proof

Clearly the polynomial f of Relation (40) satisfies Relation (39). Hence the existence is clear. The unicity follows from the fact that the difference of two such polynomials has

- degree less than n and,
- n roots.

Hence is the zero polynomial.

Lagrange interpolation: complexity estimates (1/3)

Proposition

Evaluating a polynomial $f \in \mathbf{k}[x]$ of degree less than n at n distinct points $u_0, \ldots, u_{n-1}$ or computing an interpolating polynomial at these points can be done in $\mathcal{O}(n^2)$ operations in $\mathbf{k}$.

Proof (1/3)

- We saw that evaluating the polynomial f of degree $n-1$ at one point costs $2n-2$ operations in $\mathbf{k}$. So evaluating f at $u_0, \ldots, u_{n-1}$ amounts to $2n^2 - 2n$. Let us prove now that interpolating a polynomial at $u_0, \ldots, u_{n-1}$ can be done in $\mathcal{O}(n^2)$ operations in $\mathbf{k}$.
- We first need to estimate the cost of computing the i-th interpolant $L_i(u, x)$. Consider $m_0 m_1$, $m_0 m_1 m_2$, $\ldots$ $m = m_0 \cdots m_{n-1}$ where m_i is the monic polynomial $m_i = x - u_i$. Let $p_i = m_0 m_1 \cdots m_{i-1}$ and $q_i = m/m_i$ for $i = 1 \cdots n$. We have

$$L_i(u, x) = \frac{q_i(x)}{q_i(u_i)} \tag{41}$$

Lagrange interpolation: complexity estimates (2/3)

Proof (2/3)

To estimate the cost of computing the $L_i(u,x)$'s let us start with that of m. Computing the product of the monic polynomial $p_i = m_0 m_1 \cdots m_{i-1}$ of degree i by the monic polynomial $m_i = x - u_i$ of degree 1 costs

- i multiplications (in the field $\mathbf{k}$) to get $-u_i\, p_i$ plus
- i additions (in the field $\mathbf{k}$) to add $-u_i\, p_i$ (of degree i) to $x\, p_i$ (of degree $i+1$ but without constant term)

leading to $2\,i$. Hence computing $p_2, \ldots, p_n = m$ amounts to

$$\begin{array}{rcl} \Sigma_{1 \leq i \leq n-1}\, 2\, i & = & 2\Sigma_{1 \leq i \leq n-1}\, i \\ & = & 2\, n\, (n-1)/2 \\ & = & n(n-1). \end{array} \tag{42}$$

Lagrange interpolation: complexity estimates (3/3)

Proof (3/3)

- Computing q_i implies a *division-with-remainder* of the polynomial m of degree n by the polynomial m_i of degree 1. This division will have $n - 1 + 1$ steps, each step requiring 2 operations in $\mathbf{k}$. Hence computing all q_i's amounts to $2n^2$.
- Since q_i has degree $n - 1$ computing each $q_i(u_i)$'s amounts to $2n - 2$ operations in the base field $\mathbf{k}$ Then computing all $q_i(u_i)$'s amounts to $2n^2 - 2n$. Then computing each $L_i(u, x) = q_i(x)/q_i(u_i)$ from the q_i's and $q_i(u_i)$'s costs n. Therefore computing all $L_i(u)$'s from scratch amounts to $n(n + 1) + 2\,n^2 + 2n^2 - 2n + n^2 = 6n^2 - n$.
- Computing f from the $L_i(u)$'s requires
  - to multiply each $L_i(u, x)$ (which is a polynomial of degree $n - 1$) by the number v_i leading to n^2 operations in $\mathbf{k}$ and
  - to add these $v_i\,L_i(u, x)$ leading to $n - 1$ additions of polynomials of degree at most $n - 1$ costing $(n - 1)n$ operations in $\mathbf{k}$

  amounting to $2\,n^2 - n$.
- Finally the total cost is $6n^2 - 2n - 1 + 2\,n^2 - n = 8\,n^2 - 2\,n$.

Vandermonde matrix

We consider the map

$$E: \begin{array}{ccc} \mathbf{k}^n & \to & \mathbf{k}^n \\ (f_0, \ldots, f_{n-1}) & \longmapsto & (\Sigma_{0 \le j < n} f_j u_0^j, \ldots, \Sigma_{0 \le j < n} f_j u_{n-1}^j) \end{array} \tag{43}$$

This is just the map corresponding to evaluation of polynomials of degree less than n at points $u_0, \ldots, u_{n-1}$. It is obvious that E is $\mathbf{k}$-linear and it can be represented by the *Vandermonde matrix*

$$VDM(u_0, \ldots, u_{n-1}) = \begin{pmatrix} 1 & u_0 & u_0^2 & \cdots & u_0^{n-1} \\ 1 & u_1 & u_1^2 & \cdots & u_1^{n-1} \\ 1 & u_2 & u_2^2 & \cdots & u_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & u_{n-1} & u_{n-1}^2 & \cdots & u_{n-1}^{n-1} \end{pmatrix} \tag{44}$$

From the above discussion, this matrix is invertible iff $u_i \neq u_j$ for all $0 \le i < j \le n-1$. To conclude observe that both *evaluation* and *interpolation* are linear maps between coefficients and value vectors.

Plan

Modular addition and mutiplication

Let R be a (commutative) ring (with unity) and I be an ideal of R. For $a, b \in R$ the relation

$$a - b \in I \tag{45}$$

usually denoted by

$$a \equiv b \mod I \tag{46}$$

defines an equivalence relation. If we denote by $\overline{a}^I$ (or $\overline{a}$ if not ambiguous) the class of the element a, then the residue classes of this relation forms a (commutative) ring (with unity) denoted by R/I where addition and multiplication are defined by

$$\overline{a} + \overline{b} = \overline{a + b} \quad \text{and} \quad \overline{a}\overline{b} = \overline{ab}. \tag{47}$$

Modular computation in an Euclidean domain

- Let R be an Euclidean domain and let $p \in R$ with $p \neq 0$. We consider the ideal I generated by p.
- The residue class ring R/I is often denoted by R/p and the class of $a \in R$ in R/p by $a \mod p$.
- For $a, b \in R$ the relation $a - b \in I$ means that $a - b$ is a multiple of p.
- Let (q_a, r_a) and (q_b, r_b) be the quotient-remainder pairs of a and b w.r.t. p respectively. Then, we have

$$p \mid a - b \iff p \mid (q_a - q_b)p + r_a - r_b \iff p \mid r_a - r_b. \quad (48)$$

- For $R = \mathbb{Z}$ with positive remainder or for $R = \mathbf{k}[x]$ we have in fact

$$a \equiv b \mod p \iff r_a = r_b. \quad (49)$$

- Explain why!

Modular computation in $R = \mathbf{k}[x]$

Let $R = \mathbf{k}[x]$ for a field $\mathbf{k}$. Let $u \in \mathbf{k}$ and let I be the ideal generated by the polynomial $p := x - u$. For every $a \in R$ there exists $q \in R$ such that

$$a = q(x-u) + a(u) \tag{50}$$

So in that case for every $a, b \in R$ we have

$$a \equiv b \mod p \iff a(u) = b(u) \tag{51}$$

Modular inversion

Proposition

Let R be an Euclidean domain and let a, m be in R. Then $a \mod m$ is a unit of R/m iff $\gcd(a, m) = 1$. In this case the Extended Euclidean Algorithm can be used to compute the inverse of $a \mod m$.

Proof

Indeed, let g be the gcd of a and m and let s, t be the corresponding Bézout coefficients. Hence we have

$$s\,a + t\,m = g \tag{52}$$

If $g = 1$ then $s \mod m$ is the inverse of $a \mod m$. Conversely if $a \mod m$ is invertible there exists $b \in R$ such that

$$a\,b \equiv 1 \mod m \tag{53}$$

That is, there exists $c \in R$ such that $a\,b - 1 = c\,m$. If a and m could be divided by an element d which is not a unit then we would be led to a contradiction $(d(a'\,b + c\,m') = 1)$. Hence $\gcd(a, m) = 1$.

Modular computations in $\mathbf{k}[x]$: complexity estimates

Proposition

Let $\mathbf{k}$ be a field and $f \in \mathbf{k}[x]$ with degree $n \in \mathbb{N}$. One arithmetic operation in the residue class ring $\mathbf{k}[x]/f$, that is, addition, multiplication or division by an invertible element can be done using $\mathcal{O}(n^2)$ arithmetic operations in $\mathbf{k}$.

Explain why!

Proof

Euler's totient function

Let m be a positive integer. The set

$$(\mathbb{Z}/m)^* \;=\; \{a \mod m \mid \gcd(a, m) = 1\} \tag{54}$$

is the group of units of the ring $\mathbb{Z}/m$. The Euler's totient function $m \longmapsto \phi(m)$ counts the number of elements of $(\mathbb{Z}/m)^*$. By convention $\phi(1) = 1$. Then, if p is a prime, we have $\phi(p) = p - 1$. If m is a power p^e of the prime p, then we have

$$\phi(m) \;=\; (p-1)p^{e-1} \tag{55}$$

Explain why!

Fermat's little theorem (1/2)

Proposition

If $p \in \mathbb{N}$ is a prime and $a \in \mathbb{Z}$ then we have

$$a^p \equiv a \mod p \tag{56}$$

Moreover if p does not divide a then we have $a^{p-1} \equiv 1 \mod p$.

Proof

It is sufficient to prove the claim for $a = 0 \cdots p-1$ which we do by induction on a. The cases $a = 0$ and $a = 1$ are trivial. For $a > 1$ we have

$$a^p = ((a-1)+1)^p \equiv (a-1)^p + 1^p \equiv (a-1) + 1 = a \mod p \tag{57}$$

Fermat's little theorem (2/2)

For a prime $p \in \mathbb{N}$ and $a \in \mathbb{Z}$ such that $a \neq 0$ and such that p does not divide a. It follows from Fermat's little theorem that the inverse of a mod p can be computed by

$$a^{-1} \equiv a^{p-2} \mod p \tag{58}$$

Explain why this leads to an algorithm requiring $\mathcal{O}(\log_2^3(p))$ word operations. Is this better than the modular inversion via Euclide's algorithm?

Plan

CRT (Sun-Tsu, first century AD)

Proposition

Let m and n be two relatively prime integers. Let $s, t \in \mathbb{Z}$ be such that $s\,m + t\,n = 1$. For every $a, b \in \mathbb{Z}$ there exists $c \in \mathbb{Z}$ such that

$$(\forall x \in \mathbb{Z}) \qquad \begin{cases} x \equiv a \mod m \\ x \equiv b \mod n \end{cases} \iff x \equiv c \mod m\,n \tag{59}$$

where a convenient c is given by

$$c \;=\; a + (b - a)\,s\,m = b + (a - b)\,t\,n \tag{60}$$

Therefore for every $a, b \in \mathbb{Z}$ the system of equations

$$\begin{cases} x \equiv a \mod m \\ x \equiv b \mod n \end{cases} \tag{61}$$

has a solution.

CRT: Proof

First observe that Relation (60) implies

$$c \equiv a \mod m \qquad \text{and} \qquad c \equiv b \mod n. \tag{62}$$

Now assume that $x \equiv c \mod m\,n$ holds. This implies

$$x \equiv c \mod m \qquad \text{and} \qquad x \equiv c \mod n \tag{63}$$

Thus Relations (62) and (63) lead to

$$x \equiv a \mod m \qquad \text{and} \qquad x \equiv b \mod n \tag{64}$$

Conversly

- $x \equiv a \mod m$ implies $x \equiv c \mod m$, that is, m divides $x - c$ and
- $x \equiv b \mod n$ implies $x \equiv c \mod n$, that is, n divides $x - c$.

Since m and n are relatively prime it follows that $m\,n$ divides $x - c$.
(Gauss Lemma).

CRT: a more modern version

Proposition

Let $m_0, \ldots, m_{n-1}$ *be* n *elements pairwise coprime in the Euclidean domain* R. *(Hence, for all* $0 \leq i < j < n$ *we have* $\gcd(m_i, m_j) = 1$.*) Let* $m = m_0 \cdots m_{r-1}$. *Then, we have the ring isomorphism*

$$R/m \;\simeq\; R/m_0 \times \cdots \times R/m_{n-1} \tag{65}$$

and the group isomorphism of the multiplicative groups

$$(R/m)^* \;\simeq\; (R/m_0)^* \times \cdots \times (R/m_{n-1})^* \tag{66}$$

Proof

Read the *Chinese Remaindering Algorithm* (long) section in
`http://www.csd.uwo.ca/~moreno//CS424/Lectures/EuclideanMethods.html/index.html`

Chinese Remaindering Algorithm (CRA)

Input: $m_0, \ldots, m_{n-1} \in R$ pairwise coprime and
$r_0, \ldots r_{n-1} \in R$.

Output: $r \in R$ such that $r \equiv r_i \mod m_i$ for $i = 0 \cdots n-1$.

$m := m_0 \cdots m_{n-1}$.
$r := 0$
for $i = 0 \cdots n-1$ **repeat**
  $(u_i, v_i, g_i) :=$ extendedEuclidean$(m_i, \frac{m}{m_i})$
  if $g_i \neq 1$ then **error**
  $c_i := r_i v_i \operatorname{rem} m_i$
  $r := r + c_i \frac{m}{m_i}$
return r

Proof of CRA

Proof

Assume that the algorithm terminates without error, which is the case if every g_i is **the** gcd of m_i and $\frac{m}{m_i}$ (which are assumed to be coprime). Then, for $i = 0 \cdots n-1$ we have

$$u_i \, m_i + v_i \frac{m}{m_i} \;=\; 1 \tag{67}$$

Hence

$$r_i v_i \frac{m}{m_i} \;\equiv\; r_i \mod m_i \tag{68}$$

and for $j = 0 \cdots n-1$ with $j \neq i$ we have

$$r_i v_i \frac{m}{m_i} \;\equiv\; 0 \mod m_j \tag{69}$$

The conclusion follows easily from Relation (68) and (69).

Remark about CRA

It is important to observe that the CRA computes a solution r of the system of equations given by

$$r \equiv r_i \mod m_i \quad \text{for } i = 0 \cdots n-1 \tag{70}$$

Any other solution r' of (70) satisfies $r \equiv r' \mod m$ where m is the product of the moduli $m_0, \ldots, m_{r-1}$. This follows from the fact the m_i's are pairwise coprime.
Therefore the set all solutions of (70) is of the form

$$\{r + k\, m \mid k \in R\} \tag{71}$$

However, in practice, we need only one solution. In the next two results by imposing

$$d(r) < d(m) \tag{72}$$

where d is the Euclidean size of R, we manage to restrict to a unique solution.

CRA in $R = \mathbf{k}[x]$ (1/2)

Proposition

- *Let* $R = \mathbf{k}[x]$ *for a field* $\mathbf{k}$.
- *Let* $m_0, \ldots, m_{r-1} \in R$ *be polynomials pairwise coprime* ($\gcd(m_i, m_j) = 1$ *for* $0 \le i < j \le r-1$).
- *Let* m *be their product.*
- *For* $0 \le i \le r-1$ *let* $d_i \ge 1$ *be the degree of* m_i *and* $n = \Sigma_{i=0}^{r-1} d_i$ *be the degree of* m.
- *For* $0 \le i \le r-1$ *let* $f_i \in \mathbf{k}[x]$ *be a polynomial with degree* $\deg(f_i) < d_i$.

Then, there is a unique polynomial $f \in \mathbf{k}[x]$ *such that*

$$\deg(f) < n \quad \text{and} \quad f \equiv f_i \mod m_i \ \text{ for } \ i = 0 \cdots r-1. \tag{73}$$

Moreover it can be computed in $\mathcal{O}(n^2)$ *operations in* $\mathbf{k}$.

CRA in $R = \mathbf{k}[x]$ (2/2)

Proof

Except for the complexity result (which can be found in *Modern Computer Algebra*) and the uniqueness, this theorem follows from previous discussions. The uniqueness follows from the constraint $\deg(f) < n$. Indeed, assume that there are two polynomials f and g solutions of (73). Then we have

$$f \equiv g \mod m_i \quad \text{for} \quad i = 0 \cdots r - 1. \tag{74}$$

and thus

$$f \equiv g \mod m \tag{75}$$

Hence m divides $f - g$ although $\deg(m) = n > \deg(f - g)$ holds. Therefore $f = g$.

CRA in $R = \mathbb{Z}$

Proposition

- *Let* $m_0, \ldots, m_{r-1}, m$ *be in* $R = \mathbb{Z}$ *such that the* m_i*'s are pairwise coprime and* m *is their product.*
- *Let* n *be the word length of* m*.*
- *Let* $a_0, \ldots, a_{r-1} \in R$ *be such that* $0 \leq a_i < m_i$ *for* $i = 0 \cdots r-1$*.*

Then there is a unique $a \in R$ *such that*

$$0 \leq a < m \quad \text{and} \quad a \equiv a_i \mod m_i \;\; \text{for} \;\; i = 0 \cdots r-1. \tag{76}$$

Moreover it can be computed in $\mathcal{O}(n^2)$ *word operations.*

Except for the complexity result (which can be found in *Modern Computer Algebra*) and the uniqueness, this theorem follows from previous discussions. The proof of the uniqueness is quite easy to establish.

Plan

Rational function reconstruction

Read the *Rational Function Reconstruction* (technical) section in `http://www.csd.uwo.ca/~moreno//CS424/Lectures/EuclideanMethods.html/index.html`

Plan

Introduction (1/2)

Consider a square matrix A of order n with coefficients in $\mathbb{Z}$. It is known that $\det(A)$, the determinant of A, can be computed in at most $2n^3$ operations in $\mathbb{Q}$ by means of Gaussian elimination. Let us estimate the growth of the coefficients. For simplicity, assume

- A is not singular,
- no row or column permutations are necessary,

After $k-1$ pivoting stages the current matrxi $A^{(k-1)}$ looks like

$$\begin{pmatrix} * & & & & & & & \\ 0 & * & & & & & & \\ & & & & & & & \\ & & 0 & * & & & & \\ & & & 0 & a_{kk}^{(k)} & \cdots & a_{kj}^{(k)} & \cdots \\ & & & \vdots & \vdots & & \vdots & \\ & & & 0 & a_{ik}^{(k)} & \cdots & a_{ij}^{(k)} & \cdots \\ & & & \vdots & \vdots & & \vdots & \end{pmatrix} \tag{77}$$

The entries of the matrix for $k < i \leq n$ and $k \leq j \leq n$ change according to the formula

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}} a_{kj}^{(k)}. \tag{78}$$

Introduction (2/2)

We consider the following numbers.

- Let b_k be an upper bound for the absolute value of the numerators and the denominators of all $a_{ij}^{(k)}$.
- In particular for $1 \leq i, j \leq n$ we have $\mid a_{ij} \mid \leq \ b_0$.

From Relation (78) we obtain

$$b_k \ \leq \ 2 b_{k-1}^4 \ \leq \ 4 b_{k-2}^{4^2} \ \leq \ \cdots \ \leq \ 2^k {b_0}^{4^k} \tag{79}$$

This shows an exponential upper bound. (However a polynomial bound in b_0, n can be established but the proof is far from trivial and the formula still not be very encouraging.)
In what follows, we present an approach whose goal is to control the growth of the intermediate computations when calculating the determinant of A.

Preliminaries (1/3)

Let d be this determinant. Let us choose a prime number $p \in \mathbb{Z}$ such that

$$p \ > \ 2 \, | \, d \, | \tag{80}$$

Let r be the determinant of A regarded as a matrix over $\mathbb{Z}/p\mathbb{Z}$ and let us choose for representing $\mathbb{Z}/p\mathbb{Z}$ the integers in the symmetric range

$$-\frac{p-1}{2} \cdots \frac{p-1}{2} \tag{81}$$

Hence we have

$$-\frac{p}{2} < r < \frac{p}{2} \quad \text{and} \quad -\frac{p}{2} < d < \frac{p}{2} \tag{82}$$

leading to

$$-p < d - r < p \tag{83}$$

Observe that $\det(A)$ is a polynomial in the coefficients of A. For instance with $n = 2$ we have

$$\det(A) \ = \ a_{11} \, a_{22} - a_{12} \, a_{21} \tag{84}$$

which shows that $\det(A)$ (for $n = 2$) is a polynomial in a_{11}, a_{22}, a_{12}, a_{21}.

Preliminaries (2/3)

Observe also the map

$$h: \begin{array}{ccc} \mathbb{Z} & \longrightarrow & \mathbb{Z}/p\mathbb{Z} \\ x & \longrightarrow & x \mod p = \overline{x}^p \end{array} \quad (85)$$

is a ring homomorphism. In other words for every $x, y \in \mathbb{Z}$ we have

$$\overline{x+y}^p = \overline{x}^p + \overline{y}^p \quad \text{and} \quad \overline{xy}^p = \overline{x}^p \overline{y}^p \quad (86)$$

Hence for $n = 2$ we have

$$\overline{\det(A)}^p = \overline{a_{11}}^p \, \overline{a_{22}}^p - \overline{a_{12}}^p \, \overline{a_{21}}^p \quad (87)$$

More generally we have

$$\overline{\det(A)}^p = \det(A \mod p) \quad (88)$$

that is

$$d \equiv r \mod p \quad (89)$$

which means that p divides $d - r$. This with Relation (83) leads to

$$d = r \quad (90)$$

Preliminaries (3/3)

Hence the determinant of A as a matrix over $\mathbb{Z}$ is equal to the determinant of A regarded as a matrix over $\mathbb{Z}/p\mathbb{Z}$. Therefore the computation of the determinant of A as a matrix over $\mathbb{Z}$ can be done modulo p, which provides a control on the intermediate computations. Now we have to answer the following questions:

1. How to choose p?
2. What do we win?

For choosing p we need an a-priori bound for the determinant of A. This is given by the following *Hadamard's inequality*.

Proposition

Let B be the maximal absolute value of an entry of A. Then we have

$$\mid d \mid \leq \; n^{n/2} B^n \tag{91}$$

Example

Consider

$$A = \begin{pmatrix} 4 & 5 \\ 6 & -7 \end{pmatrix} \tag{92}$$

Gaussian elimination leads to

$$A = \begin{pmatrix} 4 & 5 \\ 0 & -29/2 \end{pmatrix} \tag{93}$$

Hence $\det(A) = -58$. The Hadamard's inequality gives

$$|\det(A)| \leq 2^1\, 7^2 = 98 \tag{94}$$

The number $p = 199$ is prime and satisfies $p > 2 \times 98$. Gaussian elimination $\mod p$ leads to

$$A = \begin{pmatrix} 4 & 5 \\ 0 & 85 \end{pmatrix} \tag{95}$$

So $\det(A \mod p) = 141 = -58$ in $\mathbb{Z}/199\mathbb{Z}$.

Cost analysis (1/2)

Let us study what is the cost of this approach. Let us denote by C the determinant bound of Hadamard's inequality. Assume that our machine words are N-bit long. We make the following observations.

- The word length of C is in the order of magnitude of

$$\ell = \lceil \frac{1}{N} \log_2(C) \rceil + 1 = \lceil \frac{1}{N}\, n\, (\frac{1}{2} \log_2 n + \log_2 B) \rceil + 1. \quad (96)$$

- Prime numbers are frequent enough to find one with a word length in the same order of magnitude as C.
- So each element of $\mathbb{Z}/p\mathbb{Z}$ can be coded by an array with at most $\mathcal{O}(\ell)$ words.
- Hence, each operation (like addition, multiplication, inverse computation) in $\mathbb{Z}/p\mathbb{Z}$ costs at most $\mathcal{O}(\ell^2)$ word operations.
- Gaussian elimination $\mod p$ will require $\mathcal{O}(n^3)$ operations in $\mathbb{Z}/p\mathbb{Z}$.

Therefore we have proved the following theorem

Cost analysis (2/2)

Proposition

The determinant of a square matrix with order n, coefficients in $\mathbb{Z}$ and B as the maximal absolute value of a coefficient can be computed in $\mathcal{O}(n^3\, n^2\, (\log n + \log B)^2)$ *word operations.*

This is not in fact a big progress w.r.t. Gaussian elimination over $\mathbb{Q}$. But this can be improved using a *small primes modular computation* as follows.

Small prime approach (1/4)

Input: $A = (a_{ij})$ a square matrix over $\mathbb{Z}$ of order n with $|\, a_{ij} \,| \leq B$ for $1 \leq i, j \leq n$.

Output: $\det(A) \in \mathbb{Z}$.

$C := n^{n/2} B^n$
$r := \lceil \log_2(2\,C + 1) \rceil$
choose r distinct prime numbers $2 < m_0 < \cdots < m_{r-1} \in \mathbb{N}$
for $i = 0 \cdots r-1$ **repeat**
  $d_i := \det(A \mod m_i)$
$d := \mathit{interpolate}([m_0, \ldots, m_{r-1}], [d_0, \ldots, d_{r-1}])$
$d := d \,\mathrm{rem}\, m$
if $d \geq \frac{m}{2}$ **then**
  $d := d - m$
return(d)

Small prime approach (2/4)

Recall that $\det(A)$ is a polynomial expression in the coefficients of A. Hence by using the ring homomorphism between $\mathbb{Z}$ and $\mathbb{Z}/m_i\mathbb{Z}$ for $i = 0 \cdots r-1$ we have

$$\det(A) \equiv d_i \mod m_i \tag{97}$$

Using the CRT

$$\mathbb{Z}/m \simeq \mathbb{Z}/m_0 \times \cdots \times \mathbb{Z}/m_{r-1} \tag{98}$$

we deduce

$$\det(A) \equiv d \mod m \tag{99}$$

where m is the product of the moduli $m_0, \ldots, m_{r-1}$. Now observe that

$$\begin{array}{rcl} m & = & m_0 \cdots m_{r-1} \\ & \geq & 2^r \\ & \geq & 2\,C + 1 \\ & \geq & 2\,n^{n/2}\,B^n \\ & \geq & 2\,|\,d\,| \end{array} \tag{100}$$

Hence actually we have $\det(A) = d$.

Small prime approach (3/4)

Consider again

$$A = \begin{pmatrix} 4 & 5 \\ 6 & -7 \end{pmatrix} \tag{101}$$

We choose the four primes $2, 3, 5, 7$ so that $m = 210$. We get

$$\begin{array}{ll} \det(A) \equiv 0 \mod 2 & \det(A) \equiv 2 \mod 3 \\ \det(A) \equiv 2 \mod 5 & \det(A) \equiv -2 \mod 7 \end{array} \tag{102}$$

The solutions of the system $d \equiv d_i \mod m_i$ for $1 \leq i \leq 4$ are in

$$-58 + 210\mathbb{Z} = \{\ldots, -268, -58, 152, 362, \ldots\} \tag{103}$$

Finally $\det(A) = -58$ again.

Small prime approach (4/4)

Proposition

The determinant of a square matrix with order n, coefficients in $\mathbb{Z}$ and B as the maximal absolute value of a coefficient can be computed in $\mathcal{O}(n^4 \log^2(n\,B)(\log^2 n + \log^2 B))$ word operations.

Proof

See Theorem 5.12 in *Modern Computed Algebra*.

Remark

With the above algorithm, we achieve the following goals.

- All intermediate computations can be made modulo small prime numbers. In practice these small primes are machine integers allowing fast computations.
- The only possible large value is the determinant itself.
- The space and the time required for the whole computation can be estimated in advance.

Moreover the computations of the modular determinants (the d_i's) are pairwise independent and thus can be distributed.

Plan

Overview

We conclude this chapter with another modular algorithm. We assume that we have a highly efficient matrix multiplication over $\mathbb{Z}/p\mathbb{Z}$, for any machine-word size prime number p, and would like to take advantage of it for multipying matrcies with integer coefficients. This can be achieved by means of a modular algorithm, based on the Chinese Remaindering Algorithm.

Preliminaries (1/3)

Consider two square matrices $A = (a_{i,j}, 1 \leq i \leq j \leq n)$ and $B = (b_{i,j}, 1 \leq i \leq j \leq n)$ of order n with coefficients in $\mathbb{Z}$. Let $||A||_\infty$ and $||B||_\infty$ be the maximum absolute value of a coefficient in A and B, respectively. Let $C = (c_{i,j}, 1 \leq i \leq j \leq n)$ be the matrix product AB and let $m > 2$ be any odd integer (prime or not). For all $1 \leq i \leq j \leq n$, we have

$$c_{i,j} = \Sigma_{k=1}^{k=n} a_{i,k} b_{k,j}, \tag{104}$$

and thus

$$c_{i,j} \equiv \Sigma_{k=1}^{k=n} a_{i,k} b_{k,j} \mod m. \tag{105}$$

Preliminaries (2/3)

Now, let $\overline{A}^m = (\overline{a}_{i,j}^m, 1 \leq i \leq j \leq n)$ and $\overline{B}^m = (\overline{b}_{i,j}^m, 1 \leq i \leq j \leq n)$ be the images of A and B modulo m. (Hence the coefficient $\overline{a}_{i,j}^m$ of $\overline{A}^m$ is the remainder of $a_{i,j}$ modulo m.) Let $\overline{C}^m = (\overline{c}_{i,j}^m, 1 \leq i \leq j \leq n)$ be the matrix product $\overline{A}^m \overline{B}^m$ computed in $\mathbb{Z}/m\mathbb{Z}$. Hence, we have

$$\overline{c}_{i,j}^m = \Sigma_{k=1}^{k=n} \overline{a}_{i,k}^m \overline{b}_{k,j}^m \tag{106}$$

Combining the relations

$$\overline{a}_{i,k}^m \equiv a_{i,k} \mod m \text{ and } \overline{b}_{i,k}^m \equiv b_{i,k} \mod m \tag{107}$$

with Equations (105) and (106) we obtain

$$c_{i,j} \equiv \overline{c}_{i,j}^m \mod m. \tag{108}$$

Preliminaries (3/3)

In particular, if we use a symmetric representation $-\frac{m-1}{2} \cdots \frac{m-1}{2}$ for representing the elements of $\mathbb{Z}/m\mathbb{Z}$ and if we have $|c_{i,j}| < \frac{m}{2}$, then Equation (108) simply becomes $c_{i,j} = \overline{c}^m_{i,j}$.
Observe that for all $1 \leq i \leq j \leq n$, we have

$$| \, c_{i,j} \, | \leq \, \Sigma_{k=1}^{k=n} \, | \, a_{i,k} \, || \, b_{k,j} \, | \leq \, n || \, A \, ||_{\infty} \, || \, B \, ||_{\infty}. \qquad (109)$$

Hence we define $M = n||A||_{\infty}||B||_{\infty}$. We are ready to state a modular algorithm.

Algorithm (1/2)

Input: $A = (a_{ij})$ and $B = (b_{ij})$ two square matrices over $\mathbb{Z}$ of order n with $\mid a_{ij} \mid \leq \|A\|_\infty$ and $\mid b_{ij} \mid \leq \|B\|_\infty$, for $1 \leq i, j \leq n$.

Output: $C = (c_{ij})$ the matrix product $A\ B$.

$M := n\|A\|_\infty\|B\|_\infty$
$r := \lceil \log_2(2\,M + 1) \rceil$
choose r distinct prime numbers $2 < m_0 < \cdots < m_{r-1} \in \mathbb{N}$
$m := m_0 \cdots m_{r-1}$
for $\ell = 0 \cdots r-1$ **repeat**
  for $i = 1 \cdots n$ **repeat**
    for $j = 1 \cdots n$ **repeat**
      $\overline{c}_{i,j}^{m_\ell} := \sum_{k=1}^{k=n} \overline{a}_{i,k}^{m_\ell} \overline{b}_{k,j}^{m_\ell}$
for $i = 1 \cdots n$ **repeat**
  for $j = 1 \cdots n$ **repeat**
    $c_{i,j} := interpolate([m_0, \ldots, m_{r-1}], [\overline{c}_{i,j}^{m_0}, \ldots, \overline{c}_{i,j}^{m_{r-1}}])$
    $c_{i,j} := c_{i,j} \ \mathrm{rem}\ m$
    if $c_{i,j} \geq \frac{m}{2}$ **then**
      $c_{i,j} := c_{i,j} - m$
return$((c_{ij}))$

Algorithm (2/2)

Note that the first **for** loop computes the matrcies $\overline{C}^{m_0}, \ldots, \overline{C}^{m_{r-1}}$ by the classical method. However, one can use instead any other algorithm (like Strassen's) computing these matrices. Let us give an upper bound for the number of machine-word operations required by Algorithm **??**. It suffices to estimate each of the two **for** loops. The first one runs in $O(rn^3)$ and the second one in $O(n^2r^2)$. This is a better estimate than the one which can be given for the naive (non-modular approach): $O(n^3r^2)$.