Part one

The Kronecker – Duval Philosophy

And I saw when the Lamb opened one of the seals, and I heard, as it were the noise of thunder, one of the four beasts saying, Come and see.

And I saw, and behold a white horse: and he that sat on him had a bow; and a crown was given unto him: and he went forth conquering, and to conquer.
Revelations

The things depending from Saturn: bile, lead, onyx, asphodel, mole, hoopoe, eel.
E.C. Agrippa, *De occulta phylosophia*

Soon we will drink blood for wine.
Revolutionary of the Upper Rhine, *Book of a hundred chapters*

1

Euclid

This preliminary chapter is just devoted to recalling the Euclidean Algorithms over a univariate polynomial ring and its elementary applications: roughly speaking they are essentially the obvious generalization of those over integers.

The fundamental tool related to the Euclidean Algorithms and to solving univariate polynomials is nothing more than the elementary Division Algorithm (Section 1.1), whose iterative application produces the Euclidean Algorithm (Section 1.2), which can be extended to prove and compute Bezout's Identity (Section 1.3).

The Division- and Euclidean Algorithms and theorems have many important consequences for solving polynomial equations: they relate roots and linear factors of a polynomial (Section 1.4) allowing them, at least, to be counted, and are the basis for the theory (not the practice) of polynomial factorization (Section 1.5).

They also have another, more important, consequence which is a crucial tool in solving: they allow a computational system to be developed within quotients of polynomial rings; the discussion of this is postponed to Section 5.1.

A direct implementation of the Euclidean Algorithm provides an unexpected phenomenon, the 'coefficient explosion': during the application of the Euclidean Algorithm to two polynomials whose coefficients have small size, polynomials are produced with huge coefficients, even if the final output is simply 1. Finding efficient implementations of the Euclidean Algorithm was a crucial subject of research in the early days of Computer Algebra; in Section 1.6 I will briefly discuss this phenomenon and present efficient solutions to this problem.

1.1 The Division Algorithm

Throughout this chapter k will be a field and $\mathcal{P} := k[X]$ the univariate polynomial ring over k.

If $f = \sum_{i=0}^{n} a_i X^i \in \mathcal{P}$ with $a_n \neq 0$, denote by $\mathrm{lc}(f) := a_n$ the *leading coefficient* of f.

Theorem 1.1.1 (Division Theorem). *Given $A(X), B(X) \in \mathcal{P}$, $B \neq 0$, there are unique $Q(X), R(X) \in \mathcal{P}$ such that*

(1) $A(X) = Q(X)B(X) + R(X)$;
(2) $R \neq 0 \implies \deg(R) < \deg(B)$.

We call Q the quotient and R the remainder of A modulo B in $\mathcal{P}$.

Proof Existence: The proof is by induction on $\deg(A)$.
If $A = 0$ or $\deg(A) < \deg(B)$, then $Q := 0$ and $R := A$ obviously satisfy the thesis.
If $\deg(A) = n \geq m = \deg(B)$, we inductively assume that the theorem is true for each polynomial A_0 such that $A_0 = 0$ or $\deg(A_0) < n$. We then have

$$A(X) = a_n X^n + A_1(X), \quad B(X) = b_m X^m + B_1(X),$$

with $a_n \neq 0$, $b_m \neq 0$, $A_1 = 0$ or $\deg(A_1) < n$, $B_1 = 0$ or $\deg(B_1) < m$.
Let

$$A_0(X) := A(X) - a_n b_m^{-1} X^{n-m} B(X),$$

which, if non-zero, has degree less than n; by the inductive assumption there are then Q_0, R_0 such that

(1) $A_0(X) = Q_0(X)B(X) + R_0(X)$,
(2) $R_0 \neq 0 \implies \deg(R_0) < \deg(B)$,

so that

$$A(X) = (a_n b_m^{-1} X^{n-m} + Q_0(X))B(X) + R_0(X)$$

and therefore

$$Q(X) := a_n b_m^{-1} X^{n-m} + Q_0(X),\ R(X) := R_0(X)$$

satisfy the requirement.

Uniqueness: Assume that

(1) $A(X) = Q_1(X)B(X) + R_1(X)$,
(2) $A(X) = Q_2(X)B(X) + R_2(X)$,

(3) $R_i \neq 0 \implies \deg(R_i) < \deg(B), 1 \leq i \leq 2,$

so that

$$R_1(X) - R_2(X) = (Q_2(X) - Q_1(X))\, B(X).$$

If $R_1 \neq R_2$ then

$$\deg(R_1 - R_2) < \deg(B) \leq \deg(Q_2 - Q_1) + \deg(B) = \deg(R_1 - R_2)$$

giving a contradiction.
Therefore $R_1 - R_2 = 0$ and (since $B \neq 0$) also $Q_2 - Q_1 = 0$. $\boxed{♄}$

Corollary 1.1.2. *The ring $\mathcal{P}$ is a euclidean domain.* $\boxed{♄}$

In further applications, denote

$$Q := \mathbf{Quot}(A, B),\ R := \mathbf{Rem}(A, B).$$

Because of their uniqueness in $\mathcal{P}$, if K is a field such that $K \supseteq k$, the quotient and the remainder of A modulo B in $K[X]$ are still Q and R.

Algorithm 1.1.3. An inductive proof can be transformed into a recursive algorithm: If we assume k to be effective[1] then the iterative algorithm in Figure 1.1 performs polynomial division.

[1] The concept of effectiveness was first introduced as the notion of *endlichvielen Schritten* (finite number of steps) by Grete Hermann in 1926 for polynomial ideals in the fundamental paper

G. Hermann, *Die Frage der endlich vielen Schritte in der Theorie der Polynomideale*, *Math. Ann.* **95** (1926) 736–788,

where she wrote:

Die Behauptung, eine Berechnung kann mit endlich vielen Schritten durchgeführt werden, soll dabei bedeuten, es kann eine *obere Schranke für die Anzahl der zur Berechnung notwendigen Operationen* angegeben werden. Es genügt also z. B. nicht, ein Verfahren anzugeben, von dem man theoretisch nachweisen kann, daß es mit endlich vielen Operationen zum Ziele führt, wenn für die Anzahl dieser Operationen keine obere Schranke bekannt ist.
The assertion that a computation can be carried through in a finite number of steps shall mean that an upper bound for the number of operations needed for the computation *can be given. Thus it is not sufficient, for example, to give a procedure for which one can theoretically verify that it leads to the desired result in a finite number of operations, so long as no upper bound is known for the number of operations,*

To this, van der Waerden in

B.L. van der Waerden, Eine Bemerkung über die Unzelegbarkeit von Polynomen, *Math. Ann.* **102** (1930), 738–739,

Fig. 1.1. Polynomial Division Algorithm

```
(Q,R) := **PolynomialDivision**(A,B)
**where**
    $A, B \in k[X]$, $B \neq 0$
    $Q, R \in k[X]$ are such that
    – $A = QB + R$
    – $R \neq 0 \implies \deg(R) < \deg(B)$
$b := \mathrm{lc}(B), m := \deg(B)$
$A_0 := A, Q := 0$
**While** $A_0 \neq 0$ **and** $\deg(A_0) \geq \deg(B)$ **do**
    $a := \mathrm{lc}(A_0), n := \deg(A_0)$
    $Q := Q + ab^{-1}X^{n-m}$
    $A_0 := A_0 - ab^{-1}X^{n-m}B$
$R := A_0$
```

1.2 Euclidean Algorithm

Let $P_0, P_1 \in \mathcal{P}$, with $P_1 \neq 0$ (and, to dispose of the trivial cases, assume also that $P_0 \neq 0$). Let $P_2 :=$ **Rem**(P_0, P_1) and inductively, define

$$P_{i+1} := \mathbf{Rem}(P_{i-1}, P_i)$$

while $P_i \neq 0$. It is clear that the sequence $P_0, P_1, \ldots, P_i, \ldots$ (which is called the *polynomial remainder sequence* (PRS) of P_0, P_1) is finite since, otherwise,

added the note

Ein Körper K soll *explizite-bekann* heißen, wenn seine Elemente Symbole aus einem bekannten abzählbaren Vorrat von unterscheidbaren Symbolen sind, deren Addition, Multiplikation, Subtraktion und Division sich in endlichvielen Schritten ausführen lassen.
A field K is called explicitly given *when its elements are symbols from a known numerable set of distinguishable symbols, whose addition, multiplication, subtraction and division can be performed in* a finite number of steps.

In this book I will happily drop Hermann's requirement that an algorithm must be provided with its complexity evaluation, and will mainly follow Macaulay's opinion in

F.S. Macaulay, *The Algebraic Theory of Modular Systems*, Cambridge University Press (1916).

Macaulay considered the practical feasibility of an algorithm to be more crucial:

[The theory of polynomial ideals] might be regarded as in some measure complete if it were admitted that a problem is solved when its solution has been reduced to a finite number of feasible operations. If, however, the operations are too numerous or too involved to be carried out in practice the solution is only a theoretical one.

each P_i must be non-zero which would give an infinite decreasing sequence of natural numbers:

$$\deg(P_1) > \deg(P_2) > \cdots > \deg(P_i) > \cdots .$$

Let $D(X)$ denote the last non-zero element P_r of the sequence, and note that $r \leq \min(\deg(P_0), \deg(P_1))$. Also denote $Q_i := \mathbf{Quot}(P_{i-1}, P_i)$.

Proposition 1.2.1. $D(X) = \gcd(P_0, P_1)$.

Proof Since $P_{r-1} = Q_r P_r$, then P_r divides P_{r-1}. So let us assume that P_r divides P_i for $i > k$ and prove that it divides P_k: this is obvious from the identity

$$P_k = Q_{k+1} P_{k+1} + P_{k+2}.$$

Therefore $D = P_r$ is a common divisor of P_0 and P_1.
If $S(X)$ divides both P_0 and P_1, then since

$$P_2 = P_0 - Q_1 P_1,$$

it divides P_2. Assuming that S divides P_i, for $i < k$, then by the identity

$$P_k = P_{k-2} - Q_{k-1} P_{k-1},$$

it also divides P_k, therefore it divides P_r. $\boxed{♄}$

Greatest common divisors in $\mathcal{P}$ are obviously not unique, but they are associate (cf. Definition 1.5.1).

Again if K is a field such that $K \supseteq k$, $\gcd(A, B)$ and the PRS of A and B are the same in $K[X]$ as in $\mathcal{P}$.

Algorithm 1.2.2. If k is effective, the algorithm in Figure 1.2 computes the gcd of two polynomials; it actually computes the PRS of the two polynomials and also computes all the intermediate quotients Q_j.

Fig. 1.2. Euclidean Algorithm

```
$D :=$ GCD$(A, B)$
where
        $A, B \in \mathcal{P}$, $A \neq 0$, $B \neq 0$
        $D$ is a gcd$(A, B)$
$D := A$, $U := B$
While $U \neq 0$ do
        $(Q, V) :=$ PolynomialDivision$(D, U)$
        $D := U$, $U := V$
```

1.3 Bezout's Identity and Extended Euclidean Algorithm

Proposition 1.3.1 (Bezout's Identity). *Let $P_0, P_1 \in \mathcal{P} \setminus k$, and let us denote $D := \gcd(P_0, P_1)$. Then there are $S, T \in \mathcal{P} \setminus \{0\}$ such that*

(*i*) $P_0 S + P_1 T = D$
(*ii*) $\deg(S) < \deg(P_1)$, $\deg(T) < \deg(P_0)$

Proof Let $P_0, P_1, \dots, P_i, \dots, P_r = D$ be the PRS of P_0 and P_1. Also, for $i = 0, \dots, r-1$, let $Q_i := \mathbf{Quot}(P_{i-1}, P_i)$. Inductively define:

$$\begin{array}{lllllll}
S_0 & := & 1, & T_0 & := & 0; & \\
S_1 & := & 0, & T_1 & := & 1; & \\
S_i' & := & S_{i-2} - Q_{i-1} S_{i-1}, & T_i' & := & T_{i-2} - Q_{i-1} T_{i-1}, & 2 \le i \le r; \\
S_i & := & \mathbf{Rem}(S_i', P_1), & T_i & := & T_i' + \mathbf{Quot}(S_i', P_1) P_0, & 2 \le i \le r.
\end{array}$$

We claim that for $i = 0, \dots, r$:

(i) $P_0 S_i + P_1 T_i = P_i$;
(ii) $\deg(S_i) < \deg(P_1)$, $\deg(T_i) < \deg(P_0)$.

In fact the claims are trivial for $i = 0, 1$, and so, inductively assuming them to be true for $i < k$, and denoting $U_k := \mathbf{Quot}(S_k', P_1)$, so that

$$S_k' = U_k P_1 + S_k, \quad T_k = T_k' + U_k P_0,$$

we have

$$\begin{aligned}
P_k &= P_{k-2} - Q_{k-1} P_{k-1} \\
&= P_0 S_{k-2} + P_1 T_{k-2} - Q_{k-1} P_0 S_{k-1} - Q_{k-1} P_1 T_{k-1} \\
&= P_0 \left(S_{k-2} - Q_{k-1} S_{k-1}\right) + P_1 \left(T_{k-2} - Q_{k-1} T_{k-1}\right) \\
&= P_0 S_k' + P_1 T_k' \\
&= P_0 U_k P_1 + P_0 S_k + P_1 T_k - P_1 U_k P_0 \\
&= P_0 S_k + P_1 T_k.
\end{aligned}$$

Clearly $\deg(S_k) < \deg(P_1)$ and therefore also $\deg(T_k) < \deg(P_0)$, otherwise

$$\deg(P_1 T_k) \ge \deg(P_1 P_0) > \deg(S_k P_0)$$

and $\deg(P_1 T_k) > \deg(P_1) \ge \deg(P_k)$ would lead to an obvious contradiction. ♄

Corollary 1.3.2. *The ring $\mathcal{P}$ is a principal ideal domain.* ♄

Fig. 1.3. Extended Euclidean Algorithm

```
$(D, S, T) :=$ **ExtGCD**$(A, B)$
**where**
    $A, B \in \mathcal{P}, A \neq 0, B \neq 0$
    $D$ is a gcd$(A, B)$
    $SA + BT = D$
    $\deg(S) < \deg(B), \deg(T) < \deg(A)$
$D := A, U := B$
$S_0 := 1, S_1 := 0$
$\rightarrow T_0 := 0, T_1 := 1$
**While** $U \neq 0$ **do**
    $(Q, V) :=$ **PolynomialDivision**$(D, U)$
    $D := U, U := V$
    $S := S_0 - QS_1,$
    $\rightarrow T := T_0 - QT_1$
    $(Q, S) :=$ **PolynomialDivision**$(S, B)$
    $\rightarrow T := T + QA$
    $S_0 := S_1, S_1 := S$
    $\rightarrow T_0 := T_1, T_1 := T$
$S := S_0,$
$\rightarrow T := T_0$
```

Algorithm 1.3.3. Again, on an effective field, S and T can be computed by the algorithm in Figure 1.3.

Algorithm 1.3.4. The so-called Half-extended Euclidean Algorithm allows us to compute S, without having to compute T; it simply involves removing the lines marked by → in the algorithm in Figure 1.3. It is useful to compute inverses of field elements (see Remark 5.1.4).

1.4 Roots of Polynomials

The Division Theorem also has an obvious but important consequence on the solving of polynomial equations:

Corollary 1.4.1. *For $f(X) \in \mathcal{P}$, and $\alpha \in k$ we have:*

$$f(\alpha) = 0 \iff (X - \alpha) \text{ divides } P(X).$$

Proof Let

$$Q(X) := \textbf{Quot}(f(X), X - \alpha), \quad R(X) := \textbf{Rem}(f(X), X - \alpha);$$

since $(X - \alpha)$ is linear, either $R(X) = 0$ or $\deg(R) = 0$, i.e. $R(X)$ is a constant $r \in k$.

Therefore,

$$f(X) = Q(X)(X - \alpha) + r,$$

and evaluating in α obtains $f(\alpha) = r$, from which the proof follows. ♄

As a consequence a polynomial cannot have more roots than its degree.

1.5 Factorization of Polynomials

Definition 1.5.1. *In a domain D:*

(*i*) *two elements a and b are called* associate *if there exists $c \in D$, with c invertible, such that $a = bc$;*

(*ii*) *a non-zero and non-invertible element a is called* irreducible *if it is divisible only by invertible elements and by its associates, i.e.*

$$a = bc, \text{ and } b \text{ non-invertible} \implies c \text{ is invertible and so } b \text{ is associate to } a.$$

Definition 1.5.2. *A domain D is a* unique factorization domain *if for each non-invertible $a \in D \setminus \{0\}$*

(*i*) *there is a factorization $a = p_1 \dots p_r$ where each p_i is irreducible;*

(*ii*) *the factorization is unique in the following sense:*

if $a = q_1 \dots q_s$ is another factorization with q_i irreducible, then

- *$r = s$,*
- *each p_i is associate to some q_j,*
- *each q_j is associate to some p_i.*

Lemma 1.5.3. *If $p(X) \in k[X]$ is irreducible, p divides q_1q_2 and p does not divide q_2, then p divides q_1.*

Proof Since $\gcd(p, q_2)$ divides p, it either is associate to p or is a unit; since p does not divide q_2, we can then conclude that $\gcd(p, q_2) = 1$.
By Bezout's Identity, there are $s, t \in k[X]$, such that $sp + tq_2 = 1$ and therefore $spq_1 + tq_1q_2 = q_1$, so that p divides q_1. ♄

Lemma 1.5.4. *Let $f \in k[X]$; Let $f = p_1 \dots p_r$, $f = q_1 \dots q_s$ be two factorizations in irreducible factors. Then*

(*i*) *$r = s$,*

(*ii*) *each p_i is associate to some q_j,*

(*iii*) *each q_j is associate to some p_i.*