18

Linearly generated sequences and applications

In this chapter, we develop some of the theory of linearly generated sequences. As an application, we develop an efficient algorithm for solving sparse systems of linear equations, such as those that arise in the subexponential-time algorithms for discrete logarithms and factoring in Chapter 15. These topics illustrate the beautiful interplay between the arithmetic of polynomials, linear algebra, and the use of randomization in the design of algorithms.

18.1 Basic definitions and properties

Let F be a field, let V be an F-vector space, and consider an infinite sequence

$$\Psi = \{\alpha_i\}_{i=0}^{\infty}$$

where $\alpha_i \in V$ for $i = 0, 1, 2 \ldots.$ We say that Ψ is **linearly generated (over F)** if there exist scalars $c_0, \ldots, c_{k-1} \in F$ such that the following recurrence relation holds:

$$\alpha_{k+i} = \sum_{j=0}^{k-1} c_j \alpha_{j+i} \quad (\text{for } i = 0, 1, 2, \ldots).$$

In this case, all of the elements of the sequence Ψ are determined by the initial segment $\alpha_0, \ldots, \alpha_{k-1}$, together with the coefficients $c_0, \ldots, c_{k-1}$ defining the recurrence relation.

The general problem we consider is this: how to determine the coefficients defining such a recurrence relation, given a sufficiently long initial segment of Ψ. To study this problem, it turns out to be very useful to rephrase the problem slightly. Let $g \in F[X]$ be a polynomial of degree, say, k, and write $g = \sum_{j=0}^{k} a_j X^j$. Next, define

$$g \star \Psi := \sum_{j=0}^{k} a_j \alpha_j.$$

Then it is clear that Ψ is linearly generated if and only if there exists a non-zero polynomial g such that

$$(X^i g) \star \Psi = 0 \quad (\text{for } i = 0, 1, 2, \ldots). \tag{18.1}$$

Indeed, if there is such a non-zero polynomial g, then we can take

$$c_0 := -(a_0/a_k),\ c_1 := -(a_1/a_k),\ \ldots,\ c_{k-1} := -(a_{k-1}/a_k)$$

as coefficients defining the recurrence relation for Ψ. We call a polynomial g satisfying (18.1) a **generating polynomial** for Ψ. The sequence Ψ will in general have many generating polynomials. Note that the zero polynomial is technically considered a generating polynomial, but is not a very interesting one.

Let $G(\Psi)$ be the set of all generating polynomials for Ψ.

Theorem 18.1. *The set $G(\Psi)$ is an ideal of $F[X]$.*

Proof. First, note that for all $g, h \in F[X]$, we have $(g+h)\star\Psi = (g\star\Psi)+(h\star\Psi)$—this is clear from the definitions. It is also clear that for all $c \in F$ and $g \in F[X]$, we have $(cg) \star \Psi = c \cdot (g \star \Psi)$. From these two observations, it follows that $G(\Psi)$ is closed under addition and scalar multiplication. It is also easy to see from the definition that $G(\Psi)$ is closed under multiplication by X; indeed, if $(X^i g) \star \Psi = 0$ for all $i \geq 0$, then certainly, $(X^i(Xg)) \star \Psi = (X^{i+1}g) \star \Psi = 0$ for all $i \geq 0$. But any non-empty subset of $F[X]$ that is closed under addition, multiplication by elements of F, and multiplication by X is an ideal of $F[X]$ (see Exercise 7.27). □

Since all ideals of $F[X]$ are principal, it follows that $G(\Psi)$ is the ideal of $F[X]$ generated by some polynomial $\phi \in F[X]$—we can make this polynomial unique by choosing the monic associate (if it is non-zero), and we call this polynomial the **minimal polynomial of** Ψ. Thus, a polynomial $g \in F[X]$ is a generating polynomial for Ψ if and only if ϕ divides g; in particular, Ψ is linearly generated if and only if $\phi \neq 0$.

We can now restate our main objective as follows: given a sufficiently long initial segment of a linearly generated sequence, determine its minimal polynomial.

Example 18.1. One can always define a linearly generated sequence by simply choosing an initial segment $\alpha_0, \alpha_1, \ldots, \alpha_{k-1}$, along with scalars $c_0, \ldots, c_{k-1} \in F$ defining the recurrence relation. One can enumerate as many elements of the sequence as one wants by using storage for k elements of V, along with storage for the scalars $c_0, \ldots, c_{k-1}$, as follows:

$(\beta_0, \ldots, \beta_{k-1}) \leftarrow (\alpha_0, \ldots, \alpha_{k-1})$
repeat
  output β_0
  $\beta' \leftarrow \sum_{j=0}^{k-1} c_j \beta_j$
  $(\beta_0, \ldots, \beta_{k-1}) \leftarrow (\beta_1, \ldots, \beta_{k-1}, \beta')$
forever

Because of the structure of the above algorithm, linearly generated sequences are sometimes also called **shift register sequences**. Also observe that if F is a finite field, and V is finite dimensional, the value stored in the "register" $(\beta_0, \ldots, \beta_{k-1})$ must repeat at some point. It follows that the linearly generated sequence must be ultimately periodic (see definitions above Exercise 4.21). □

Example 18.2. Linearly generated sequences can also arise in a natural way, as this example and the next illustrate. Let $E := F[X]/(f)$, where $f \in F[X]$ is a monic polynomial of degree $\ell > 0$, and let α be an element of E. Consider the sequence $\Psi := \{\alpha^i\}_{i=0}^{\infty}$ of powers of α. For every polynomial $g = \sum_{j=0}^{k} a_j X^j \in F[X]$, we have

$$g \star \Psi = \sum_{j=0}^{k} a_j \alpha^j = g(\alpha).$$

Now, if $g(\alpha) = 0$, then clearly $(X^i g) \star \Psi = \alpha^i g(\alpha) = 0$ for all $i \geq 0$. Conversely, if $(X^i g) \star \Psi = 0$ for all $i \geq 0$, then in particular, $g(\alpha) = 0$. Thus, g is a generating polynomial for Ψ if and only if $g(\alpha) = 0$. It follows that the minimal polynomial ϕ of Ψ is the same as the minimal polynomial of α over F, as defined in §16.5. Furthermore, $\phi \neq 0$, and the degree m of ϕ may be characterized as the smallest positive integer m such that $\{\alpha^i\}_{i=0}^{m}$ is linearly dependent; moreover, as E has dimension ℓ over F, we must have $m \leq \ell$. □

Example 18.3. Let V be a vector space over F of dimension $\ell > 0$, and let $\tau : V \to V$ be an F-linear map. Let $\beta \in V$, and consider the sequence $\Psi := \{\alpha_i\}_{i=0}^{\infty}$, where $\alpha_i = \tau^i(\beta)$; that is, $\alpha_0 = \beta$, $\alpha_1 = \tau(\beta)$, $\alpha_2 = \tau(\tau(\beta))$, and so on. For every polynomial $g = \sum_{j=0}^{k} a_j X^j \in F[X]$, we have

$$g \star \Psi = \sum_{j=0}^{k} a_j \tau^j(\beta),$$

and for every $i \geq 0$, we have

$$(X^i g) \star \Psi = \sum_{j=0}^{k} a_j \tau^{i+j}(\beta) = \tau^i\Big(\sum_{j=0}^{k} a_j \tau^j(\beta)\Big) = \tau^i(g \star \Psi).$$

Thus, if $g \star \Psi = 0$, then clearly $(X^i g) \star \Psi = \tau^i(g \star \Psi) = \tau^i(0) = 0$ for all $i \geq 0$. Conversely, if $(X^i g) \star \Psi = 0$ for all $i \geq 0$, then in particular, $g \star \Psi = 0$. Thus, g is a generating polynomial for Ψ if and only if $g \star \Psi = 0$. The minimal polynomial ϕ of Ψ is non-zero and its degree m is at most ℓ; indeed, m may be characterized as the least non-negative integer such that $\{\tau^i(\beta)\}_{i=0}^m$ is linearly dependent, and since V has dimension ℓ over F, we must have $m \leq \ell$.

The previous example can be seen as a special case of this one, by taking V to be E, τ to be the α-multiplication map on E, and setting β to 1. □

The problem of computing the minimal polynomial of a linearly generated sequence can always be solved by means of Gaussian elimination. For example, the minimal polynomial of the sequence discussed in Example 18.2 can be computed using the algorithm described in §17.2. The minimal polynomial of the sequence discussed in Example 18.3 can be computed in a similar manner. Also, Exercise 18.3 below shows how one can reformulate another special case of the problem so that it is easily solved by Gaussian elimination. However, in the following sections, we will present algorithms for computing minimal polynomials for certain types of linearly generated sequences that are much more efficient than any algorithm based on Gaussian elimination.

EXERCISE 18.1. Show that the only sequence for which 1 is a generating polynomial is the "all zero" sequence.

EXERCISE 18.2. Let $\Psi = \{\alpha_i\}_{i=0}^\infty$ be a sequence of elements of an F-vector space V. Further, suppose that Ψ has non-zero minimal polynomial ϕ.

(a) Show that for all polynomials $g, h \in F[X]$, if $g \equiv h \pmod{\phi}$, then $g \star \Psi = h \star \Psi$.

(b) Let $m := \deg(\phi)$. Show that if $g \in F[X]$ and $(X^i g) \star \Psi = 0$ for all $i = 0, \ldots, m-1$, then g is a generating polynomial for Ψ.

EXERCISE 18.3. This exercise develops an alternative characterization of linearly generated sequences. Let $\Psi = \{z_i\}_{i=0}^\infty$ be a sequence of elements of F. Further, suppose that Ψ has minimal polynomial $\phi = \sum_{j=0}^m c_j X^j$ with $m > 0$ and $c_m = 1$. Define the matrix

$$A := \begin{pmatrix} z_0 & z_1 & \cdots & z_{m-1} \\ z_1 & z_2 & \cdots & z_m \\ \vdots & \vdots & \ddots & \vdots \\ z_{m-1} & z_m & \cdots & z_{2m-2} \end{pmatrix} \in F^{m \times m}$$

and the vector

$$w := (z_m, \dots, z_{2m-1}) \in F^{1\times m}.$$

Show that

$$v = (-c_0, \dots, -c_{m-1}) \in F^{1\times m}$$

is the *unique* solution to the equation

$$vA = w.$$

Hint: show that the rows of A form a linearly independent family of vectors by making use of Exercise 18.2 and the fact that no polynomial of degree less than m is a generating polynomial for Ψ.

EXERCISE 18.4. Let $c_0, \dots, c_{k-1} \in F$ and $z_0, \dots, z_{k-1} \in F$. For each $i \geq 0$, let

$$z_{k+i} := \sum_{j=0}^{k-1} c_j z_{j+i}.$$

Given $n \geq 0$, along with $c_0, \dots, c_{k-1}$ and $z_0, \dots, z_{k-1}$, show how to compute z_n using $O(\mathrm{len}(n)k^2)$ operations in F.

EXERCISE 18.5. Let V be a vector space over F, and consider the set $V^{\times\infty}$ of all infinite sequences $\{\alpha_i\}_{i=0}^{\infty}$, where the α_i's are in V. Let us define the scalar product of $g \in F[X]$ and $\Psi \in V^{\times\infty}$ as

$$g \cdot \Psi = \{(X^i g) \star \Psi\}_{i=0}^{\infty} \in V^{\times\infty}.$$

Show that with this scalar product, and addition defined component-wise, $V^{\times\infty}$ is an $F[X]$-module, and that a polynomial $g \in F[X]$ is a generating polynomial for $\Psi \in V^{\times\infty}$ if and only if $g \cdot \Psi = 0$.

18.2 Computing minimal polynomials: a special case

We now tackle the problem of efficiently computing the minimal polynomial of a linearly generated sequence from a sufficiently long initial segment.

We shall first address a special case of this problem, namely, the case where the vector space V is just the field F. In this case, we have

$$\Psi = \{z_i\}_{i=0}^{\infty},$$

where $z_i \in F$ for $i = 0, 1, 2, \dots$.

Suppose that we do not know the minimal polynomial ϕ of Ψ, but we know an upper bound $M > 0$ on its degree. Then it turns out that the initial segment $z_0, z_1, \dots z_{2M-1}$ completely determines ϕ, and moreover, we can very efficiently compute ϕ given this initial segment. The following theorem provides the essential ingredient.

Theorem 18.2. *Let $\Psi = \{z_i\}_{i=0}^{\infty}$ be a sequence of elements of F, and define the reversed Laurent series*

$$z := \sum_{i=0}^{\infty} z_i X^{-(i+1)} \in F((X^{-1})),$$

whose coefficients are the elements of the sequence Ψ. Then for every $g \in F[X]$, we have $g \in G(\Psi)$ if and only if $gz \in F[X]$. In particular, Ψ is linearly generated if and only if z is a rational function, in which case, its minimal polynomial is the denominator of z when expressed as a fraction in lowest terms.

Proof. Observe that for every polynomial $g \in F[X]$ and every integer $i \geq 0$, the coefficient of $X^{-(i+1)}$ in the product gz is equal to $X^i g \star \Psi$—just look at the formulas defining these expressions! It follows that g is a generating polynomial for Ψ if and only if the coefficients of the negative powers of X in gz are all zero, which is the same as saying that $gz \in F[X]$. Further, if $g \neq 0$ and $h := gz \in F[X]$, then $\deg(h) < \deg(g)$—this follows simply from the fact that $\deg(z) < 0$ (together with the fact that $\deg(h) = \deg(g) + \deg(z)$). All the statements in the theorem follow immediately from these observations. □

By virtue of Theorem 18.2, we can compute the minimal polynomial ϕ of Ψ using the algorithm in §17.5.1 for computing the numerator and denominator of a rational function from its reversed Laurent series expansion. More precisely, we can compute ϕ given the bound M on its degree, along with the first $2M$ elements $z_0, \ldots, z_{2M-1}$ of Ψ, using $O(M^2)$ operations in F. Just for completeness, we write down this algorithm:

1. Run the extended Euclidean algorithm on inputs

   $$f := X^{2M} \text{ and } h := z_0 X^{2M-1} + z_1 X^{2M-2} + \cdots + z_{2M-1},$$

   and apply Theorem 17.8 with f, h, $r^* := M$, and $t^* := M$, to obtain the polynomials r', s', t'.
2. Output $\phi := t' / \operatorname{lc}(t')$.

EXERCISE 18.6. Suppose F is a finite field and that $\Psi := \{z_i\}_{i=0}^{\infty}$ is linearly generated, with minimal polynomial ϕ. Further, suppose $X \nmid \phi$. Show that Ψ is purely periodic with period equal to the multiplicative order of $[X]_\phi \in (F[X]/(\phi))^*$. Hint: use Exercise 17.12 and Theorem 18.2.

18.3 Computing minimal polynomials: a more general case

Having dealt with the problem of finding the minimal polynomial of a linearly generated sequence Ψ, whose elements lie in F, we address the more general problem, where the elements of Ψ lie in a vector space V over F. We shall only deal with a special case of this problem, but it is one which has useful applications:

- First, we shall assume that V has finite dimension $\ell > 0$ over F.
- Second, we shall assume that the sequence $\Psi = \{\alpha_i\}_{i=0}^{\infty}$ has **full rank**, by which we mean the following: if the minimal polynomial ϕ of Ψ over F has degree m, then $\{\alpha_i\}_{i=0}^{m-1}$ is linearly independent. This property implies that the minimal polynomial of Ψ is the monic polynomial $\phi \in F[X]$ of least degree such that $\phi \star \Psi = 0$. The sequences considered in Examples 18.2 and 18.3 are of this type.
- Third, we shall assume that F is a finite field.

The dual space. Before presenting our algorithm for computing minimal polynomials, we need to discuss the **dual space** $\mathcal{D}_F(V)$ of V (over F), which consists of all F-linear maps from V into F. Thus, $\mathcal{D}_F(V) = \mathrm{Hom}_F(V, F)$, and is a vector space over F, with addition and scalar multiplication defined point-wise (see Theorem 13.12). We shall call elements of $\mathcal{D}_F(V)$ **projections**.

Now, fix a basis $\mathcal{S} = \{\gamma_i\}_{i=1}^{\ell}$ for V. As was discussed in §14.2, every element $\delta \in V$ has a unique coordinate vector $\mathrm{Vec}_{\mathcal{S}}(\delta) = (c_1, \ldots, c_\ell) \in F^{1\times\ell}$, where $\delta = \sum_i c_i\gamma_i$. Moreover, the map $\mathrm{Vec}_{\mathcal{S}} : V \to F^{1\times\ell}$ is a vector space isomorphism.

To each projection $\pi \in \mathcal{D}_F(V)$ we may also associate the **coordinate vector** $(\pi(\gamma_1), \ldots, \pi(\gamma_\ell))^\top \in F^{\ell\times 1}$. If $\mathcal{U}$ is the basis for F consisting of the single element 1_F, then the coordinate vector of π is $\mathrm{Mat}_{\mathcal{S},\mathcal{U}}(\pi)$, that is, the matrix of π relative to the bases $\mathcal{S}$ and $\mathcal{U}$. By Theorem 14.4, the map $\mathrm{Mat}_{\mathcal{S},\mathcal{U}} : \mathcal{D}_F(V) \to F^{\ell\times 1}$ is a vector space isomorphism.

In working with algorithms that compute with elements of V and $\mathcal{D}_F(V)$, we shall assume that such elements are represented using coordinate vectors relative to some convenient, fixed basis for V. If $\delta \in V$ has coordinate vector $(c_1, \ldots, c_\ell) \in F^{1\times\ell}$, and $\pi \in \mathcal{D}_F(V)$ has coordinate vector $(d_1, \ldots, d_\ell)^\top \in F^{\ell\times 1}$, then $\pi(\delta)$ is easily computed, using $O(\ell)$ operations in F, as $\sum_{i=1}^{\ell} c_i d_i$.

We now return to the problem of computing the minimal polynomial ϕ of the linearly generated sequence $\Psi = \{\alpha_i\}_{i=0}^{\infty}$. Assume we have a bound $M > 0$ on the degree of ϕ. Since Ψ has full rank and $\dim_F(V) = \ell$, we may assume that $M \leq \ell$.

For each $\pi \in \mathcal{D}_F(V)$, we may consider the *projected* sequence $\Psi_\pi := \{\pi(\alpha_i)\}_{i=0}^{\infty}$. Observe that ϕ is a generating polynomial for Ψ_π; indeed, for every polynomial $g \in F[X]$, we have $g \star \Psi_\pi = \pi(g \star \Psi)$, and hence, for all $i \geq 0$, we have $(X^i\phi) \star \Psi_\pi = \pi((X^i\phi) \star \Psi) = \pi(0) = 0$. Let $\phi_\pi \in F[X]$ denote the minimal polynomial of Ψ_π. Since ϕ_π divides every generating polynomial of Ψ_π, and since ϕ is a generating polynomial for Ψ_π, it follows that ϕ_π divides ϕ.

This suggests the following algorithm for efficiently computing the minimal polynomial of Ψ, using the first $2M$ terms of Ψ:

Algorithm MP. Given the first $2M$ terms of the sequence $\Psi = \{\alpha_i\}_{i=0}^{\infty}$, do the following:

> $g \leftarrow 1 \in F[X]$
> repeat
>   choose $\pi \in \mathcal{D}_F(V)$ at random
>   compute the first $2M$ terms of the projected sequence Ψ_π
>   use the algorithm in §18.2 to compute the minimal polynomial
> ϕ_π of Ψ_π
> $g \leftarrow \mathrm{lcm}(g, \phi_\pi)$
> until $g \star \Psi = 0$
> output g

A few remarks on the above procedure are in order:

- in every iteration of the main loop, g is the least common multiple of a number of divisors of ϕ, and hence is itself a divisor of ϕ; in particular, $\deg(g) \le M$;
- under our assumption that Ψ has full rank, and since g is a monic divisor of ϕ, if $g \star \Psi = 0$, we may safely conclude that $g = \phi$;
- under our assumption that F is finite, choosing a random element π of $\mathcal{D}_F(V)$ amounts to simply choosing at random the entries of the coordinate vector of π, relative to some basis for V;
- we also assume that elements of V are represented as coordinate vectors, so that applying a projection $\pi \in \mathcal{D}_F(V)$ to an element of V takes $O(\ell)$ operations in F; in particular, in each loop iteration, we can compute the first $2M$ terms of the projected sequence Ψ_π using $O(M\ell)$ operations in F;
- similarly, adding two elements of V, or multiplying an element of V by a scalar, takes $O(\ell)$ operations in F; in particular, in each loop iteration, we can compute $g \star \Psi$ using $O(M\ell)$ operations in F (and using the first $M + 1 \le 2M$ terms of Ψ).

Based on the above observations, it follows that when the algorithm halts, its output is correct, and that the cost of each loop iteration is $O(M\ell)$ operations in F. The remaining question to be answered is this: what is the expected number of iterations of the main loop? The answer to this question is $O(1)$, which leads to a total expected cost of Algorithm MP of $O(M\ell)$ operations in F.

The key to establishing that the expected number of iterations of the main loop is constant is provided by the following theorem.

Theorem 18.3. *Let $\Psi = \{\alpha_i\}_{i=0}^{\infty}$ be a linearly generated sequence over the field F, where the α_i's are elements of a vector space V of finite dimension $\ell > 0$. Let ϕ be the minimal polynomial of Ψ over F, let $m := \deg(\phi)$, and assume that Ψ has full rank (i.e., $\{\alpha_i\}_{i=0}^{m-1}$ is linearly independent). Finally, let $F[X]_{<m}$ denote the vector space over F consisting of all polynomials in $F[X]$ of degree less than m.*

Under the above assumptions, there exists a surjective F-linear map

$$\sigma : \mathcal{D}_F(V) \to F[X]_{<m}$$

such that for all $\pi \in \mathcal{D}_F(V)$, the minimal polynomial ϕ_π of the projected sequence $\Psi_\pi := \{\pi(\alpha_i)\}_{i=0}^{\infty}$ satisfies

$$\phi_\pi = \frac{\phi}{\gcd(\sigma(\pi), \phi)}.$$

Proof. While the statement of this theorem looks a bit complicated, its proof is quite straightforward, given our characterization of linearly generated sequences in Theorem 18.2 in terms of rational functions. We build the linear map σ as the composition of two linear maps, σ_0 and σ_1.

Let us define the map

$$\begin{aligned} \sigma_0 : \quad \mathcal{D}_F(V) &\to F((X^{-1})) \\ \pi &\mapsto \sum_{i=0}^{\infty} \pi(\alpha_i) X^{-(i+1)}. \end{aligned}$$

We also define the map σ_1 to be the ϕ-multiplication map on $F((X^{-1}))$—that is, the map that sends $z \in F((X^{-1}))$ to $\phi \cdot z \in F((X^{-1}))$. The map σ is just the composition $\sigma = \sigma_1 \circ \sigma_0$. It is clear that both σ_0 and σ_1 are F-linear maps, and hence, so is σ.

First, observe that for $\pi \in \mathcal{D}_F(V)$, the series $z := \sigma_0(\pi)$ is the series associated with the projected sequence Ψ_π, as in Theorem 18.2. Let ϕ_π be the minimal polynomial of Ψ_π. Since ϕ is a generating polynomial for Ψ, it is also a generating polynomial for Ψ_π. Therefore, Theorem 18.2 tells us that

$$h := \sigma(\pi) = \phi \cdot z \in F[X]_{<m},$$

and that ϕ_π is the denominator of z when expressed as a fraction in lowest terms. Now, we have $z = h/\phi$, and it follows that $\phi_\pi = \phi/\gcd(h, \phi)$ is this denominator.

Second, the hypothesis that $\{\alpha_i\}_{i=0}^{m-1}$ is linearly independent implies that $\dim_F(\operatorname{Im}\sigma_0) \geq m$ (see Exercise 13.21). Also, observe that σ_1 is an injective map. Therefore, $\dim_F(\operatorname{Im}\sigma) \geq m$. In the previous paragraph, we observed that $\operatorname{Im}\sigma \subseteq F[X]_{<m}$, and since $\dim_F(F[X]_{<m}) = m$, we may conclude that $\operatorname{Im}\sigma = F[X]_{<m}$. That proves the theorem. □

Given the above theorem, we can analyze the expected number of iterations of the main loop of Algorithm MP.

First of all, we may as well assume that the degree m of ϕ is greater than 0, as otherwise, we are sure to get ϕ in the very first iteration. Let $\pi_1, \ldots, \pi_s$ be the random projections chosen in the first s iterations of Algorithm MP. By Theorem 18.3, each $\sigma(\pi_i)$ is uniformly distributed over $F[X]_{<m}$, and we have $g = \phi$ at the end of loop iteration s if and only if $\gcd(\phi, \sigma(\pi_1), \ldots, \sigma(\pi_s)) = 1$.

Let us define $\Lambda_F^\phi(s)$ to be the probability that $\gcd(\phi, f_1, \ldots, f_s) = 1$, where $f_1, \ldots, f_s$ are randomly chosen from $F[X]_{<m}$. Thus, the probability that we have $g = \phi$ at the end of loop iteration s is equal to $\Lambda_F^\phi(s)$. While one can analyze the quantity $\Lambda_F^\phi(s)$, it turns out to be easier, and sufficient for our purposes, to analyze a different quantity. Let us define $\Lambda_F^m(s)$ to be the probability that $\gcd(f_1, \ldots, f_s) = 1$, where $f_1, \ldots, f_s$ are randomly chosen from $F[X]_{<m}$. Clearly, $\Lambda_F^\phi(s) \geq \Lambda_F^m(s)$.

Theorem 18.4. *If F is a finite field of cardinality q, and m and s are positive integers, then we have*

$$\Lambda_F^m(s) = 1 - 1/q^{s-1} + (q-1)/q^{sm}.$$

Proof. For each positive integer n, let U_n denote the set of all tuples of polynomials $(f_1, \ldots, f_s) \in F[X]_{<n}^{\times s}$ with $\gcd(f_1, \ldots, f_s) = 1$, and let $u_n := |U_n|$. Also, for each monic polynomial $h \in F[X]$ of degree less that n, let $U_{n,h}$ denote the set of all s-tuples of polynomials of degree less than n whose gcd is h. Observe that the set $U_{n,h}$ is in one-to-one correspondence with U_{n-k}, where $k := \deg(h)$, via the map that sends $(f_1, \ldots, f_s) \in U_{n,h}$ to $(f_1/h, \ldots, f_s/h) \in U_{n-k}$. As there are q^k possible choices for h of degree k, if we define $V_{n,k}$ to be the set of tuples $(f_1, \ldots, f_s) \in F[X]_{<n}^{\times s}$ with $\deg(\gcd(f_1, \ldots, f_s)) = k$, we see that $|V_{n,k}| = q^k u_{n-k}$. Every non-zero tuple in $F[X]_{<n}^{\times s}$ appears in exactly one of the sets $V_{n,k}$, for $k = 0, \ldots, n-1$. Taking into account the zero tuple, it follows that

$$q^{sn} = 1 + \sum_{k=0}^{n-1} q^k u_{n-k}, \tag{18.2}$$

which holds for all $n \geq 1$. Replacing n by $n-1$ in (18.2), we obtain

$$q^{s(n-1)} = 1 + \sum_{k=0}^{n-2} q^k u_{n-1-k}, \tag{18.3}$$

which holds for all $n \geq 2$, and indeed, holds for $n = 1$ as well. Subtracting q times (18.3) from (18.2), we deduce that for all $n \geq 1$,

$$q^{sn} - q^{sn-s+1} = 1 + u_n - q,$$

and rearranging terms:

$$u_n = q^{sn} - q^{sn-s+1} + q - 1.$$

Therefore,

$$\Lambda_F^m(s) = u_m/q^{sm} = 1 - 1/q^{s-1} + (q-1)/q^{sm}. \quad \square$$

From the above theorem, it follows that for $s \geq 1$, the probability P_s that Algorithm MP runs for more than s loop iterations is at most $1/q^{s-1}$. If L is the total number of loop iterations, then

$$\mathsf{E}[L] = \sum_{i \geq 1} \mathsf{P}[L \geq i] = 1 + \sum_{s \geq 1} P_s \leq 1 + \sum_{s \geq 1} 1/q^{s-1} = 1 + \frac{q}{q-1} \leq 3.$$

Let us summarize all of the above analysis with the following:

Theorem 18.5. *Let Ψ be a sequence of elements of an F-vector space V of finite dimension $\ell > 0$ over F, where F is a finite field. Assume that Ψ is linearly generated over F with minimal polynomial $\phi \in F[X]$ of degree m, and that Ψ has full rank (i.e., the first m terms of Ψ form a linearly independent family of elements). Then given an upper bound $M > 0$ on m, along with the first $2M$ elements of Ψ, Algorithm* MP *correctly computes ϕ using an expected number of $O(M\ell)$ operations in F.*

We close this section with the following observation. Suppose the sequence Ψ is of the form $\{\tau^i(\beta)\}_{i=0}^{\infty}$, where $\beta \in V$ and $\tau : V \to V$ is an F-linear map. Suppose that with respect to some basis $\mathcal{S}$ for V, elements of V are represented by their coordinate vectors (which are elements of $F^{1\times\ell}$), and elements of $\mathcal{D}_F(V)$ are represented by their coordinate vectors (which are elements of $F^{\ell\times 1}$). The linear map τ also has a corresponding matrix $A = \mathrm{Mat}_{\mathcal{S},\mathcal{S}}(V, V) \in F^{\ell\times\ell}$, so that evaluating τ at a point α in V corresponds to multiplying the coordinate vector of α on the right by A. Now, suppose $\beta \in V$ has coordinate vector $v \in F^{1\times\ell}$ and that $\pi \in \mathcal{D}_F(V)$ has coordinate vector $w \in F^{\ell\times 1}$. Then if Ψ' is the sequence of coordinate vectors of the elements of Ψ, we have

$$\Psi' = \{vA^i\}_{i=0}^{\infty} \text{ and } \Psi_\pi = \{vA^i w\}_{i=0}^{\infty}.$$

This more concrete, matrix-oriented point of view is sometimes useful; in particular, it makes quite transparent the symmetry of the roles played by β and π in forming the projected sequence.

EXERCISE 18.7. If $|F| = q$ and $\phi \in F[X]$ is monic and factors into monic irreducible polynomials in $F[X]$ as $\phi = \phi_1^{e_1} \cdots \phi_r^{e_r}$, show that

$$\Lambda_F^{\phi}(1) = \prod_{i=1}^{r}(1 - q^{-\deg(\phi_i)}) \geq 1 - \sum_{i=1}^{r} q^{-\deg(\phi_i)}.$$

From this, conclude that the probability that Algorithm MP terminates after just one loop iteration is $1 - O(m/q)$, where $m = \deg(\phi)$. Thus, if q is very large relative to m, it is highly likely that Algorithm MP terminates after just one iteration of the main loop.

18.4 Solving sparse linear systems

Let V be a vector space of finite dimension $\ell > 0$ over a finite field F, and let $\tau : V \to V$ be an F-linear map. The goal of this section is to develop time- and space-efficient algorithms for solving equations of the form

$$\tau(\gamma) = \delta; \tag{18.4}$$

that is, given τ and $\delta \in V$, find $\gamma \in V$ satisfying (18.4). The algorithms we develop will have the following properties: they will be probabilistic, and will use an expected number of $O(\ell^2)$ operations in F, an expected number of $O(\ell)$ evaluations of τ, and space for $O(\ell)$ elements of F. By an "evaluation of τ," we mean the computation of $\tau(\alpha)$ for a given $\alpha \in V$.

We shall assume that elements of V are represented as coordinate vectors with respect to some fixed basis for V. This means that a single element of V is represented as a vector of ℓ elements of F. Now, if the matrix of τ with respect to the given basis is sparse, having, say, $\ell^{1+o(1)}$ non-zero entries, then the space required to represent τ is $\ell^{1+o(1)}$ elements of F, and the time required to evaluate τ is $\ell^{1+o(1)}$ operations in F. Under these assumptions, our algorithms to solve (18.4) use an expected number of $\ell^{2+o(1)}$ operations in F, and space for $\ell^{1+o(1)}$ elements of F. This is to be compared with standard Gaussian elimination: even if the original matrix is sparse, during the execution of the algorithm, most of the entries in the matrix may eventually be "filled in" with non-zero field elements, leading to a running time of $\Omega(\ell^3)$ operations in F, and a space requirement of $\Omega(\ell^2)$ elements of F. Thus, the algorithms presented here will be much more efficient than Gaussian elimination when the matrix of τ is sparse.

We hasten to point out that the algorithms presented here may be more efficient than Gaussian elimination in other cases, as well. All that matters is that τ can be evaluated using $o(\ell^2)$ operations in F and/or represented using space for $o(\ell^2)$ elements of F—in either case, we obtain a time and/or space improvement over Gaussian elimination. Indeed, there are applications where the matrix of the linear map τ may not be sparse, but nevertheless has special structure that allows it to be represented and evaluated in subquadratic time and/or space.

We shall only present algorithms that work in two special, but important, cases:

- the first case is where τ is bijective,
- the second case is where τ is not bijective, $\delta = 0$, and a non-zero solution γ to (18.4) is required (i.e., we are looking for a non-zero element of $\operatorname{Ker} \tau$).

In both cases, the key will be to use Algorithm MP in §18.3 to find the minimal polynomial ϕ of the linearly generated sequence

$$\Psi := \{\alpha_i\}_{i=0}^{\infty} \quad (\alpha_i := \tau^i(\beta),\ i = 0, 1, \ldots), \tag{18.5}$$

where β is a suitably chosen element of V. From the discussion in Example 18.3, this sequence has full rank, and so we may use Algorithm MP. We may use $M := \ell$ as an upper bound on the degree of ϕ (assuming we know nothing more about τ and β that would allow us to use a smaller upper bound). In using Algorithm MP in this application, note that we do not want to store $\alpha_0, \ldots, \alpha_{2\ell-1}$—if we did, we would not satisfy our stated space bound. Instead of storing the α_i's in a "warehouse," we use a "just in time" strategy for computing them, as follows:

- In the body of the main loop of Algorithm MP, where we calculate the projections $z_i := \pi(\alpha_i)$, for $i = 0 \ldots 2\ell - 1$, we perform the computation as follows:

  $\alpha \leftarrow \beta$
  for $i \leftarrow 0$ to $2\ell - 1$ do
  $\quad z_i \leftarrow \pi(\alpha),\ \alpha \leftarrow \tau(\alpha)$

- In the test at the bottom of the main loop of Algorithm MP, if $g = \sum_{j=0}^{k} a_j X^j$, we compute $v := g \star \Psi \in V$ using the following Horner-like scheme:

  $v \leftarrow 0$
  for $j \leftarrow k$ down to 0 do
  $\quad v \leftarrow \tau(v) + a_j \cdot \beta$

With this implementation, Algorithm MP uses an expected number of $O(\ell^2)$ operations in F, an expected number of $O(\ell)$ evaluations of τ, and space for $O(\ell)$ elements of F. Of course, the "warehouse" strategy is faster than the "just in time" strategy by a constant factor, but it uses about ℓ times as much space; thus, for large ℓ, using the "just in time" strategy is a very good time/space trade-off.

The bijective case. Now consider the case where τ is bijective, and we want to solve (18.4) for a given $\delta \in V$. We may as well assume that $\delta \neq 0$, since otherwise, $\gamma = 0$ is the unique solution to (18.4). We proceed as follows. First, using Algorithm MP as discussed above, compute the minimal polynomial ϕ of the sequence Ψ defined in (18.5), using $\beta := \delta$. Let $\phi = \sum_{j=0}^{m} c_j X^j$, where $c_m = 1$ and $m > 0$. Then we have

$$c_0\delta + c_1\tau(\delta) + \cdots + c_m\tau^m(\delta) = 0. \tag{18.6}$$

We claim that $c_0 \neq 0$. To prove the claim, suppose that $c_0 = 0$. Then applying τ^{-1} to (18.6), we would obtain

$$c_1\delta + \cdots + c_m\tau^{m-1}(\delta) = 0,$$

which would imply that ϕ/X is a generating polynomial for Ψ, contradicting the minimality of ϕ. That proves the claim.

Since $c_0 \neq 0$, we can apply τ^{-1} to (18.6), and solve for $\gamma = \tau^{-1}(\delta)$ as follows:

$$\gamma = -c_0^{-1}(c_1\delta + \cdots + c_m\tau^{m-1}(\delta)).$$

To actually compute γ, we use the same "just in time" strategy as was used in the implementation of the computation of $g \star \Psi$ in Algorithm MP, which costs $O(\ell^2)$ operations in F, $O(\ell)$ evaluations of τ, and space for $O(\ell)$ elements of F.

The non-bijective case. Now consider the case where τ is not bijective, and we want to find non-zero $\gamma \in V$ such that $\tau(\gamma) = 0$. The idea is this. Suppose we choose an arbitrary, non-zero element β of V, and use Algorithm MP to compute the minimal polynomial ϕ of the sequence Ψ defined in (18.5), using this value of β. Let $\phi = \sum_{j=0}^{m} c_j X^j$, where $m > 0$ and $c_m = 1$. Then we have

$$c_0\beta + c_1\tau(\beta) + \cdots + c_m\tau^m(\beta) = 0. \tag{18.7}$$

Let

$$\gamma := c_1\beta + \cdots + c_m\tau^{m-1}(\beta).$$

We must have $\gamma \neq 0$, since $\gamma = 0$ would imply that $\lfloor \phi/X \rfloor$ is a non-zero generating polynomial for Ψ, contradicting the minimality of ϕ. If it happens that $c_0 = 0$, then equation (18.7) implies that $\tau(\gamma) = 0$, and we are done. As before, to actually compute γ, we use the same "just in time" strategy as was used in the implementation of the computation of $g \star \Psi$ in Algorithm MP, which costs $O(\ell^2)$ operations in F, $O(\ell)$ evaluations of τ, and space for $O(\ell)$ elements of F.

The above approach fails if $c_0 \neq 0$. However, in this "bad" case, equation (18.7) implies that $\beta = -c_0^{-1}\tau(\gamma)$; in particular, $\beta \in \operatorname{Im}\tau$. One way to avoid such a "bad" β is to randomize: as τ is not surjective, the image of τ is a subspace of V of dimension strictly less than ℓ, and therefore, a *randomly* chosen β lies in the image of τ with probability at most $1/|F|$. So a simple technique is to choose repeatedly β at random until we get a "good" β. The overall complexity of the resulting algorithm will be as required: $O(\ell^2)$ expected operations in F, $O(\ell)$ expected evaluations of τ, and space for $O(\ell)$ elements of F.

As a special case of this situation, consider the problem that arose in Chapter 15 in connection with algorithms for computing discrete logarithms and factoring. We had to solve the following problem: given an $\ell \times (\ell - 1)$ matrix A with entries in a finite field F, containing $\ell^{1+o(1)}$ non-zero entries, find non-zero $v \in F^{1\times\ell}$ such that $vA = 0$. To solve this problem, we can augment the matrix A, adding an extra column of zeros, to get an $\ell \times \ell$ matrix A'. Now, let $V = F^{1\times\ell}$ and let τ be the F-linear map on V that sends $\gamma \in V$ to $\gamma A'$. A non-zero solution γ to the equation $\tau(\gamma) = 0$ will provide us with the solution to our original problem; thus, we can apply the above technique directly, solving this problem using $\ell^{2+o(1)}$ expected operations in F, and space for $\ell^{1+o(1)}$ elements of F. As a side remark, in this particular application, we can choose a "good" β in the above algorithm without randomization: just choose $\beta := (0, \dots, 0, 1)$, which is clearly not in the image of τ.

18.5 Computing minimal polynomials in $F[X]/(f)$ (II)

Let us return to the problem discussed in §17.2: F is a field, $f \in F[X]$ is a monic polynomial of degree $\ell > 0$, and $E := F[X]/(f)$; we are given an element $\alpha \in E$, and want to compute the minimal polynomial $\phi \in F[X]$ of α over F. As discussed in Example 18.2, this problem is equivalent to the problem of computing the minimal polynomial of the sequence

$$\Psi := \{\alpha_i\}_{i=0}^{\infty} \quad (\alpha_i := \alpha^i, \ i = 0, 1, \dots),$$

and the sequence has full rank; therefore, we can use Algorithm MP in §18.3 directly to solve this problem, assuming F is a finite field.

If we use the "just in time" strategy in the implementation of Algorithm MP, as was used in §18.4, we get an algorithm that computes the minimal polynomial of α using $O(\ell^3)$ expected operations in F, but space for just $O(\ell^2)$ elements of F. Thus, in terms of space, this approach is far superior to the algorithm in §17.2, based on Gaussian elimination. In terms of time complexity, the algorithm based on linearly generated sequences is a bit slower than the one based on Gaussian elimination (but only by a constant factor). However, if we use any subquadratic-time algorithm for polynomial arithmetic (see §17.6 and §17.7), we immediately get an algorithm that runs in subcubic time, while still using linear space. In the exercises below, you are asked to develop an algorithm that computes the minimal polynomial of α using just $O(\ell^{2.5})$ operations in F, at the expense of requiring space for $O(\ell^{1.5})$ elements of F—this algorithm does not rely on fast polynomial arithmetic, and can be made even faster if such arithmetic is used.

EXERCISE 18.8. Let $f \in F[X]$ be a monic polynomial of degree $\ell > 0$ over a field F, and let $E := F[X]/(f)$. Also, let $\xi := [X]_f \in E$. For computational purposes, we assume that elements of E and $\mathcal{D}_F(E)$ are represented as coordinate vectors with respect to the usual "polynomial" basis $\{\xi^{i-1}\}_{i=1}^{\ell}$. For $\beta \in E$, let M_β denote the β-multiplication map on E that sends $\alpha \in E$ to $\alpha\beta \in E$, which is an F-linear map from E into E.

(a) Given as input the polynomial f defining E, along with a projection $\pi \in \mathcal{D}_F(E)$ and an element $\beta \in E$, show how to compute the projection $\pi \circ M_\beta \in \mathcal{D}_F(E)$, using $O(\ell^2)$ operations in F.

(b) Given as input the polynomial f defining E, along with a projection $\pi \in \mathcal{D}_F(E)$, an element $\alpha \in E$, and a parameter $k > 0$, show how to compute $(\pi(1), \pi(\alpha), \ldots, \pi(\alpha^{k-1}))$ using just $O(k\ell + k^{1/2}\ell^2)$ operations in F, and space for $O(k^{1/2}\ell)$ elements of F. Hint: use the same hint as in Exercise 17.3.

EXERCISE 18.9. Let $f \in F[X]$ be a monic polynomial over a finite field F of degree $\ell > 0$, and let $E := F[X]/(f)$. Show how to use the result of the previous exercise, as well as Exercise 17.3, to get an algorithm that computes the minimal polynomial of $\alpha \in E$ over F using $O(\ell^{2.5})$ expected operations in F, and space for $O(\ell^{1.5})$ operations in F.

EXERCISE 18.10. Let $f \in F[X]$ be a monic polynomial of degree $\ell > 0$ over a field F (not necessarily finite), and let $E := F[X]/(f)$. Further, suppose that f is irreducible, so that E is itself a field. Show how to compute the minimal polynomial of $\alpha \in E$ over F *deterministically*, using algorithms that satisfy the following complexity bounds:

(a) $O(\ell^3)$ operations in F and space for $O(\ell)$ elements of F;

(b) $O(\ell^{2.5})$ operations in F and space for $O(\ell^{1.5})$ elements of F.

18.6 The algebra of linear transformations (∗)

Throughout this chapter, one could hear the whispers of the algebra of linear transformations. We develop some of the aspects of this theory here, leaving a number of details as exercises. It will not play a role in any material that follows, but it serves to provide the reader with a "bigger picture."

Let F be a field and V be an F-vector space. We denote by $\mathcal{L}_F(V)$ the set of all F-linear maps from V into V. Thus, $\mathcal{L}_F(V) = \mathrm{Hom}_F(V, V)$, and is a vector space over F, with addition and scalar multiplication defined point-wise (see Theorem 13.12). Elements of $\mathcal{L}_F(V)$ are called **linear transformations**.

For $\tau, \tau' \in \mathcal{L}_F(V)$, the composed map, $\tau \circ \tau'$, which sends $\alpha \in V$ to $\tau(\tau'(\alpha))$

is also an element of $\mathcal{L}_F(V)$. As always, function composition is associative (i.e., for $\tau, \tau', \tau'' \in \mathcal{L}_F(V)$, we have $\tau \circ (\tau' \circ \tau'') = (\tau \circ \tau') \circ \tau''$); however, function composition is not in general commutative (i.e., we may have $\tau \circ \tau' \neq \tau' \circ \tau$ for some $\tau, \tau' \in \mathcal{L}_F(V)$). The following theorem considers the interaction between composition, addition, and scalar multiplication.

Theorem 18.6. *For all $\tau, \tau', \tau'' \in \mathcal{L}_F(V)$, and for all $c \in F$, we have:*

(i) $\tau \circ (\tau' + \tau'') = \tau \circ \tau' + \tau \circ \tau''$;

(ii) $(\tau' + \tau'') \circ \tau = \tau' \circ \tau + \tau'' \circ \tau$;

(iii) $(c\tau) \circ \tau' = c(\tau \circ \tau') = \tau \circ (c\tau')$.

Proof. Exercise. □

Under the addition operation and scalar multiplication of the vector space $\mathcal{L}_F(V)$, and defining multiplication on $\mathcal{L}_F(V)$ using the "$\circ$" operation, the previous theorem implies that $\mathcal{L}_F(V)$ satisfies all the properties of an F-algebra (see Definition 16.1), except for the fact that multiplication is not commutative (the identity map acts as the multiplicative identity). Thus, we can think of $\mathcal{L}_F(V)$ as a *non-commutative F-algebra.*

Let $\tau \in \mathcal{L}_F(V)$ be a linear transformation. For each integer $i \geq 0$, the map τ^i (i.e., the i-fold composition of τ) is also an element of $\mathcal{L}_F(V)$. Note that τ^0 is by definition just the identity map on V. For each polynomial $g \in F[X]$, with $g = \sum_i a_i X^i$, we denote by $g(\tau)$ the linear transformation

$$g(\tau) := \sum_i a_i \tau^i \in \mathcal{L}_F(V).$$

Thus, for $\alpha \in V$, the value of $g(\tau)$ at α is $\sum_i a_i \tau^i(\alpha)$.

Theorem 18.7. *For all $\tau \in \mathcal{L}_F(V)$, for all $c \in F$, and for all $g, h \in F[X]$, we have:*

(i) $g(\tau) + h(\tau) = (g + h)(\tau)$;

(ii) $c \cdot g(\tau) = (cg)(\tau)$;

(iii) $g(\tau) \circ h(\tau) = (gh)(\tau) = h(\tau) \circ g(\tau)$.

Proof. Exercise. □

Let $\tau \in \mathcal{L}_F(V)$ be a linear transformation. We define

$$F[\tau] := \{g(\tau) : g \in F[X]\},$$

which is a subset of $\mathcal{L}_F(V)$. By the previous theorem, it is clear that $F[\tau]$ is closed under addition, multiplication (i.e., composition), and scalar multiplication, and that $F[\tau]$ is in fact an F-algebra in the usual sense (i.e., multiplication is commutative). Moreover, the expressions $F[\tau]$ and $g(\tau)$ (for $g \in F[X]$) have the same meaning as in §16.1.

Let ϕ_τ be the minimal polynomial of τ over F, so that $F[\tau]$ is isomorphic as an F-algebra to $F[X]/(\phi_\tau)$. We can also characterize ϕ_τ as follows:

> if there exists a non-zero polynomial $g \in F[X]$ such that $g(\tau) = 0$, then ϕ_τ is the monic polynomial of least degree with this property; otherwise, $\phi_\tau = 0$.

Another way to characterize ϕ_τ is as follows:

> ϕ_τ is the minimal polynomial of the sequence $\{\tau^i\}_{i=0}^{\infty}$.

If V has finite dimension $\ell > 0$, then by Theorem 14.4, $\mathcal{L}_F(V)$ is isomorphic as an F-vector space to $F^{\ell\times\ell}$, and so in particular, has dimension ℓ^2. Therefore, there must be a linear dependence among $1, \tau, \ldots, \tau^{\ell^2}$, which implies that the minimal polynomial of τ is non-zero with degree at most ℓ^2 (and at least 1). We shall show below that in this case, the minimal polynomial of τ actually has degree at most ℓ.

For a fixed $\tau \in \mathcal{L}_F(V)$, we can define a "scalar multiplication" operation $\odot$, that maps $g \in F[X]$ and $\alpha \in V$ to

$$g \odot \alpha := g(\tau)(\alpha) \in V;$$

that is, if $g = \sum_i a_i X^i$, then

$$g \odot \alpha = \sum_i a_i \tau^i(\alpha).$$

Theorem 18.8. *The scalar multiplication $\odot$, together with the usual addition operation on V, makes V into an $F[X]$-module; that is, for all $g, h \in F[X]$ and $\alpha, \beta \in V$, we have*

$$g \odot (h \odot \alpha) = (gh) \odot \alpha,\ (g + h) \odot \alpha = g \odot \alpha + h \odot \alpha,$$
$$g \odot (\alpha + \beta) = g \odot \alpha + g \odot \beta,\ 1 \odot \alpha = \alpha.$$

Proof. Exercise. □

Note that each choice of τ gives rise to a different $F[X]$-module structure, but all of these structures are extensions of the usual vector space structure, in the sense that for all $c \in F$ and $\alpha \in V$, we have $c \odot \alpha = c\alpha$.

Now, for fixed $\tau \in \mathcal{L}_F(V)$ and $\alpha \in V$, consider the $F[X]$-linear map $\rho_{\tau,\alpha} : F[X] \to V$ that sends $g \in F[X]$ to $g \odot \alpha = g(\tau)(\alpha)$. The kernel of this map must be a submodule, and hence an ideal, of $F[X]$; since every ideal of $F[X]$ is principal, it follows that $\operatorname{Ker} \rho_{\tau,\alpha}$ is the ideal of $F[X]$ generated by some polynomial $\phi_{\tau,\alpha}$, which we can make unique by insisting that it is monic or zero. We call $\phi_{\tau,\alpha}$ the **minimal polynomial of α under τ**. We can also characterize $\phi_{\tau,\alpha}$ as follows:

> if there exists a non-zero polynomial $g \in F[X]$ such that $g(\tau)(\alpha) = 0$, then $\phi_{\tau,\alpha}$ the monic polynomial of least degree with this property; otherwise, $\phi_{\tau,\alpha} = 0$.

Another way to characterize $\phi_{\tau,\alpha}$ is as follows:

> $\phi_{\tau,\alpha}$ is the minimal polynomial of the sequence $\{\tau^i(\alpha)\}_{i=0}^{\infty}$.

Note that since $\phi_\tau(\tau)$ is the zero map, we have

$$\phi_\tau \odot \alpha = \phi_\tau(\tau)(\alpha) = 0,$$

and hence $\phi_\tau \in \operatorname{Ker} \rho_{\tau,\alpha}$, which means that $\phi_{\tau,\alpha} \mid \phi_\tau$.

Now consider the image of $\rho_{\tau,\alpha}$, which we shall denote by $\langle \alpha \rangle_\tau$. As an $F[X]$-module, $\langle \alpha \rangle_\tau$ is isomorphic to $F[X]/(\phi_{\tau,\alpha})$. In particular, if $\phi_{\tau,\alpha}$ is non-zero and has degree m, then $\langle \alpha \rangle_\tau$ is a vector space of dimension m over F; indeed, the elements $\alpha, \tau(\alpha), \ldots, \tau^{m-1}(\alpha)$ form a basis for $\langle \alpha \rangle_\tau$ over F; moreover, m is the smallest non-negative integer such that $\{\tau^i(\alpha)\}_{i=0}^{m}$ is linearly dependent.

Observe that for every $\beta \in \langle \alpha \rangle_\tau$, we have $\phi_{\tau,\alpha} \odot \beta = 0$; indeed, if $\beta = g \odot \alpha$, then

$$\phi_{\tau,\alpha} \odot (g \odot \alpha) = (\phi_{\tau,\alpha} g) \odot \alpha = g \odot (\phi_{\tau,\alpha} \odot \alpha) = g \odot 0 = 0.$$

The following three theorems develop some simple facts; the proofs of these are straightforward, and left as exercises. In each theorem, τ is an element of $\mathcal{L}_F(V)$, and $\odot$ is the associated scalar multiplication that makes V into an $F[X]$-module.

Theorem 18.9. *Let $\alpha \in V$ have minimal polynomial $f \in F[X]$ under τ, and let $\beta \in V$ have minimal polynomial $g \in F[X]$ under τ. If $\gcd(f, g) = 1$, then $\langle \alpha \rangle_\tau \cap \langle \beta \rangle_\tau = \{0\}$, and $\alpha + \beta$ has minimal polynomial $f \cdot g$ under τ.*

Theorem 18.10. *Let $\alpha \in V$. Let $f \in F[X]$ be a monic irreducible polynomial such that $f^e \odot \alpha = 0$ but $f^{e-1} \odot \alpha \neq 0$ for some integer $e \geq 1$. Then f^e is the minimal polynomial of α under τ.*

Theorem 18.11. *Let $\alpha \in V$, and suppose that α has minimal polynomial $f \in F[X]$ under τ, with $f \neq 0$. Let $g \in F[X]$. Then $g \odot \alpha$ has minimal polynomial $f / \gcd(f, g)$ under τ.*

We are now ready to state the main result of this section, whose statement and proof are analogous to that of Theorem 6.41:

Theorem 18.12. *Let $\tau \in \mathcal{L}_F(V)$, and suppose that τ has non-zero minimal polynomial ϕ. Then there exists $\beta \in V$ such that the minimal polynomial of β under τ is ϕ.*

Proof. Let $\odot$ be the scalar multiplication associated with τ. Let $\phi = \phi_1^{e_1} \cdots \phi_r^{e_r}$ be the factorization of ϕ into monic irreducible polynomials in $F[X]$.

First, we claim that for each $i = 1, \ldots, r$, there exists $\alpha_i \in V$ such that $\phi/\phi_i \odot \alpha_i \neq 0$. Suppose the claim were false: then for some i, we would have $\phi/\phi_i \odot \alpha = 0$ for all $\alpha \in V$; however, this means that $(\phi/\phi_i)(\tau) = 0$, contradicting the minimality property in the definition of the minimal polynomial ϕ. That proves the claim.

Let $\alpha_1, \ldots, \alpha_r$ be as in the above claim. Then by Theorem 18.10, each $\phi/\phi_i^{e_i} \odot \alpha_i$ has minimal polynomial $\phi_i^{e_i}$ under τ. Finally, by Theorem 18.9,

$$\beta := \phi/\phi_1^{e_1} \odot \alpha_1 + \cdots + \phi/\phi_r^{e_r} \odot \alpha_r$$

has minimal polynomial ϕ under τ. $\square$

Theorem 18.12 says that if τ has minimal polynomial ϕ of degree $m \geq 0$, then there exists $\beta \in V$ such that $\{\tau^i(\beta)\}_{i=0}^{m-1}$ is linearly independent. From this, it immediately follows that:

Theorem 18.13. *If V has finite dimension $\ell > 0$, then for every $\tau \in \mathcal{L}_F(V)$, the minimal polynomial of τ is non-zero of degree at most ℓ.*

We close this section with a simple observation. Let V be an arbitrary $F[X]$-module with scalar multiplication $\odot$. Restricting the scalar multiplication from $F[X]$ to F, we can naturally view V as an F-vector space. Let $\tau : V \to V$ be the map that sends $\alpha \in V$ to $X \odot \alpha$. It is easy to see that $\tau \in \mathcal{L}_F(V)$, and that for all polynomials $g \in F[X]$, and all $\alpha \in V$, we have $g \odot \alpha = g(\tau)(\alpha)$. Thus, instead of starting with a vector space and defining an $F[X]$-module structure in terms of a given linear map, we can go the other direction, starting from an $F[X]$-module and obtaining a corresponding linear map. Furthermore, using the language introduced in Examples 13.19 and 13.20, we see that the $F[X]$-exponent of V is the ideal of $F[X]$ generated by the minimal polynomial of τ, and the $F[X]$-order of any element $\alpha \in V$ is the ideal of $F[X]$ generated by the minimal polynomial of α under τ. Theorem 18.12 says that there exists an element in V whose $F[X]$-order is equal to the $F[X]$-exponent of V, assuming the latter is non-zero.

So depending on one's mood, one can place emphasis either on the linear map τ, or just talk about $F[X]$-modules without mentioning any linear maps.

EXERCISE 18.11. Let $\tau \in \mathcal{L}_F(V)$ have non-zero minimal polynomial ϕ of degree m, and let $\phi = \phi_1^{e_1} \cdots \phi_r^{e_r}$ be the factorization of ϕ into monic irreducible polynomials in $F[X]$. Let $\odot$ be the scalar multiplication associated with τ. Show that $\beta \in V$ has minimal polynomial ϕ under τ if and only if $\phi/\phi_i \odot \beta \neq 0$ for $i = 1, \ldots, r$.

EXERCISE 18.12. Let $\tau \in \mathcal{L}_F(V)$ have non-zero minimal polynomial ϕ. Show that τ is bijective if and only if $X \nmid \phi$.

EXERCISE 18.13. Let F be a finite field, and let V have finite dimension $\ell > 0$ over F. Let $\tau \in \mathcal{L}_F(V)$ have minimal polynomial ϕ, with $\deg(\phi) = m$ (and of course, by Theorem 18.13, we have $m \le \ell$). Suppose that $\alpha_1, \ldots, \alpha_s$ are randomly chosen elements of V. Let g_j be the minimal polynomial of α_j under τ, for $j = 1, \ldots, s$. Let Q be the probability that $\mathrm{lcm}(g_1, \ldots, g_s) = \phi$. The goal of this exercise is to show that $Q \ge \Lambda_F^{\phi}(s)$, where $\Lambda_F^{\phi}(s)$ is as defined in §18.3.

(a) Using Theorem 18.12 and Theorem 18.11, show that if $m = \ell$, then $Q = \Lambda_F^{\phi}(s)$.

(b) Without the assumption that $m = \ell$, things are a bit more challenging. Adopting the matrix-oriented point of view discussed at the end of §18.3, and transposing everything, show that

- there exists $\pi \in \mathcal{D}_F(V)$ such that the sequence $\{\pi \circ \tau^i\}_{i=0}^{\infty}$ has minimal polynomial ϕ, and
- if, for $j = 1, \ldots, s$, we define h_j to be the minimal polynomial of the sequence $\{\pi(\tau^i(\alpha_j))\}_{i=0}^{\infty}$, then the probability that $\mathrm{lcm}(h_1, \ldots, h_s) = \phi$ is equal to $\Lambda_F^{\phi}(s)$.

(c) Show that $h_j \mid g_j$, for $j = 1, \ldots, s$, and conclude that $Q \ge \Lambda_F^{\phi}(s)$.

EXERCISE 18.14. Let $f, g \in F[X]$ with $f \neq 0$, and let $h := f/\gcd(f, g)$. Show that $g \cdot F[X]/(f)$ and $F[X]/(h)$ are isomorphic as $F[X]$-modules.

EXERCISE 18.15. In this exercise, you are to derive the **fundamental theorem of finite dimensional $F[X]$-modules**, which is completely analogous to the fundamental theorem of finite abelian groups. Both of these results are really special cases of a more general decomposition theorem for modules over a principal ideal domain. Let V be an $F[X]$-module. Assume that as an F-vector space, V has finite dimension $\ell > 0$, and that the $F[X]$-exponent of V is generated by the monic polynomial $\phi \in F[X]$ (note that $1 \le \deg(\phi) \le \ell$). Show that there exist monic, non-constant polynomials $\phi_1, \ldots, \phi_t \in F[X]$ such that

- $\phi_i \mid \phi_{i+1}$ for $i = 1, \ldots, t-1$, and
- V is isomorphic, as an $F[X]$-module, to the direct product of $F[X]$-modules

$$V' := F[X]/(\phi_1) \times \cdots \times F[X]/(\phi_t).$$

Moreover, show that the polynomials $\phi_1, \ldots, \phi_t$ satisfying these conditions are uniquely determined, and that $\phi_t = \phi$. Hint: one can just mimic the proof of Theorem 6.45, where the exponent of a group corresponds to the $F[X]$-exponent of an $F[X]$-module, and the order of a group element corresponds to the $F[X]$-order of an element of an $F[X]$-module—everything translates rather directly, with just a few minor, technical differences, and the previous exercise is useful in proving the uniqueness part of the theorem.

EXERCISE 18.16. Let us adopt the same assumptions and notation as in Exercise 18.15, and let $\tau \in \mathcal{L}_F(V)$ be the map that sends $\alpha \in V$ to $X \odot \alpha$. Further, let $\sigma : V \to V'$ be the isomorphism of that exercise, and let $\tau' \in \mathcal{L}_F(V')$ be the X-multiplication map on V'.

(a) Show that $\sigma \circ \tau = \tau' \circ \sigma$.

(b) From part (a), derive the following: there exists a basis for V over F, with respect to which the matrix of τ is the "block diagonal" matrix

$$T = \begin{pmatrix} C_1 & & & \\ & C_2 & & \\ & & \ddots & \\ & & & C_t \end{pmatrix},$$

where each C_i is the companion matrix of ϕ_i (see Example 14.1).

EXERCISE 18.17. Let us adopt the same assumptions and notation as in Exercise 18.15.

(a) Using the result of that exercise, show that V is isomorphic, as an $F[X]$-module, to a direct product of $F[X]$-modules

$$F[X]/(f_1^{e_1}) \times \cdots \times F[X]/(f_r^{e_r}),$$

where the f_i's are monic irreducible polynomials (not necessarily distinct) and the e_i's are positive integers, and this direct product is unique up to the order of the factors.

(b) Using part (a), show that there exists a basis for V over F, with respect to which the matrix of τ is the "block diagonal" matrix

$$T' = \begin{pmatrix} C'_1 & & & \\ & C'_2 & & \\ & & \ddots & \\ & & & C'_r \end{pmatrix},$$

where each C'_i is the companion matrix of $f_i^{e_i}$.

EXERCISE 18.18. Let us adopt the same assumptions and notation as in Exercise 18.15.

(a) Suppose $\alpha \in V$ corresponds to $([g_1]_{\phi_1}, \ldots, [g_t]_{\phi_t}) \in V'$ under the isomorphism of that exercise. Show that the $F[X]$-order of α is generated by the polynomial

$$\operatorname{lcm}(\phi_1 / \gcd(g_1, \phi_1), \ldots, \phi_t / \gcd(g_t, \phi_t)).$$

(b) Using part (a), give a short and simple proof of the result of Exercise 18.13.

18.7 Notes

Berlekamp [15] and Massey [64] discuss an algorithm for finding the minimal polynomial of a linearly generated sequence that is closely related to the one presented in §18.2, and which has a similar complexity. This connection between Euclid's algorithm and finding minimal polynomials of linearly generated sequences has been observed by many authors, including Mills [68], Welch and Scholtz [108], and Dornstetter [36].

The algorithm presented in §18.3 is due to Wiedemann [109], as are the algorithms for solving sparse linear systems in §18.4, as well as the statement and proof outline of the result in Exercise 18.13.

Our proof of Theorem 18.4 is based on an exposition by Morrison [69].

Using fast matrix and polynomial arithmetic, Shoup [96] shows how to implement the algorithms in §18.5 so as to use just $O(\ell^{(\omega+1)/2})$ operations in F, where ω is the exponent for matrix multiplication (see §14.6), and so $(\omega + 1)/2 < 1.7$.†

† The running times of these algorithms can be improved using faster algorithms for modular composition—see footnote on p. 485.