Chapter 6

Polynomials and Codes

This final chapter brings together three ways we can use polynomials over finite fields to make linear codes.

- Many (but not all) of you saw in Algebra II how to construct non-prime finite fields from rings of polynomials. In these notes we shall summarise these ideas, and set up some small non-prime fields which we can actually use for making codes, using the techniques of earlier chapters.
- Alternatively, using almost the same process we can make cyclic codes instead of fields.
- Finally, using the polynomials in a very different way, we can make Reed-Solomon codes.

6.1 Polynomials over $\mathbb{F}_q$

Just as we have the familiar polynomials with coefficients in $\mathbb{Z}$, we can also make polynomials with coefficients in $\mathbb{F}_q$.

Definition 6.1. The **ring of polynomials over** $\mathbb{F}_q$ is

$$\mathbb{F}_q[x] = \{f(x) = a_0 + a_1x + \ldots + a_dx^d \mid d \geq 0, a_i \in \mathbb{F}_q\}.$$

Then if $a_d \neq 0$, we say d is the degree of $f(x)$, and we can add and multiply as usual, but always reducing the coefficients mod q.

Example 37. In $\mathbb{F}_5[x]$, let $p(x) = 4x + 3$ and $q(x) = 3x^2 + 2x + 1$.
Then $p(x) + q(x) = 3x^2 + x + 4$ and $p(x)q(x) = 2x^3 + 2x^2 + 3$. △

But if we are looking for more finite fields, this does not seem to help much: $\mathbb{F}_q[x]$ is infinite, and it is a ring not a field, because most $f(x)$ have no multiplicative inverse.

We can make $\mathbb{F}_q[x]$ finite simply by restricting degree.

Definition 6.2. Let

$$\mathbb{F}_q[x]_{<k} = \{f(x) = a_0 + a_1x + \ldots + a_dx^d \mid 0 \le d < k, a_i \in \mathbb{F}_q\},$$

which, as we can always add a few terms with zero coefficients, is the same as

$$\mathbb{F}_q[x]_{<k} = \{f(x) = a_0 + a_1x + \ldots + a_{k-1}x^{k-1} \mid a_i \in \mathbb{F}_q\}.$$

But then $\mathbb{F}_q[x]_{<k}$ is not even a ring, as it is not closed under multiplication. It is, however, closed under addition, and under multiplication by a scalar from $\mathbb{F}_q$. In fact, it is a vector space, of dimension k over $\mathbb{F}_q$. It is isomorphic to $\mathbb{F}_q^k$ by the obvious map $\phi(f(x)) = (a_0, \ldots, a_{k-1})$. In section 6.5 we'll use this vector space of polynomials to construct Reed-Solomon codes.

To make more finite fields, we need a different approach. Recall that $\mathbb{Z}$ is also an infinite ring, but in Chapter 2 we made $\mathbb{Z}/n$ by regarding n as zero, and identifying any $m \in \mathbb{Z}$ with its remainder mod n. Then $\mathbb{Z}/n$ is always finite, and may be either a ring (e.g. $\mathbb{Z}/6$) or a field (e.g. $\mathbb{Z}/5$). Similarly, we shall now make $\mathbb{F}_q[x]/(f(x))$,[1] by regarding $f(x)$ as zero, and replacing any $g(x)$ with $r(x)$, where $g(x) = q(x)f(x) + r(x)$, and $\deg(r(x)) < \deg(f(x))$.

In $\mathbb{Z}/n$ the elements were really equivalence classes, $\overline{r} = \{r + qn \mid q \in \mathbb{Z}\}$. Similarly, in $\mathbb{F}_q[x]/(f(x))$ we have elements $\overline{r(x)} = \{r(x) + q(x)f(x) \mid q(x) \in \mathbb{F}_q[x]\}$. Again, we shall drop the overline, for convenience.

Suppose that $\deg(f(x)) = d$. Then as $0 \le \deg(r(x)) < d$ we know that $\mid \mathbb{F}_q[x]/(f(x)) \mid = q^d$. Let us now investigate the smallest possible cases, with $q = 2, d = 2$.

Example 38. Consider $\mathbb{F}_2[x]/(f(x))$, where $\deg(f(x)) = 2$.
Then $\mathbb{F}_2[x]/(f(x)) = \{0, 1, x, x + 1\}$, and its addition table is:

$+$	0	1	x	$x+1$
0	0	1	x	$x+1$
1	1	0	$x+1$	x
x	x	$x+1$	0	1
$x+1$	$x+1$	x	1	0

But for its multiplication table, we have to know $f(x)$. In $\mathbb{F}_2[x]/(x^2 + 1)$, we have $x^2 + 1 = 0$, so $x^2 = 1$. Also, $(x + 1)^2 = x^2 + 2x + 1 = 0$, and $x(x + 1) = x^2 + x = x + 1$. So we have:

$\times$	1	x	$x+1$
1	1	x	$x+1$
x	x	1	$x+1$
$x+1$	$x+1$	$x+1$	0

Since $x + 1$ has no multiplicative inverse, $\mathbb{F}_2[x]/(x^2 + 1)$ is a ring but not a field.

[1] Why the extra brackets round $f(x)$, which we did not put round n? One reason is that, in LaTeX, $\mathbb{F}_q[x]/x^2+x+1$ is not as clear as it can be on a board: are we dividing out just by the x^2? Another reason, for those of you who did Algebra II, is that $(f(x))$ is the notation for the *ideal* $\{q(x)f(x) \mid q(x) \in F_q[x]\}$, and in fact that is exactly what we are dividing out by (regarding as 0).

However, in $\mathbb{F}_2[x]/(x^2+x+1)$, we have $x^2+x+1=0$, so $x^2=x+1$. Then $x(x+1) = x^2+x = x+1+x = 1$, and $(x+1)^2 = x^2+1 = x+1+1 = x$. We get:

$\times$	1	x	$x+1$
1	1	x	$x+1$
x	x	$x+1$	1
$x+1$	$x+1$	1	x

So this is a field, with 4 elements. We call it $\mathbb{F}_4$. The difference is that, over $\mathbb{F}_2$, x^2+1 factors as $(x+1)(x+1)$, so $(x+1)$ is a zero-divisor and has no inverse. (See Q78) But x^2+x+1 does not factor over $\mathbb{F}_2$. △

Definition 6.3. Let $f(x)$ be a polynomial in $\mathbb{F}_q[x]$, of degree d. Then if $f(x) = p(x)q(x)$, with both $p(x)$ and $q(x)$ of degree $< d$, we say $f(x)$ is **reducible**. Otherwise it is **irreducible**.

Although we shall not prove it formally, the following proposition is suggested by the examples above.

Proposition 6.4. *Let $\mathbb{F}_q$ be a field, and $f(x)$ a polynomial in $\mathbb{F}_q[x]$. If $f(x)$ is irreducible in $\mathbb{F}_q[x]$, then $\mathbb{F}_q[x]/(f(x))$ is a field; otherwise it is a ring.*

In Section 6.2 we shall use irreducible $f(x)$ to make new finite fields; in 6.3 we shall use certain particular reducible $f(x)$ to make cyclic codes.

6.2 Non-prime Finite Fields

We shall now construct $\mathbb{F}_9$ and $\mathbb{F}_8$, in much the same way as we did $\mathbb{F}_4$. But to help us in choosing $f(x)$, and to be able to do arithmetic without large tables, we need a couple more ideas.

Definition 6.5. A polynomial $f(x) = a_d x^d + \cdots + a_1 x + a_0$ is **monic** if $a_d = 1$.

Proposition 6.6. *If $f(x) = \lambda m(x) \in \mathbb{F}_q[x]$, with $\lambda \in \mathbb{F}_q$, then $\mathbb{F}_q[x]/(f(x)) = \mathbb{F}_q[x]/(m(x))$.*

Proof. In $\mathbb{F}_q[x]$, we have $g(x) = q(x)f(x) + r(x)$ if and only if $g(x) = (\lambda q(x))m(x) + r(x)$. Thus in both $\mathbb{F}_q[x]/(f(x))$ and $\mathbb{F}_q[x]/(m(x))$, $g(x)$ is represented by the same remainder $r(x)$. □

It follows that we need only consider monic polynomials as possible $f(x)$.

Definition 6.7. In a finite field $\mathbb{F}_q$, an element is **primitive** if its powers give us all of $\mathbb{F}_q \backslash \{0\}$.

Example 39. In $\mathbb{F}_7$, powers of 3 are $1,\ 3,\ 9=2,\ 6,\ 18=4,\ 12=5,\ 15=1$. But powers of 2 are $1,\ 2,\ 4,\ 8=1$. So 3 is primitive in $\mathbb{F}_7$, but 2 is not. △

It is a fact, which we shall not prove, that every finite field has at least one primitive element. (See also Q81 and Q86.)

We are now ready to construct $\mathbb{F}_9$. Since $9 = 3^2$, we need $\mathbb{F}_3[x]/(f(x))$, where $f(x) = a_2x^2 + a_1x + a_0$ is of degree 2, monic, and irreducible. Thus $a_2 = 1$, and $a_0 \neq 0$ (or $f(x)$ would have a factor x). Of the six remaining possibilities, we can find which are reducible by calculating in $\mathbb{F}_3[x]$:

$$\begin{aligned}(x+1)(x+1) &= x^2+2x+1,\\ (x+1)(x+2) &= x^2+3x+2 = x^2+2,\\ (x+2)(x+2) &= x^2+4x+4 = x^2+x+1.\end{aligned}$$

It follows that x^2+1, x^2+x+2, and x^2+2x+2 are irreducible in $\mathbb{F}_3[x]$; we shall use $f(x) = x^2+x+2$. Adding in $\mathbb{F}_3[x]/(x^2+x+2)$ is easy; we just reduce coefficients mod 3. For multiplication, instead of making a full 8×8 table, we need only calculate the powers of x. To do this, note first that since $x^2+x+2 = 0$, we have $x^2 = -x-2 = 2x+1$. Then

$$x^3 = x(2x+1) = 2x^2 + x = 2(2x+1) + x = 5x + 2 = 2x + 2.$$

We can calculate x^4 as either $(x^2)^2 = (2x+1)^2$ or as $x \cdot x^3 = x(2x+2)$; either way we get 2. Then the rest is easy: $x^i = 2x^{i-4}$, and we have the following table:

i	0	1	2	3	4	5	6	7	8
x^i	1	x	$2x+1$	$2x+2$	2	$2x$	$x+2$	$x+1$	1

Note that we do indeed have every non-zero element of $\mathbb{F}_3[x]/(x^2+x+2)$, so x is primitive in $\mathbb{F}_3[x]/(x^2+x+2)$; if it were not, we would have got back to 1 too soon (see Q84) It is also clear that, for higher powers of $x, x^i = x^{i-8}$.

For $\mathbb{F}_8$, similarly, we want $\mathbb{F}_2[x]/(x^3+a_2x^2+a_1x+1)$ but should not use the reducible $x^3+1 = (x+1)(x^2+x+1)$ or $x^3+x^2+x+1 = (x+1)(x^2+1)$. We choose $f(x) = x^3+x+1$ and so $x^3 = x+1$, and the table is

i	0	1	2	3	4	5	6	7
x^i	1	x	x^2	$x+1$	x^2+x	x^2+x+1	x^2+1	1

We see that for higher powers of $x, x^i = x^{i-7}$, and that again x is primitive in $\mathbb{F}_2[x]/(x^3+x+1)$.

Definition 6.8. If x is primitive in $\mathbb{F}_q[x]/(f(x))$ we say that $f(x)$ is a **primitive polynomial** over $\mathbb{F}_q$.

For doing arithmetic in $\mathbb{F}_q$, it is very useful that every non-zero element can be written both as x^i and as $a_{r-1}x^{r-1} + \cdots + a_0$. Powers of x are easy to multiply, and can be reduced mod $q-1$. The short polynomials are easy to add, and then we reduce the coefficients mod p. And we can use our table of powers to convert easily between the two.

Example 40. In $\mathbb{F}_8 = \mathbb{F}_2[x]/(x^3+x+1)$ as above,

$$\begin{aligned} x^3 + x^4 &= (x+1) + (x^2+x) = x^2 + 1 = x^6 \\ (x+1)(x^2+x+1) &= x^3 \cdot x^5 = x^8 = x^7 \cdot x^1 = x \\ (x+1)^{-1} &= (x^3)^{-1} = x^4 = x^2 + x, \text{ since } x^3 \cdot x^4 = x^7 = 1 \end{aligned}$$

△

We now *leave behind* our rule (Ch. 2) that q is always prime. We can then generalize the two constructions above in the following proposition (although we will not formally prove it):

Proposition 6.9. *Let $q = p^r$, where p is prime, and $r \geq 2$ an integer, and let $f(x) \in \mathbb{F}_p[x]$ be monic, irreducible, and of degree r. Then $\mathbb{F}_p[x]/(f(x))$ is a field, $\mathbb{F}_q$.*

Choosing a different such $f(x)$ will give a field where multiplication looks different, but in fact the two fields will be isomorphic. For convenience, we choose a primitive $f(x)$.

Now that we can do arithmetic, we can make vector spaces and codes over these new $\mathbb{F}_q$. Q87-92 give you a chance to try out all the usual methods with these new fields.

6.3 Cyclic Codes

A cyclic code $C \subseteq \mathbb{F}_q^n$ is a linear code such that any cyclic shift of a codeword is also a codeword. This can be understood in terms of the permutation automorphism group of C, as saying the the group generated by the cyclic permutation of all n positions is a subgroup of the permutation automorphism group, $\langle(1,\ 2,\ \ldots,\ n)\rangle = \{(1,\ 2,\ \ldots,\ n)^i \mid 1 \leq i \leq n\} \subseteq \mathrm{PAut}(C)$ (those of you who studied Algebra II may recognise this as isomorphic the cyclic group of order n, $C_n = \langle g \mid g^n = e\rangle = \{e, g, g^2, \ldots g^{n-1}\}$).

Definition 6.10. A code C is **cyclic** if it is linear and

$$(a_0, a_1, a_2, \ldots, a_{n-1}) \in C \iff (a_{n-1}, a_0, a_1, \ldots, a_{n-2}) \in C.$$

Note that we now index the positions from 0 to $n-1$, rather than 1 to n.

Example 41. The code $C = \{(0,0,0,0), (0,1,0,1), (1,0,1,0), (1,1,1,1)\} \subseteq \mathbb{F}_2^n$ is cyclic. △

So, some of the codes we used in previous chapters were cyclic. But we shall now find a different way to think about cyclic codes, using polynomials over $\mathbb{F}_q$. (As we know, q can now be any prime power. But for our examples we shall stick to q prime - and small!)

Consider $\mathbb{F}_q/(x^n-1)$. Its elements are polynomials $a_0 + a_1x + \cdots + a_{n-1}x^{n-1}$. Over any $\mathbb{F}_q$, $x^n - 1 = (x-1)(x^{n-1} + \cdots + x + 1)$, so $x^n - 1$ is reducible. Thus $\mathbb{F}_q/(x^n-1)$ is a ring, not a field.

Notation: When it is clear what field we are using, we shall write simply $\mathbf{R}_n$ for $\mathbb{F}_q/(x^n-1)$. Note that this is $\mathbf{R}$ for ring, not $\mathbb{R}$ for the real numbers.

There is an obvious correspondence between polynomials in $\mathbf{R}_n$ and vectors in $\mathbb{F}_q^n$:

$$a(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1} \longleftrightarrow \mathbf{a} = (a_0, a_1, a_2, \ldots, a_{n-1}).$$

We also have:

Lemma 6.11. *If $a(x) \longleftrightarrow \mathbf{a}$ as above, then $x \cdot a(x) \longleftrightarrow (a_{n-1}, a_0, a_1, \ldots, a_{n-2})$, a cyclic shift of $\mathbf{a}$.*

Proof. In $\mathbf{R}_n$ we have $x^n - 1 = 0$, so $x^n = 1$. Then

$$x \cdot a(x) = a_0x + a_1x^2 + \cdots + a_{n-1}x^n = a_{n-1} + a_0x + a_1x^2 + \cdots + a_{n-2}x^{n-1}.$$

$\square$

From now on we shall think of words (and in particular codewords) as *both* vectors in $\mathbb{F}_q^n$, *and* polynomials in $\mathbf{R}_n$ over $\mathbb{F}_q$.

Proposition 6.12. *A code $C \subseteq \mathbf{R}_n$ is cyclic if and only if:*

i) $a(x), b(x) \in C \Longrightarrow a(x) + b(x) \in C$,

ii) $a(x) \in C$ *and* $r(x) \in \mathbf{R}_n \Longrightarrow r(x)a(x) \in C$,

This looks very like the definition of a linear code: we check closure for adding and for multiplying. But now we are multiplying not just by a scalar λ from $\mathbb{F}_q$; we are multiplying by any other polynomial in $\mathbf{R}_n$. This was not an option in $\mathbb{F}_q^n$. (You might also recognise that this proposition says that C is cyclic if and only if it is an *ideal* in the ring $\mathbf{R}_n$.)

Proof. $\Longrightarrow$: Suppose C is cyclic. Then C is linear so i) holds. For ii): Since any cyclic shift of $\mathbf{a} \in C$ is also in C, we know by Lemma 6.11 that for the corresponding $a(x) \in C$ in $\mathbf{R}_n$, $x \cdot a(x)$ is also in C. But then also $x^m \cdot a(x) \in C$ for any m. So if $r(x) = r_0 + r_1x + \cdots + r_{n-1}x^{n-1}$, then $r(x) \cdot a(x) = r_0a(x) + r_1x \cdot a(x) + \cdots + r_{n-1}x^{n-1} \cdot a(x)$ is also in C (as C is linear).

$\Longleftarrow$: By i), and ii) with $r(x)$ a scalar $r_o \in \mathbb{F}_q$, we know C is linear. By Lemma 6.11, any cyclic shift of $\mathbf{a} \in C$ corresponds to $x^m \cdot a(x)$; by ii) with $r(x) = x^m$ this is also in C. $\square$

We can adapt the 'span' notation for cyclic codes:

Definition 6.13. For $f(x) \in \mathbf{R}_n$, $\langle f(x) \rangle = \{a(x)f(x) \mid a(x) \in \mathbf{R}_n\}$, the code **generated by** $f(x)$.

Proposition 6.14. *For any $f(x) \in \mathbf{R}_n$, $\langle f(x) \rangle$ is a cyclic code.*

Proof. It is very easy to check properties i) and ii) of Proposition 6.12 $\square$

Example 42. Let us take x^2+1 in $\mathbf{R}_3 = \mathbb{F}_2[x]/(x^3-1)$, and calculate all its multiples. We have to reduce the powers mod 3 (because $x^3 = 1$), and also reduce coefficients mod 2. (Note that once we have the multiples of $1, x$, and x^2, we could instead get the others by adding.)

$r(x)$	$r(x)\cdot(x^2+1)$	
0	0	
1	x^2+1	
x	x^3+x	$=x+1$
$x+1$	x^3+x^2+x+1	$=x^2+x$
x^2	x^4+x^2	$=x^2+x$
x^2+1	x^4+2x^2+1	$=x+1$
x^2+x	$x^4+x^3+x^2+x$	$=x^2+1$
x^2+x+1	$x^4+x^3+2x^2+x+1$	$=0$

So

$$\begin{aligned}\langle f(x)\rangle &= \{0, 1+x, 1+x^2, x+x^2\} \subseteq \mathbf{R}_3 \\ &\longleftrightarrow \{(0,0,0),(1,1,0),(1,0,1),(0,1,1)\} \subseteq \mathbb{F}_2^3\end{aligned}$$

△

It turns out that any cyclic code can be made like this.

Theorem 6.15. *Let C be a cyclic code in $\mathbf{R}_n$ over $\mathbb{F}_q$, $\mathbb{C} \neq \{0\}$. Then:*

i) there is a unique polynomial $g(x)$, which is the monic polynomial of smallest degree in C.

ii) $C = \langle g(x)\rangle$.

iii) $g(x)$ is a factor of $x^n - 1$.

Proof. First note that if C contains a polynomial $r(x)$ of degree $d \geq 0$, then since C is linear it must also contain a *monic* polynomial of degree d, because we can always just multiply $r(x)$ by the right $\lambda \in \mathbb{F}_q$.

i) Clearly there is a monic polynomial of smallest degree. Suppose there are two such, $g(x)$ and $h(x)$. Then let $r(x) = g(x) - h(x) \in C$. Since the terms of highest degree will cancel, $r(x)$ has smaller degree. So we have a contradiction.

Having proved i) it also follows that if $r(x) \in C$ has degree less than that of $g(x)$, then $r(x)$ must in fact be 0 (which is regarded as having degree $-\infty$, but is not monic). We use this idea in many proofs.

ii) Clearly $\langle g(x)\rangle \subseteq C$. Now suppose $a(x) \in C$. In $\mathbb{F}_q[x]$, we can always write $a(x) = q(x)g(x) + r(x)$, where $\deg(r(x)) \leq \deg(g(x))$. But then, in $\mathbb{F}_q[x]$ and in $\mathbf{R}_n$ also, $r(x) = a(x) - q(x)g(x)$, so $r(x) \in C$, so $r(x) = 0$, and $a(x) = q(x)g(x) \in \langle g(x)\rangle$.

iii) In $\mathbb{F}_q[x]$, we can write $x^n - 1 = q(x)g(x) + r(x)$, where $\deg(r(x)) < \deg(g(x))$. But in $\mathbf{R}_n$, $x^n - 1 = 0$, so $r(x) = -q(x)g(x) \in \langle g(x) \rangle \subseteq C$. So again $r(x) = 0$, and $x^n - 1 = q(x)g(x)$.

□

Definition 6.16. In a cyclic code C, the monic polynomial of least degree is the **generator-polynomial** of C.

Example 43. In the example above, $C = \langle x^2+1 \rangle \subseteq \mathbf{R}_3$. But also, by Theorem 6.15 part ii), $C = \langle x+1 \rangle$. Although x^2+1 also generates C, $x+1$ is C's generator-polynomial. △

Theorem 6.15 part ii) says that every cyclic code is generated by a single polynomial. (In terms of ring theory, it is not just an ideal, but a *principal* ideal.) Part i) says that this generator-polynomial is unique.[2] Part iii) says that every cyclic code's generator-polynomial is a factor of $x^n - 1$. In fact, the converse is also true: every monic factor $g(x)$ of $x^n - 1$ is the unique generator-polynomial of the cyclic code $\langle g(x) \rangle$. (see Q94) It follows that distinct factors generate distinct codes. So we have a way to actually find all the cyclic codes in $\mathbf{R}_n$: take each (monic) divisor of $x^n - 1$ in turn in as the generator-polynomial $g(x)$.

Example 44. To find all binary cyclic codes of block-length 3, we consider $\mathbf{R}_3 = \mathbb{F}_2[x]/(x^3 - 1)$. In $\mathbb{F}_2[x]$, $x^3 - 1 = (x+1)(x^2 + x + 1)$, and $x^2 + x + 1$ is irreducible. So we have four divisors, and four codes, but there is not much work to do: we have already worked out $\langle x+1 \rangle$, and for $\langle x^2 + x + 1 \rangle$ notice that $x(x^2 + x + 1) = x^2 + x + 1$.

generator	code in $\mathbf{R}_3$	code in $\mathbb{F}_2^3$
1	all of $\mathbf{R}_3$	all of $\mathbb{F}_2^3$
$x+1$	$\{0, 1+x, 1+x^2, x+x^2\}$	$\{(0,0,0), (1,1,0), (1,0,1), (0,1,1)\}$
x^2+x+1	$\{0, 1+x+x^2\}$	$\{(0,0,0), (1,1,1)\}$
x^3-1	$\{0\}$	$\{(0,0,0)\}$

You will recognise our very first code, yet again. △

6.4 Matrices for Cyclic Codes

Since our cyclic codes live in $\mathbb{F}_q^n$ as well as in $\mathbf{R}_n$, we should be able to find generator- and check-matrices for them.

Proposition 6.17. *If C is a cyclic code with generator-polynomial $g(x) = g_0 + g_1 x + \cdots + g_r x^r$, then $\dim(C) = n - r$, and C has generator-matrix*

$$G = \begin{pmatrix} g_0 & g_1 & \cdots & g_r & & & & \\ & g_0 & g_1 & \cdots & g_r & & 0 & \\ & & g_0 & g_1 & \cdots & g_r & & \\ & 0 & & \ddots & \ddots & & \ddots & \\ & & & & g_0 & g_1 & \cdots & g_r \end{pmatrix} \in M_{n-r,n}(\mathbb{F}_q).$$

[2] In contrast, a code usually has many different generator-*matrices*.

Proof. We need to show that the rows of G are a basis for C. For linear independence, we note first that, as factors of x^n-1, any generator-polynomial must have $g_0 \neq 0$. Also, although G is not in RREF, it is in echelon form, and the echelon of non-zero g_0s ensures that the rows are linearly independent.[3]

We must also show that the rows of G (in $\mathbb{F}_q^n$) generate every codeword in C. In $\mathbf{R}_n$, they correspond to the polynomials $g(x), xg(x), \ldots, x^{n-r-1}g(x)$. Now suppose $a(x) \in C \subseteq \mathbf{R}_n$. In the proof of Theorem 6.15 ii) we showed that, in $\mathbb{F}_q[x]$, $a(x) = q(x)g(x)$, which is a linear combination of just such $x^i g(x)$. We only need to check that the degrees work out. Since $\deg(a(x)) \leq n-1$, and $\deg(g(x)) = r$, we must have $\deg(q(x)) \leq n-r-1$. Thus $a(x) = q(x)g(x) = q_0 g(x) + q_1 xg(x) + \cdots + q_{n-r-1}x^{n-r-1}g(x)$, and this is exactly a linear combination of rows of G, as required. □

Example 45. Let us find generator-matrices for all ternary cyclic codes of block-length 4, in $\mathbf{R}_4 = \mathbb{F}_3[x]/(x^4-1)$. First we must factor x^4-1 into irreducible polynomials: $x^4-1 = (x^2-1)(x^2+1) = (x-1)(x+1)(x^2+1)$. There are $2^3 = 8$ products of these factors.

	generator polynomial	generator matrix		generator polynomial	generator matrix
C_1	1	$\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{pmatrix}$	C_5	$x^4-1=0$	$\begin{pmatrix}0&0&0&0\end{pmatrix}$
C_2	$x+1$	$\begin{pmatrix}1&1&0&0\\0&1&1&0\\0&0&1&1\end{pmatrix}$	C_6	$x-1$	$\begin{pmatrix}-1&1&0&0\\0&-1&1&0\\0&0&-1&1\end{pmatrix}$
C_3	x^2+1	$\begin{pmatrix}1&0&1&0\\0&1&0&1\end{pmatrix}$	C_7	$(x-1)(x+1)=x^2-1$	$\begin{pmatrix}-1&0&1&0\\0&-1&0&1\end{pmatrix}$
C_4	$(x+1)(x^2+1)$ $= x^3+x^2+x+1$	$\begin{pmatrix}1&1&1&1\end{pmatrix}$	C_8	$(x-1)(x^2+1)$ $= x^3-x^2+x-1$	$\begin{pmatrix}-1&1&-1&1\end{pmatrix}$

△

Codes C_1 and C_5 are trivial, and in fact they are dual to each other. (See also Q95.)

To find check-matrices for the other six codes, we can easily row-reduce the generator-matrix to a standard form and then use Proposition 4.5. Alternatively, using Theorem 6.15 iii) we can define a check-polynomial for a code, and then make a check-matrix using that:

Definition 6.18. Let the cyclic $[n,k]$-code $C \subseteq \mathbf{R}_n$ have generator-polynomial $g(x)$, and $g(x)h(x) = x^n-1$ in $\mathbb{F}_q[x]$. Then $h(x)$ is the **check-polynomial** of C.

Lemma 6.19. *The check-polynomial $h(x)$ of a cyclic $[n,k]$-code is monic, of degree k.*

Proof. Since $x^n-1 = g(x)h(x)$ and $g(x)$ is monic, $h(x)$ must be monic also. By Proposition 6.17, if $\deg(g(x)) = r$, then $k = n-r$. But $\deg(g(x))+\deg(h(x)) = n$, so $\deg(h(x)) = k$. □

[3] If $\Sigma\lambda_i \mathrm{row}_i = 0$, then we must have $\lambda_1 = 0$, so $\lambda_2 = 0$, ...

Just as the generator-polynomial generates a cyclic code in $\mathbf{R}_n$, so the check-polynomial can be used to check whether a polynomial is in the code or not.

Proposition 6.20. *Let C be a cyclic code in $\mathbf{R}_n$, with check-polynomial $h(x)$. Then $c(x) \in C$ if and only if $c(x)h(x) = 0$, the zero-polynomial.*

Proof. First, let $g(x)$ be the generator-polynomial for C, so $g(x)h(x) = 0$ in $\mathbf{R}_n$.

$\Longrightarrow$ If $c(x) \in C$, then $c(x) = a(x)g(x)$ for some $a(x) \in \mathbf{R}_n$. But then $c(x)h(x) = a(x)g(x)h(x) = a(x) \cdot 0 = 0$.

$\Longleftarrow$ We know that in $\mathbb{F}_q[x]$, any $c(x) = q(x)g(x) + r(x)$, with $\deg(r(x)) < \deg\ (g(x)) = n - k$. Then if in $\mathbf{R}_n$ we have $c(x)h(x) = 0$, we know $q(x)g(x)h(x) + r(x)h(x) = 0$. But since $g(x)h(x) = x^n - 1 = 0$, it follows that $r(x)h(x) = 0$ in $\mathbf{R}_n$. In $\mathbb{F}_q[x]$, this tells us only that $r(x)h(x)$ is a multiple of $x^n - 1$. But $\deg(r(x)h(x)) < (n - k) + k = n$, so $r(x)h(x)$ is in fact 0 in $\mathbb{F}_q[x]$. Since this ring has no zero-divisors,[4] and $h(x) \neq 0$, we do have $r(x) = 0$. So $c(x) = a(x)g(x)$ as required. $\square$

Suppose now that we make a matrix H from $h(x)$ just as we made a generator-matrix G for C from $g(x)$. Is H a check-matrix for C? If it is, then H is also a generator-matrix for the dual code $C^\perp$, and so $h(x)$ is the generator-polynomial for $C^\perp$. Unfortunately, the truth is **not** quite that simple!

Definition 6.21. Let $h(x) = h_0 + h_1 x + \cdots + h_k x^k$. Then the **reciprocal polynomial** of $h(x)$ is $\overline{h}(x) = h_k + h_{k-1}x + \cdots + h_0 x^k$.

Can we say that $\overline{h}(x) = x^k h(x^{-1})$? In $\mathbb{F}_q[x]$, there is no x^{-1}. But in $\mathbf{R}_n$, since $x^n = 1$, we can write x^{-1} for x^{n-1}, so this equation is valid.

It turns out that if, instead of $h(x)$, we use the reciprocal polynomial $\overline{h}(x)$ to make a matrix, we do indeed get a check-matrix for C (and so a generator-matrix for $C^\perp$).

Proposition 6.22. *Let C be a cyclic $[n, k]$ code with check-polynomial $h(x) = h_0 + h_1 x + \cdots + h_k x^k$. Then*

i) C has check-matrix

$$H = \begin{pmatrix} h_k & h_{k-1} & \cdots & h_0 & & & & & \\ & h_k & h_{k-1} & \cdots & h_0 & & & 0 & \\ & & h_k & h_{k-1} & \cdots & h_0 & & & \\ & 0 & & \ddots & \ddots & & & \ddots & \\ & & & & h_k & h_{k-1} & & \cdots & h_0 \end{pmatrix} \in M_{n-k,n}(\mathbb{F}_q).$$

ii) The dual code $C^\perp$ is cyclic and generated by the reciprocal polynomial $\overline{h}(x)$.

Part ii) almost says that $\overline{h}(x)$ is the generator-polynomial for $C^\perp$, but strictly speaking $\overline{h}(x)$ might not be monic, so we would take $h_0^{-1}\overline{h}(x)$ instead.

[4] It is an *integral domain*.

Proof. **[Optional]**

i) We shall show that H is a generator-matrix for $C^\perp$. As $h(x)$ is monic, $h_k = 1$, and so again the echelon form shows that the rows are independent, so they generate a code of dimension $n-k$. This is the dimension of $C^\perp$, so it will be enough to show that the rows of H are in $C^\perp$.

Consider any codeword $c(x) = c_0 + c_1 x + \cdots + c_{n-1}x^{n-1} \longleftrightarrow \mathbf{c} = (c_0, c_1, \ldots, c_{n-1})$ in C. Since $h(x)$ is the check-polynomial, we have

$$c(x)h(x) = (c_0 + c_1 x + \cdots + c_{n-1}x^{n-1})(h_0 + h_1 x + \cdots + h_k x^k) = 0.$$

So all its coefficients must be 0, including:

$$\begin{array}{lcllcl}
\text{coeff of } x^k & : & c_0 h_k + c_1 h_{k-1} + \cdots + c_k h_0 & = & \mathbf{c}\cdot \text{ row 1 of } H, \\
\text{coeff of } x^{k+1} & : & c_1 h_k + c_2 h_{k-1} + \cdots + c_{k+1} h_0 & = & \mathbf{c}\cdot \text{ row 2 of } H, \\
\vdots & & \vdots & & \vdots \\
\text{coeff of } x^{n-1} & : & c_{n-k-1} h_k + c_{n-k} h_{k-1} \cdots + c_{n-1} h_0 & = & \mathbf{c}\cdot \text{ row } n-k \text{ of } H.
\end{array}$$

Thus every row of H is orthogonal to any $\mathbf{c}$ in C, as required.

ii) First, we check that $\overline{h}(x)$ is a factor of $x^n - 1$. Since $h(x)g(x) = x^n - 1$, we also know that $h(x^{-1})g(x^{-1}) = x^{-n} - 1$. Multiplying both side by x^n, we get

$$\begin{array}{rcl}
x^k h(x^{-1}) x^{n-k} g(x^{-1}) & = & x^n(x^{-n} - 1), \text{ so} \\
\overline{h}(x) x^{n-k} g(x^{-1}) & = & 1 - x^n,
\end{array}$$

and $\overline{h}(x)$ is a factor of $x^n - 1$ as required. Now if $\overline{h}(x)$ is monic then (by the remarks following Theorem 6.15 iii) it is the generator-polynomial of $\langle \overline{h}(x) \rangle$, so by Proposition 6.17 $\langle \overline{h}(x) \rangle$ has generator-matrix H, which by i) is the check-matrix for C. Thus $\langle \overline{h}(x) \rangle = C^\perp$. If $\overline{h}(x)$ is not monic then $h_0^{-1}\overline{h}(x)$ is the generator-polynomial of $\langle \overline{h}(x) \rangle$, which by Proposition 6.17 has generator-matrix $h_0^{-1}H$. But multiplying by h_0 is a row-operation, so H is also a generator-matrix, and again $\langle \overline{h}(x) \rangle = C^\perp$.

$\square$