25.4 Syndrome Decoding

Standard Arrays

In the last section, we worked through an algorithm for decoding a received word $\vec{r}$ back into the original transmitted codeword $\vec{c}$ after being sent over a noisy channel. The trouble with the previous algorithm, namely **Algorithm 25.3.17**, was that it was only capable of decoding $\vec{r}$ for which one error had been induced by the channel. This is where some of our most fundamental results from group theory will come into use: to develop a more general and efficient technique for decoding linear codes.

Before we introduce any of the new results, we need to recall some important results from chapters 10 and 11. The results have been stated again below to serve as a reminder.

Definition 10.1.1

Group A **Group** is a nonempty set G, together with an operation, which can be thought of as a function $* : G \times G \to G$, that assigns to each ordered pair (a, b) of elements in G an element $a * b \in G$, that satisfies the following properties:

1. **Associativity:** The operation is associative: $(a * b) * c = a * (b * c)$ for all $a, b, c \in G$.
2. **Identity:** There is an element e (called the identity) in G, such that $a * e = e * a = a$ for all $a \in G$.
3. **Inverses:** For each $a \in G$, there is an element $b \in G$ (called the inverse of a) such that $a * b = b * a = e$.

Subgroup

Subgroup If G is a group and H is a subset of G which is also a group (using the same operation), then we say H is a subgroup of G and we write $H < G$

Recall that groups may be either additive or multiplicative. With the above definitions in mind, we want to recall two definitions from Chapter 18 that are going to allow us to define and construct a **standard array** for a linear code C. This structure is going to allow us the opportunity to explore two new decoding algorithms.

Definition 18.1.1

Right Coset of H in G. Let G be a group and H be a subgroup of G. For any a in G, the set

$$Ha = \{ha \mid h \in H\}$$

is called the **right coset of H in G containing a.** The element a is called the coset representative of Ha.

Note that when we defined cosets in Chapter 18, we defined both left and right cosets. For our purposes, we will only be utilizing right cosets. We also need to note that for our intended application, we will be using addition as our group operation and our right coset definition may be more telling written as such:

$$H + a = \{h + a \mid h \in H\}$$

There is one more theorem that is so fundamental to group theory, as well as the intuition behind our desired structure that it would be a disservice to the reader to not explicitly recall it below:

Theorem 18.2.1

Lagrange's Theorem If G is a finite group and H is a subgroup of G, the $|H|$ divides $|G|$.

Now, let's do an example.

Example 25.4.1

Finding Cosets Let's consider the group from **Example 25.3.4**:

$$G = \{(0\ 0), (1\ 0), (0\ 1), (1\ 1)\}$$

Now, consider the subgroup $H = \{(0\ 0), (1\ 0)\} < G$

Using vector addition over $\mathbb{Z}_2$ and the subgroup H, our right cosets of H are:

$$H + (0\ 0) = \{(0\ 0), (1\ 0)\} = H$$

and

$$H + (0\ 1) = \{(0\ 1), (1\ 1)\} = H + (1\ 1)$$

We have now recalled enough results from previous chapters to define our desired structure! Remember that our code C was a subspace of size q^k of a larger space $V_n(F)$. We are now going to view C as a subgroup of the finite group $V_n(F)$. This way, we can take advantage of our results on groups and subgroups that we recalled. Note that the subgroup C is of size q^k and the group $V_n(F)$ is of size q^n. We may conclude that C will have $q^k - q^n = q^{n-k}$ distinct cosets. Of course, each coset is going to have a coset representative. We are going to take the vector of minimum weight in each coset as the representative and for our purposes, call it the **coset leader**.

We are now going to construct our **standard array**. We are going to make a table. First we calculate all the distinct cosets of our subgroup C. These will be the rows of our table. Our first row will be our $H + e$ coset, in other words, all the codewords of C. The first entry of each row will be the coset leader of each coset, so that the first column of our table consists

of all the coset leaders. Perhaps this idea is best expressed by analyzing an example:

Example 25.4.2

Standard Array Consider the code C generated by the matrix

$$G = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

The standard array for the code C is:

coset leaders							
000000	110100	011010	101001	101110	110011	011101	000111
000001	110101	011011	101000	101111	110010	011100	000110
000010	110110	011000	101011	101100	110001	011111	000101
000100	110000	011110	101101	101010	110111	011001	000011
001000	111100	010010	100001	100110	111011	010101	001111
010000	100100	001010	111001	111110	100011	001101	010111
100000	010100	111010	001001	001110	010011	111101	100111
001100	111000	010110	100101	100010	111111	010001	001011

In the example above we have a $(6, 3)$-code over the field $\mathbb{F}_2$. Notice that the first row of the standard array consists of all $2^3 = 8$ codewords of C, or the coset C. Notice that, as labelled, the first column consists of the coset leaders. Each element of the remaining rows represents the codeword in the j^{th} position $\vec{c}_j$ plus the coset leader in the i^{th} position $\vec{l}_i$. Each row is therefore the coset $H + \vec{l}_i$. Notice also that the table consists of every element of the group $V_n(F)$ appearing once. This makes sense because all 8 cosets of C are listed and we know from chapter 18 that cosets are equivalence classes and therefore partition the group. When trying to construct the above array yourself, you may notice that while the coset leaders of the first seven cosets are quite easily identified, there are a few options for the coset leader of the last coset. We notice that there are actually three vectors of the minimum weight 2. In this case, any one will do as the coset leader. We arbitrarily chose to use 001100, but 100010 or 010001 would have also done nicely.

Standard Array Decoding

We now have enough information to present another decoding algorithm for linear codes, namely **standard array decoding**.

Algorithm 25.4.3

Standard Array Decoding Algorithm Construct the standard array S for the linear code C.

Suppose we receive the word $\vec{r}$.

1. Find $\vec{r}$ in the standard array S.
2. Correct $\vec{r}$ to the codeword at the top of it's column.

Example 25.4.4

Standard Array Decoding Algorithm For this example, suppose we are using the code from **Example 25.4.2**. We have the standard array constructed above for our convenience

Suppose we receive the word $\vec{r} = (1\ 0\ 0\ 1\ 1\ 0)$.

1. Find $\vec{r}$ in the standard array S. It is in row 5, column 5.
2. The codeword at the top of column 5 is $\vec{c} = (1\ 0\ 1\ 1\ 1\ 0)$. Correct $\vec{r}$ to $\vec{c}$.

As an observation, notice the error vector $\vec{e} = (0\ 0\ 1\ 0\ 0\ 0)$, is actually the coset leader of that coset. This is an important feature of standard array decoding.

Syndrome Decoding

The standard array algorithm is useful, but requires a lot of precomputation by requiring that the standard array be computed and constructed prior to running the algorithm. We will now look at a technique which relies on similar ideas, but is a lot faster to implement. First we need a definition.

Definition 25.4.5

Syndrome Let H be the parity check matrix for a code C. We call the vector $s^T = H\vec{r}^T$ the **syndrome** of $\vec{r}$.

This idea may be recognizable from the last section on linear codes. We computed syndromes in our single error decoding algorithm, or **Algorithm 25.3.17**. In order to utilize syndromes in our next algorithm, we need to precompute the syndromes of each coset leader. Let's compute the syndromes for the coset leaders from our previous example.

Example 25.4.6

Computing Syndromes of Coset Leaders Consider the code C from **Example 25.4.2**. The parity check matrix for this code is:

$$H = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

We construct the syndromes for each of the eight coset leaders of C. ie. $H + \vec{l_i}$ for $1 \leq i \leq 8$. We will record the coset leaders and their corresponding syndromes in the table below.

coset leader		syndrome
000000	$\rightarrow$	000
000001	$\rightarrow$	101
000010	$\rightarrow$	011
000100	$\rightarrow$	110
001000	$\rightarrow$	100
010000	$\rightarrow$	010
100000	$\rightarrow$	100
001100	$\rightarrow$	111

We almost have enough information to introduce the goal of this section: the **syndrome decoding algorithm**. We only need one more theorem.

Theorem 25.4.7

Syndromes and Cosets Let H be a parity check matrix for a linear code C. Two vectors $\vec{x}$ and $\vec{y}$ are in the same coset of C if and only if they have the same syndrome. ie. $H\vec{x} = H\vec{y}$.

We can now introduce our last algorithm for decoding linear codes!

Algorithm 25.4.8

Syndrome Decoding Algorithm Construct a one-to-one correspondence between the coset leaders and their syndromes. Let H be the parity check matrix of C and $\vec{r}$ be the vector received over the channel.

1. Compute the syndrome of $\vec{s}^T = H\vec{r}^T$.
2. Find the coset leader $\vec{l}$ associated with $\vec{s}^T$.
3. Correct $\vec{l}$ to $\vec{c} = \vec{r} - \vec{l}$

Let's do an example. We will again use the code from **Example 25.4.2** for convenience.

Example 25.4.9

Syndrome Decoding Notice that we constructed the one-to-one correspondence for C in **Example 25.4.6**. Let H be the parity check matrix of C which is also presented in **Example 25.4.6**.

and $\vec{r} = (1\ 0\ 0\ 0\ 1\ 1)$ be the vector received over the channel.

1. Compute the syndrome of $\vec{s}^T = H\vec{r}^T = (0\ 1\ 0)^T$.
2. By looking at our table in **Example 25.4.6** the coset leader $\vec{l} = (0\ 1\ 0\ 0\ 0\ 0)$ is associated with the syndrome $(0\ 1\ 0)^T$.
3. Correct $\vec{r}$ to $\vec{c} = \vec{r} - \vec{l} = (1\ 0\ 0\ 0\ 1\ 1) - (0\ 1\ 0\ 0\ 0\ 0) = (1\ 1\ 0\ 0\ 1\ 1)$