

# Reflection Group Codes and their Decoding

W. Wesley Peterson, *Fellow, IEEE*, J. B. Nation and Marc Fossorier, *Fellow, IEEE*

**Abstract**—This paper builds on Mittelholzer and Lahtonen’s study of group codes for the gaussian channel based on reflection groups. A careful analysis of the action of a reflection group on its roots leads to the development of improved methods for encoding and decoding. The new algorithm is proved to achieve maximum likelihood decoding. The complexity of decoding is analyzed, and it is shown that a proper choice of the sequence of subgroups used in the algorithm can yield significant gains in the efficiency of decoding.

**Index Terms**—group codes, reflection group, Coxeter group

## I. INTRODUCTION

In 1968, Slepian introduced “group codes for the gaussian channel” [1]. The idea was to choose a group of orthogonal matrices and a point on a sphere, and then use the orbit of that point by that group as a set of signals for communication. The roots of this work go back at least a decade before that, and in 1965 he had introduced “permutation codes” [2], which were codes derived by choosing a point on a sphere and acting on it with a group of operations consisting of permutations of the coordinates and reversals of the signs of coordinates. Slepian recognized that these two operations can be written as orthogonal matrices, and therefore permutation codes are group codes for the gaussian channel. Furthermore he recognized that these two operations are reflections, and therefore the groups are reflection groups. In a very comprehensive paper in 1996, Mittelholzer and Lahtonen [3] considered the group codes for the gaussian channel generated by all reflection groups, and they were able to characterize all these codes and calculate the optimum initial point for each of them. They also gave efficient decoding algorithms for all these codes.

Some of these codes have better minimum distance than the widely used nPSK modulation, and they can compare favorably with QAM modulation. We wondered

W. Peterson is with the Department of ICS, University of Hawaii, Honolulu, HI 96822. J. Nation is with the Department of Mathematics, University of Hawaii, Honolulu, HI 96822, USA email: jb@math.hawaii.edu. M. Fossorier is with ETIS ENSEA, UCP, CNRS UMR-8051, 6 avenue du Ponceau, 95014 Cergy Pontoise, France email: mfossorier@ieee.org.

why they are not used in practical implementations. Our goal is to see how these codes compare in practical situations. To this end, we have been working on making the implementation efficient.

Mittelholzer and Lahtonen found an efficient decoding algorithm using a large permutation code subgroup of the reflection group. We have found that by using a nested sequence of subgroups that are in most cases reflections subgroups, there is an algorithm that is fast and facilitates encoding and decoding of information.

Other groups can be used for group codes. Niyomsataya, Miri and Nevins have analyzed the use of (infinite) affine reflection group codes with the group action restricted to a bounded region [4]. With Ann Shepler, the authors have shown that certain complex reflection groups can be used for group codes, despite differences in the geometry. We are continuing to investigate along those lines.

## II. REFLECTION GROUPS AND REFLECTION GROUP CODES

The paper by Mittelholzer and Lahtonen [3] is quite complete and thorough. We will therefore summarize some facts about reflection groups and some of the results of their paper without proofs and build on that material. Proofs can be found in the standard textbooks on reflection groups, e.g., Grove and Benson [5], Humphreys [6] or Kane [7]. There is significant interplay between geometry and algebra in reflection groups, and it is useful to view both perspectives. We begin with a more geometric description.

A reflection group is a group of orthogonal matrices that is generated by a set of reflections. Each reflection in the group is a reflection in a (hyper)plane that goes through the origin. One reflection acting on the reflecting plane of another makes a third reflection plane. Certain configurations result in a finite number of reflection planes and hence a finite group. For example, in two dimensions, there will be a finite reflection group if and only if the angle between reflecting planes divides 360 degrees. There is a list of all possible irreducible reflection groups in the Mittelholzer and Lahtonen paper.

These reflection planes divide the entire space into finitely many regions. You can choose any region to be the *fundamental region*. Then the planes that bound

the fundamental region are the *fundamental planes* and the reflections associated with those planes are the *fundamental reflections*. For a reflection group that acts irreducibly on an  $n$ -dimensional space, the number of fundamental planes and the number of fundamental reflections is  $n$ . Every element of the group can be written as a product of these fundamental reflections. There will be more reflections among the other elements, but not all the elements will be reflections. For the group  $E_6$ , for example, there are six fundamental reflections, 36 reflections in all, and a total of 51840 group elements. Every group element (except the identity element) maps each region into a different region. Thus if  $\mathbf{v}$  is a vector not on the boundary between two regions and  $g \neq I$  is a group element, then  $\mathbf{v}$  and  $g\mathbf{v}$  are in different regions.

We will consider a unit sphere and instead of talking about points on the sphere we will talk about vectors from the origin to points on the sphere. Let us choose an initial vector that is in the fundamental region. Then the orbit that results from applying all the group elements to the initial vector consists of one vector in each region. In group coding, a correspondence is established between messages and this set of vectors. Given a message to be transmitted, you choose one of these vectors and send it, and a vector that differs because of noise is received. The error probability depends on the exact choice of initial vector. Mittelholzer and Lahtonen showed that the optimum initial vector is a vector that is at the same distance from every bounding plane of the fundamental region, and so essentially in the very middle of the region. A received vector should be decoded into the code vector that is in the same region. You could do maximum-likelihood decoding by calculating the distance from the received vector to each code vector, but there are much more efficient ways to decode.

For every reflection, there are two unit vectors normal to the plane, one on each side. Those are called *roots*. The ones that are on the same side of their respective reflection planes as the fundamental region are called *positive roots*, and the ones on the opposite side are called *negative roots*. A vector is on the positive side of a reflection plane if its inner product with the positive root is positive, and thus a vector is in the fundamental region if and only if its inner product with every fundamental root is positive.

Now let us introduce the algebraic notation for the preceding ideas. Consider a reflection group  $\mathbf{G}$  acting on  $\mathbb{R}^n$ , with identity denoted by  $I$ . For a vector  $\alpha$  of unit length,  $S_\alpha$  denotes the reflection along  $\alpha$ , given by

$$S_\alpha(\mathbf{x}) = \mathbf{x} - 2(\mathbf{x}, \alpha)\alpha$$

where  $(\mathbf{x}, \alpha)$  denotes the standard Euclidean inner prod-

uct (dot product). Let  $\Delta_{\mathbf{G}}$  denote the set of roots of  $\mathbf{G}$ , i.e.,

$$\Delta_{\mathbf{G}} = \{\alpha \in \mathbb{R}^n : \|\alpha\| = 1 \text{ and } S_\alpha \in \mathbf{G}\}.$$

For  $\alpha \in \Delta_{\mathbf{G}}$ , let  $N_\alpha$  denote the reflecting plane:

$$N_\alpha = \{\mathbf{x} \in \mathbb{R}^n : (\mathbf{x}, \alpha) = 0\}.$$

The projection of a vector onto  $N_\alpha$  is given by

$$P_{N_\alpha}(\mathbf{x}) = \mathbf{x} - (\mathbf{x}, \alpha)\alpha.$$

The choice of an initial vector  $\mathbf{x}_0$  (on the unit sphere but not on one of the reflecting planes) determines the positive roots  $\Delta_{\mathbf{G}}^+$ , the fundamental region  $\text{FR}(\mathbf{G})$ , and the fundamental root system  $\Sigma$  in the following ways.

$$\begin{aligned} \Delta_{\mathbf{G}}^+ &= \{\alpha \in \Delta_{\mathbf{G}} : (\alpha, \mathbf{x}_0) > 0\} \\ \text{FR}(\mathbf{G}) &= \{\mathbf{y} \in \mathbb{R}^n : (\mathbf{y}, \alpha) > 0 \text{ for all } \alpha \in \Delta_{\mathbf{G}}^+\} \\ \Sigma &= \{\rho \in \Delta_{\mathbf{G}}^+ : P_{N_\rho}(\mathbf{x}_0) \in \overline{\text{FR}(\mathbf{G})} - \{\mathbf{0}\}\} \\ &= \{\rho \in \Delta_{\mathbf{G}}^+ : \rho \neq \mathbf{x}_0 \text{ and } (P_{N_\rho}(\mathbf{x}_0), \alpha) \geq 0 \\ &\quad \text{for all } \alpha \in \Delta_{\mathbf{G}}^+\} \end{aligned}$$

where  $\overline{X}$  denotes the closure of  $X$ . It is straightforward to see that these algebraic formulations agree with the geometric descriptions given earlier.

A straightforward method to find an optimal choice of the initial vector  $\mathbf{x}_0$ , so that it is a unit vector equidistant from the walls of the fundamental region, is given in Mittelholzer and Lahtonen [3]; see subsection IX-A.

*Theorem 1:* Every positive root can be written as a linear combination of fundamental roots with all positive coefficients, and every negative root can be written as a linear combination of fundamental roots with all negative coefficients.

Thus we have

$$\text{FR}(\mathbf{G}) = \{\mathbf{y} : (\mathbf{y}, \alpha) > 0 \text{ for all } \alpha \in \Sigma\}.$$

Indeed, if  $\mathbf{v}$  is any vector in the fundamental region, then a root  $\mathbf{r}$  is positive if and only if  $(\mathbf{r}, \mathbf{v}) > 0$ .

*Theorem 2:* If  $g \in \mathbf{G}$  and  $\alpha \in \Delta_{\mathbf{G}}$ , then  $gS_\alpha g^{-1} = S_{g\alpha}$ .

Thus if  $g$  is a group element and  $\mathbf{r}$  is a root, then  $g\mathbf{r}$  is also a root. Here,  $\mathbf{r}$  and  $g\mathbf{r}$  might both be positive or both negative, or one each. Define  $\Delta_G(g)$  to be the set of positive roots that are carried into negative roots by  $g$ .

*Theorem 3:* Assume that  $\mathbf{v}$  is a vector in a region and  $g$  is a group element. The reflecting plane corresponding to a positive root  $\alpha$  is between  $\mathbf{v}$  and  $g\mathbf{v}$  if and only if  $\alpha \in \Delta_G(g^{-1})$ .

*Proof:* The reflecting plane is between  $\mathbf{v}$  and  $g\mathbf{v}$  if and only if  $(\mathbf{v}, \alpha)$  and  $(g\mathbf{v}, \alpha) = (\mathbf{v}, g^{-1}\alpha)$  have opposite signs, which means that  $g^{-1}\alpha$  must be a negative root, and therefore  $\alpha \in \Delta_G(g^{-1})$ . ■

Every group element can be expressed as a product of the fundamental reflections. That representation may not be unique, as several expressions could evaluate to the same group element. For each group element there is a minimum number of factors in the expression. There may even be more than one expression with a minimum number of factors for a group element. An expression for a group element with the minimum number of factors, i.e., one that cannot be shortened, is said to be *reduced*. The minimum number of factors in expressions for a group element  $g$  is called the length of  $g$ , denoted  $\ell_G(g)$  or just  $\ell(g)$  if  $\mathbf{G}$  is understood.

*Theorem 4:* If  $S_\alpha$  is a fundamental reflection of  $\mathbf{G}$  and  $\alpha$  the corresponding positive root, then  $S_\alpha\alpha = -\alpha$  and  $S_\alpha$  permutes all the other positive roots.

*Lemma 5:* Let  $\Sigma$  be a fundamental root system for  $\mathbf{G}$ , and let  $\alpha \in \Sigma$ . (So in particular,  $\alpha$  is a positive root of  $\mathbf{G}$ .) Let  $\mathbf{x}_0$  be any vector in the fundamental region of  $\mathbf{G}$ . Consider  $S_\alpha g$  where  $g$  is an arbitrary element of  $\mathbf{G}$ . Then  $\ell(S_\alpha g) = \ell(g) \pm 1$ , and the following are equivalent.

- 1)  $\ell(S_\alpha g) = \ell(g) + 1$ .
- 2)  $S_\alpha\varphi$  is reduced, where  $\varphi$  is any reduced expression for  $g$ .
- 3)  $(\alpha, g\mathbf{x}_0) > 0$ .
- 4)  $g^{-1}\alpha$  is a positive root.

It follows that  $\ell(S_\alpha g) = \ell(g) - 1$  whenever  $g^{-1}\alpha$  is a negative root.

The next result deals with calculating the distance from a vector to  $\mathbf{x}_0$ , which is a measure of how far it is from the fundamental region.

*Theorem 6:* Let  $\mathbf{G}$  be a reflection group and  $\alpha$  a positive root of  $\mathbf{G}$ . Let  $\mathbf{x}_0$  be any vector in the fundamental region of  $\mathbf{G}$ .

- 1) If  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ , then  $\|S_\alpha\mathbf{a} - \mathbf{b}\| < \|\mathbf{a} - \mathbf{b}\|$  if and only if  $(\alpha, \mathbf{a})(\alpha, \mathbf{b}) < 0$ .
- 2) If  $\mathbf{a} \in \mathbb{R}^n$ , then  $\|S_\alpha\mathbf{a} - \mathbf{x}_0\| < \|\mathbf{a} - \mathbf{x}_0\|$  if and only if  $(\alpha, \mathbf{a}) < 0$ .

Comparing Lemma 5(3) and Theorem 6(2), we see that for a fundamental positive root  $\alpha$  and any element  $g \in \mathbf{G}$ , we have

$$\begin{aligned} \|S_\alpha g\mathbf{x}_0 - \mathbf{x}_0\| &< \|g\mathbf{x}_0 - \mathbf{x}_0\| && \text{iff } (\alpha, g\mathbf{x}_0) < 0 \\ &&& \text{iff } \ell(S_\alpha g) = \ell(g) - 1. \end{aligned}$$

This observation is basic to our method.

Note that two points  $\mathbf{u}$  and  $\mathbf{v}$  are on the same side of a reflecting plane  $N_\alpha$  if the dot products  $(\mathbf{u}, \alpha)$  and  $(\mathbf{v}, \alpha)$  have the same sign, and they are on opposite sides

of  $N_\alpha$  if  $(\mathbf{u}, \alpha)$  and  $(\mathbf{v}, \alpha)$  have opposite signs. In the latter case, we say that  $N_\alpha$  separates  $\mathbf{u}$  and  $\mathbf{v}$ .

*Corollary 7:* If  $\alpha$  is any root and  $\mathbf{u}$  and  $\mathbf{v}$  are on the same side of the reflection plane  $N_\alpha$ , then  $\mathbf{v}$  is closer than  $S_\alpha\mathbf{v}$  to  $\mathbf{u}$ . Similarly, if  $\mathbf{u}$  and  $\mathbf{v}$  are on opposite sides of  $N_\alpha$ , then  $S_\alpha\mathbf{v}$  is closer than  $\mathbf{v}$  to  $\mathbf{u}$ .

*Theorem 8:* The following are equivalent for an element  $g$  of a reflection group  $\mathbf{G}$ .

- 1)  $\ell(g) = k$ .
- 2) For any vector  $\mathbf{v}$  that is not in any reflecting plane, the number of reflecting planes that separate  $\mathbf{v}$  and  $g\mathbf{v}$  is  $k$ .
- 3)  $|\Delta_G(g)| = k$ .

Moreover, if  $\mathbf{v}$  is a vector in any region and  $\alpha$  is a fundamental reflection of  $\mathbf{G}$ , then  $\mathbf{v}$  and  $S_\alpha\mathbf{v}$  are on opposite sides of the reflection plane for  $\alpha$ , but it turns out that  $\mathbf{v}$  and  $S_\alpha\mathbf{v}$  are on the same side of every other reflecting plane.

The preceding theorems are the basis for a fairly efficient decoding method for group codes derived from reflection groups. Let  $\mathbf{x}_0$  denote an initial vector chosen in the fundamental region. To send a message corresponding to the group element  $g$ , we transmit the vector  $g^{-1}\mathbf{x}_0$ . Suppose  $\mathbf{u}$  is the received vector. If there is not too much added noise, then  $\mathbf{u}$  should be in the same region as  $g^{-1}\mathbf{x}_0$ , and consequently  $g\mathbf{u}$  should be in the same region as  $\mathbf{x}_0$ , that is, in the fundamental region. Given  $\mathbf{u}$ , we want to find  $g$ . The outline of a decoding algorithm to find  $g$  is as follows.

The element  $g$  has some finite length, say  $\ell(g) = k$ . Then by Theorem 8, there are  $k$  reflecting planes between  $\mathbf{u}$  and  $\mathbf{x}_0$ . Let  $S_i$  for  $i = 1, \dots, k$  be the fundamental reflections for these planes, in the order in which you would cross them going from  $\mathbf{u}$  to  $\mathbf{x}_0$ . Define a sequence of vectors by  $\mathbf{u}_0 = \mathbf{u}$  and  $\mathbf{u}_i = S_i\mathbf{u}_{i-1}$  for  $i = 1, \dots, k$ . There will be one fewer reflecting planes between  $\mathbf{u}_i$  and  $\mathbf{x}_0$  than between  $\mathbf{u}_{i-1}$  and  $\mathbf{x}_0$ , and therefore there will be  $k - i$  reflecting planes between  $\mathbf{u}_i$  and  $\mathbf{x}_0$ . By induction, there will be no reflecting planes between  $\mathbf{u}_k$  and  $\mathbf{x}_0$ . Therefore,  $\mathbf{u}_k = S_k S_{k-1} \cdots S_1 \mathbf{u}$  and  $\mathbf{x}_0$  are in the same region, whence  $g = S_k S_{k-1} \cdots S_1$ .

Now, how do you know how to choose the fundamental reflections  $S_1, S_2, S_3, \dots, S_k$ ? You could use trial and error, and there are two ways to test whether a reflection is a suitable choice as  $S_i$ . Let  $S_\alpha$  be a reflection and  $\alpha$  the corresponding root. One way is to calculate the inner product  $(\mathbf{u}_i, \alpha)$ . If this is negative, then  $S_\alpha$  is a suitable choice for  $S_i$  because then  $\mathbf{u}_i$  is on the opposite side of the reflection plane for  $S_\alpha$  from the fundamental region, where  $\mathbf{x}_0$  is. The other way is to calculate the distance between  $\mathbf{u}_i$  and  $\mathbf{x}_0$  and the distance between  $S_\alpha\mathbf{u}_i$  and  $\mathbf{x}_0$ . If the latter is smaller, then the reflection plane for

$S_\alpha$  must be between  $\mathbf{u}_i$  and  $\mathbf{x}_0$ , by Theorem 6. Then  $S_\alpha$  must be a suitable choice.

### III. SUBGROUPS OF REFLECTION GROUPS

Mittelholzer and Lahtonen could improve on this method of decoding by using a subgroup of the reflection group that is a Slepian permutation code. Another variation on this theme, using subgroups to decode more efficiently, is found in Lahtonen [9]. We have found that further refinements are possible by recursively using a sequence of subgroups, each containing the next.

A subgroup  $\mathbf{H}$  of a reflection group  $\mathbf{G}$  is a *reflection subgroup* if it is generated by a subset of the reflections of  $\mathbf{G}$ . Not all subgroups of reflection groups are reflection groups. For example, the subgroup of all matrices whose determinant is 1 contains no reflections. (The determinant of a reflection is  $-1$ .) Of particular importance among the reflection subgroups are the *parabolic subgroups*. A subgroup  $\mathbf{H}$  of a reflection group  $\mathbf{G}$  is a parabolic subgroup if  $\mathbf{H}$  is generated by a subset of the fundamental reflections of  $\mathbf{G}$ . There exist reflection subgroups that are not parabolic subgroups. Mittelholzer and Lahtonen found that  $D_8$  is a subgroup of  $E_8$  and  $A_7$  is a subgroup of  $E_7$ , and in neither case is the subgroup a parabolic subgroup. In fact, since parabolic subgroups are generated by a subset of the fundamental reflections and the dimension of the space acted on by a reflection group is equal to the number of its fundamental reflections, we see that a parabolic subgroup always has smaller dimension than the whole group. Thus we must distinguish parabolic subgroups, reflection subgroups, and general subgroups. The former ones have nicer properties, but we will have occasion to use all three types.

Now consider a reflection group  $\mathbf{G}$  with  $m$  elements and  $\mathbf{H}$  a reflection subgroup with  $k$  elements. Then the whole space is divided into  $m$  regions, and one of those regions is chosen to be the fundamental region of  $\mathbf{G}$ . The initial vector  $\mathbf{x}_0$  is then chosen in the fundamental region. The positive roots of  $\mathbf{G}$  are all the roots  $\mathbf{r}$  for which  $(\mathbf{r}, \mathbf{x}_0) > 0$ . The fundamental region can then be described as all those vectors  $\mathbf{v}$  such that  $(\mathbf{v}, \mathbf{r}) > 0$  for every positive root  $\mathbf{r} \in \Delta_{\mathbf{G}}^+$ .

The reflection subgroup  $\mathbf{H}$  also defines regions, and the whole space is divided into  $k$  of these regions. The fundamental region for the subgroup is likewise chosen to be its region that contains  $\mathbf{x}_0$ . Then the following theorem, which is crucial to subgroup decoding, holds.

*Theorem 9:* If  $\mathbf{H}$  is a reflection subgroup of  $\mathbf{G}$ , then the fundamental region for  $\mathbf{G}$  is contained in the fundamental region for  $\mathbf{H}$ .

*Proof:* Since  $\mathbf{H} \leq \mathbf{G}$ , roots of  $\mathbf{H}$  are also roots of  $\mathbf{G}$ . Because we are using the same initial vector  $\mathbf{x}_0$  to determine the fundamental regions, positive roots of  $\mathbf{H}$  are positive roots of  $\mathbf{G}$ , in symbols  $\Delta_{\mathbf{H}}^+ \subseteq \Delta_{\mathbf{G}}^+$ . Thus

$$\begin{aligned} \text{FR}(\mathbf{H}) &= \{\mathbf{y} : (\mathbf{y}, \alpha) > 0 \text{ for all } \alpha \in \Delta_{\mathbf{H}}^+\} \\ &\supseteq \{\mathbf{y} : (\mathbf{y}, \alpha) > 0 \text{ for all } \alpha \in \Delta_{\mathbf{G}}^+\} \\ &= \text{FR}(\mathbf{G}), \end{aligned}$$

as claimed.  $\blacksquare$

Since a group acts transitively on its regions, and the choice of the fundamental region is arbitrary, it follows that each region of  $\mathbf{H}$  contains  $|\mathbf{G}|/|\mathbf{H}|$  regions for  $\mathbf{G}$ .

We will apply Theorem 9 recursively to a chain of reflection subgroups

$$\{I\} = \mathbf{G}_0 < \mathbf{G}_1 < \cdots < \mathbf{G}_m = \mathbf{G}$$

in which case the fundamental regions will be nested in the reverse order:

$$\mathfrak{R}^n = \text{FR}(I) \supseteq \text{FR}(\mathbf{G}_1) \supseteq \cdots \supseteq \text{FR}(\mathbf{G}).$$

In fact, our algorithm will iteratively transform the received vector into successively smaller fundamental regions for a nested sequence of subgroups.

For every  $g \in \mathbf{G}$ , recall that  $\Delta_G(g)$  denotes the set of positive roots  $\alpha$  in  $\mathbf{G}$  for which  $g\alpha$  is a negative root, and for a reflection subgroup  $\mathbf{H}$  define  $\Delta_H(g)$  to be the set of positive roots  $\alpha$  of  $\mathbf{H}$  for which  $g\alpha$  is a negative root. Fixing the subgroup  $\mathbf{H}$ , let us define  $\Delta_H^*(g)$  to be the set of positive roots  $\alpha$  in  $\mathbf{G}$  that are not roots of  $\mathbf{H}$  and for which  $g\alpha$  is a negative root. So  $\Delta_G(g) = \Delta_H(g) \dot{\cup} \Delta_H^*(g)$ .

*Theorem 10:* Every left coset of a reflection subgroup  $\mathbf{H}$  of a reflection group  $\mathbf{G}$  contains a unique element  $g_0$  for which  $\Delta_H(g_0)$  is empty.

*Proof:* Fix any coset of  $\mathbf{H}$ , and let  $g$  be an element of that coset with  $\Delta_H(g)$  nonempty, say  $\alpha \in \Delta_H(g)$ . Then  $\alpha$  is a positive root of  $\mathbf{H}$ , and we may assume that it is a fundamental root of  $\mathbf{H}$ . For, denoting the fundamental roots of  $\mathbf{H}$  by  $\beta_1, \dots, \beta_k$ , there are non-negative constants  $c_1, \dots, c_k$  such that  $\alpha = \sum c_i \beta_i$ . Then  $g\alpha = \sum c_i g\beta_i$ , and if  $g\alpha$  is a negative root, then some  $g\beta_i$  must be negative.

Let  $g_1 = gS_\alpha$ . By Theorem 4,  $S_\alpha$  takes  $\alpha$  to  $-\alpha$ , and permutes the remaining positive roots of  $\mathbf{H}$ . Thus the action of  $g_1$  on the positive roots of  $\mathbf{H}$  is that  $g_1\alpha = -g\alpha$ , which is a positive root, while  $g_1$  permutes the remaining images of roots of  $\mathbf{H}$  under  $g$ . Thus  $\Delta_H(g_1)$  contains one fewer element than  $\Delta_H(g)$ , i.e.,  $|\Delta_H(g_1)| = |\Delta_H(g)| - 1$ , and  $g_1$  is in the same coset as  $g$ . In other words, if  $\Delta_H(g)$  is not empty, then there is another element  $g_1$  of the coset  $g\mathbf{H}$  with one fewer root.

It follows that there must be an element  $g_0$  in the coset  $g\mathbf{H}$  for which  $\Delta_H(g_0)$  is empty.

Now let us consider whether there could be two such elements,  $g_0$  and  $g_0h$ , where  $h$  is not the identity element.  $\Delta_H(h)$  must not be empty, because if it were empty, that would imply that  $\ell_H(h) = 0$ , which in turn implies that  $h$  is the identity element. Assume that  $g_0$  carries no positive roots of  $\mathbf{H}$  into negative roots, and note that this implies that no negative roots of  $\mathbf{H}$  are carried into positive roots. However,  $h$  carries some positive roots of  $\mathbf{H}$  into negative roots of  $\mathbf{H}$ . (Note that  $h$  carries every root of  $\mathbf{H}$  into a root of  $\mathbf{H}$ .) Now for  $g_0h$ , the positive roots of  $\mathbf{H}$  that are carried into negative roots of  $\mathbf{H}$  by  $h$  will be carried into negative roots by  $g_0$ , with the result that  $g_0h$  carries some positive roots of  $\mathbf{H}$  into negative roots. Therefore if  $|\Delta_H(g_0)| = 0$  and  $h$  is an element of  $\mathbf{H}$ , but not the identity element, then  $|\Delta_H(g_0h)| \neq 0$ . ■

We will want to use the element  $g_0$  of Theorem 10 as the coset leader for its coset. Let us prove that  $g_0$  is also the unique shortest element in its coset. (However, a coset leader may have more than one minimum-length expression.)

*Lemma 11:* Let  $\mathbf{G}$  be a reflection group and  $\mathbf{H}$  a reflection subgroup, and let  $g$  be an element of shortest length in the coset  $g\mathbf{H}$ . If  $S_\alpha$  is a fundamental reflection in  $\mathbf{G}$  such  $\ell_G(S_\alpha g) = \ell_G(g) - 1$ , then  $S_\alpha g$  is a minimum-length element in its coset.

*Proof:* Suppose there is a shorter element  $f$  than  $S_\alpha g$  in  $S_\alpha g\mathbf{H}$ . Then by Lemma 5,  $\ell(S_\alpha f) \leq \ell(f) + 1 < \ell(g)$ , whence  $g$  is not a minimum-length element of  $g\mathbf{H}$ , contrary to hypothesis. ■

*Theorem 12:* Let  $\mathbf{H}$  be a reflection subgroup of a reflection group  $\mathbf{G}$ , and let  $g_0$  be an element of  $\mathbf{G}$  with  $\Delta_H(g_0)$  empty. Then  $g_0$  is the unique shortest element in the coset  $g_0\mathbf{H}$ , and thus its coset leader.

*Proof:* We use induction on the length of  $g$  to prove the following statement: *if  $g$  is an element of minimal length in its coset  $g\mathbf{H}$ , then  $\Delta_H(g)$  is empty.* In conjunction with Theorem 10, this shows that the element of minimal length and the unique element with  $\Delta_H(g)$  empty in each coset coincide.

The statement is certainly true when  $\ell(g) = 0$ , that is, when  $g = I$ . Assume that the property holds for all elements of length less than  $m$ , and let  $g$  be an element such that  $\ell(g) = m$  and  $\ell(x) \geq m$  for all  $x \in g\mathbf{H}$ . In view of Lemma 11, there exist a positive fundamental root  $\alpha$  of  $\mathbf{G}$  and an element  $g_1$  such that  $g = S_\alpha g_1$  and  $\ell(g_1) = m - 1$ . (These may not be unique.) Clearly  $g_1$  must be of minimal length in its coset  $g_1\mathbf{H}$ , and hence by induction  $\Delta_H(g_1) = \emptyset$ .

We want to show that  $\Delta_H(g)$  is empty, so suppose to the contrary that it is not. Since  $S_\alpha$  changes the sign

of only one root pair, and  $\Delta_H(g_1)$  is empty, that means there is a positive root  $\beta$  of  $\mathbf{H}$  such that  $g_1(\beta) = \alpha$ . But then

$$S_\alpha g_1 = S_{g_1(\beta)} g_1 = g_1 S_\beta g_1^{-1} g_1 = g_1 S_\beta \in g_1\mathbf{H},$$

a contradiction. Therefore an element of minimal length in a coset has  $\Delta_H(g)$  empty, and by Theorem 10 this element is unique in the coset. ■

The property of the next theorem is crucial for the decoding algorithm.

*Theorem 13:* If  $\mathbf{G}$  is a reflection group and  $\mathbf{H}$  is a reflection subgroup, and  $L$  is the coset leader of the coset  $L\mathbf{H}$ , and if  $S_\alpha$  is a fundamental reflection, then the coset leader of the coset  $S_\alpha L\mathbf{H}$  is either  $S_\alpha L$ , or else it is  $L$ . Note that in the former case, the length of the coset leader  $S_\alpha L$  is the length of  $L$  plus or minus one.

A simple example will illustrate the various cases that arise in the proof. Consider the group  $A_4$  with the sequence of parabolic subgroups

$$I < A_1 < A_2 < A_3 < A_4$$

where  $A_1$  is the subgroup generated by  $\{A\}$ , the subgroup  $A_2$  is generated by  $\{A, B\}$ , etc. Then  $L = CD$  is a coset leader for  $A_4$  over  $A_3$ , and we want to write the elements  $XL$  with  $X = A, B, C, D$  as products of coset leaders. We have

$$\begin{array}{ll} ACD = CD \cdot A & CD \text{ is the coset leader} \\ BCD = BCD & BCD \text{ is the coset leader} \\ CCD = D & D \text{ is the coset leader} \\ DCD = CD \cdot C & CD \text{ is the coset leader} \end{array}$$

because  $A$  commutes with  $CD$ ,  $C^2 = I$ , and  $(CD)^3 = I$ , respectively.

*Proof:* Denote the length  $\ell(L)$  by  $m$ . Then by Theorem 12, every other element of the coset  $L\mathbf{H}$  has length greater than  $m$ . By Lemma 5, the length of  $S_\alpha L$  is either  $m - 1$  or  $m + 1$ , and the length of every other element of the coset  $S_\alpha L\mathbf{H}$  is greater than  $m - 1$ .

If the coset leader of  $S_\alpha L\mathbf{H}$  is  $S_\alpha L$ , then we are done. So we consider what happens when it is some other element, say  $K$ . That means there is a element  $h \in \mathbf{H}$  such that  $S_\alpha L = Kh$  with  $h \neq I$ . Now  $\Delta_H(L) = \emptyset$  since  $L$  is a coset leader, while  $S_\alpha$  changes the sign of only one positive root, *viz.*  $\alpha$ , since  $S_\alpha$  is a fundamental reflection of  $\mathbf{G}$ . Thus the only possible positive root of  $\mathbf{H}$  that  $S_\alpha L$  could carry into a negative root would be  $L^{-1}\alpha$ , if indeed  $L^{-1}\alpha$  is a root of  $\mathbf{H}$ . We conclude that  $\Delta_H(S_\alpha L) \subseteq \{L^{-1}\alpha\}$ . However,  $\Delta_H(S_\alpha L)$  is nonempty since  $S_\alpha L$  is not a coset leader, so  $\Delta_H(S_\alpha L) = \{L^{-1}\alpha\}$ .

On the other hand,  $\Delta_H(Kh) = \Delta_H(h)$  because  $h$ , being in  $\mathbf{H}$ , permutes the roots of  $\mathbf{H}$  with some

sign changes, while  $K$ , being a coset leader, effects no further sign changes on the roots of  $\mathbf{H}$ . So  $\Delta_H(h) = \Delta_H(Kh) = \Delta_H(S_\alpha L) = \{L^{-1}\alpha\}$ , whence by Theorem 8 we have  $\ell_H(h) = 1$  and  $h = S_\beta$  for some fundamental root  $\beta$  of  $\mathbf{H}$ . Since  $\Delta_H(h) = \{\beta\}$  we have  $\beta = L^{-1}\alpha$ . Hence  $L\beta = \alpha$  and  $S_\alpha = LS_\beta L^{-1}$  with  $S_\beta \in \mathbf{H}$ . Therefore

$$S_\alpha L = LS_\beta L^{-1} L = LS_\beta = Kh = KS_\beta$$

yielding  $L = K$ , as desired. ■

### A. Generation

In this section we describe how one can efficiently find the reflection subgroups of a reflection group.

The first relevant result is Proposition 1.14 of Humphreys [6]. It tells us that there are no “extra” reflections in  $\mathbf{G}$ .

*Theorem 14:* Let  $Y \subseteq \Delta_{\mathbf{G}}$  be a set of roots of a finite reflection group  $\mathbf{G}$ , and let  $\mathbf{S}$  be the subgroup generated by  $\{S_\alpha : \alpha \in Y\}$ . The roots of  $\mathbf{S}$  are the smallest set  $E$  such that

- 1)  $Y \subseteq E$ ,
- 2)  $\alpha, \beta \in E$  implies  $S_\alpha(\beta) \in E$ .

Secondly, we have a fundamental theorem of Coxeter; see, e.g., Chapter 6 of Kane [7].

*Theorem 15:* Every finite reflection group is given by a presentation  $\langle X, R \rangle$  with generators  $X = \{s_1, \dots, s_n\}$  and relations

- 1)  $s_i^2 = 1$  for all  $i$ ,
- 2)  $(s_i s_j)^{m_{ij}} = 1$  for  $i < j$ .

It is this presentation, of course, that is represented in the *Coxeter graph* of the group. It is well known that there are only a small number of types of finite irreducible reflection groups, i.e., finite reflection groups that are not a direct product of smaller groups, or equivalently, whose Coxeter graph is connected. These are listed in Mittelholzer and Lahtonen [3], or the textbooks Grove and Benson [5], Humphreys [6] and Kane [7]. These basic reflection groups are commonly designated as  $A_n, B_n, D_n, E_6, E_7, E_8, F_4, H_2^n, I_3$  and  $I_4$ . We will use these freely as examples, especially in the last section which concerns the details of coding with specific reflection groups.

To this mix let us add three more basic facts.

- (a) If  $\Sigma$  is a fundamental root system for  $\mathbf{G}$  and  $\alpha, \beta \in \Sigma$  with  $\alpha \neq \beta$ , then  $(\alpha, \beta) \leq 0$ .
- (b) If  $Y \subseteq \Delta_{\mathbf{G}}^+$  is a set of positive roots with the property that  $(\alpha, \beta) \leq 0$  whenever  $\alpha \neq \beta \in Y$ , then  $Y$  is linearly independent.

- (c) If  $\alpha$  and  $\beta$  are positive roots with  $(\alpha, \beta) = -\cos \frac{\pi}{m}$ , then the angle between the vectors is  $\pi - \frac{\pi}{m}$ , and  $(S_\alpha S_\beta)^m = I$ .

Now we can describe the algorithm for finding reflection subgroups as follows. We are given a reflection group  $\mathbf{G}$ , with an initial vector  $\mathbf{x}_0$  and its corresponding fundamental root system  $\Sigma$ .

- 1) Determine the set of all roots  $\Delta_{\mathbf{G}}$ , and all positive roots  $\Delta_{\mathbf{G}}^+$ , for  $\mathbf{G}$ . (This is straightforward; see Theorem 14.)
- 2) Find subsets  $Y \subseteq \Delta_{\mathbf{G}}^+$  with pairwise nonpositive dot products.
- 3) Find the isomorphism type of the subgroup generated by  $\{S_\alpha : \alpha \in Y\}$  using fact (c) above and Theorem 15.

The program is much more efficient if in step 2), subsets  $Y$  that are subsets of larger subsets  $Y'$  or of the fundamental reflections in  $\mathbf{G}$  are not done. This is equivalent to not doing subgroups that are parabolic subgroups of either  $\mathbf{G}$  or any reflection subgroup of  $\mathbf{G}$ .

A fundamental root system  $\Sigma$  is characterized by the following properties:

- 1)  $\Sigma$  is linearly independent.
- 2) Every root in  $\Delta_{\mathbf{G}}$  is a linear combination of elements of  $\Sigma$  with coefficients that are all non-negative or all nonpositive.

It would be nice if the algorithm given above always yielded a fundamental root system for the subgroup in question, so that each subgroup is just counted once. This is usually the case, but not always. A set of positive roots with pairwise nonpositive dot products is independent. However, in a dihedral group  $H_2^n$  with  $n \geq 5$  there will be positive roots, with an angle greater than  $90^\circ$  between them, that have nonadjacent reflecting lines. This same problem occurs in the exceptional groups  $I_3$  and  $I_4$ , which have  $H_2^5$  as a subgroup. With those exceptions, the algorithm always yields a fundamental root system for the subgroup.

*Theorem 16:* Let  $\mathbf{G}$  be one of the irreducible Coxeter groups  $A_n, B_n, D_n, E_6, E_7, E_8$  or  $F_4$ . Let  $Z \subseteq \Delta_{\mathbf{G}}^+$  be a set of positive roots with pairwise nonpositive dot products. Then  $Z$  is a fundamental system for the subgroup generated by  $\{S_\alpha : \alpha \in Z\}$ .

For the groups  $H_2^n, I_3$  and  $I_4$  not covered by the theorem, our program also tests whether a set  $Y$  of positive roots is a fundamental root system for the subgroup it generates. If not, that set is rejected.

Let us say that two subgroups that are isomorphic are of the same type. This method yields many subgroups, but the number of types is fairly small. For example, for  $E_6$ , the following types of subgroups occurred. The

number is the number of distinct occurrences (generated by distinct subsets of the reflections of  $E_6$ ) of that type, not counting parabolic subgroups of  $E_6$  or of previously listed subgroups:

36	$A_5 \times A_1$
25	$D_5$
144	$A_4 \times A_1$
40	$A_2 \times A_2 \times A_2$
218	$A_3 \times A_1 \times A_1$
119	$A_2 \times A_2 \times A_1$
28	$D_4$
97	$A_2 \times A_1 \times A_1$
36	$A_1 \times A_1 \times A_1 \times A_1$

Furthermore, when we calculate the coset leaders of subgroups of the same type, we do not necessarily get coset leaders of the same length. For example, for the 25 subgroups of type  $D_5$ , for three subgroups, the maximum length coset leader has length 8, for maximum lengths 9, 10, 11, and 12, there are 4 subgroups each, for maximum length 13, 14, and 15, there are two subgroups each. Furthermore, this does not include the two parabolic subgroups of type  $D_5$ . The maximum length coset for these is the number of roots in  $E_6$  that are not in  $D_5$ , which is  $36 - 20 = 16$ . We will find that the decoding tree that we construct will have its height equal to the length of the longest coset leader, and thus some of these subgroups are more attractive for decoding than parabolic subgroups.

For some other results on the parabolic subgroups containing a given reflection subgroup, see Dyer [8].

### B. Subgroup tables

Table I lists some reflection subgroups of small index in various reflection groups. These subgroups were found by finding sets of positive roots with pairwise nonpositive direct products, which is quite efficient and includes the parabolic subgroups. Subgroups of a particular type can occur in multiple copies. The table is far from complete.

The *length* column gives the length of the longest coset leader for a subgroup, which is a parameter in determining the complexity of encoding and decoding. This will also be the length of the directed graph  $\Gamma$  of coset leaders described in the next section. These graphs have no cycles. The length of such a graph is the maximum number of *edges* in a path. By Theorem 8, the maximum length of an element in a reflection group is the number of positive roots. In fact, for parabolic

TABLE I  
REFLECTION SUBGROUPS

Group	Subgroup	Index	Length
$A_n$	$A_{n-1}$	$n + 1$	$\lceil \frac{n}{2} \rceil$ to $n$
$B_n$	$D_n$	2	1
	$B_{n-1} \times A_1$	$n$	$\lfloor \frac{n}{2} \rfloor$ to $n - 1$
$D_n$	$A_{n-1}$	$2^n$	$\lceil \frac{n^2+n}{4} \rceil$ to $\frac{n^2+n}{2}$
	$D_{n-1}$	$2n$	$n + 1$ to $2n - 2$
$E_6$	$A_{n-1}$	$2^{n-1}$	$\lceil \frac{n^2-n}{4} \rceil$ to $\frac{n^2-n}{2}$
	$D_5$	27	8 to 16
$E_7$	$A_5 \times A_1$	36	5 to 10
	$E_6$	56	14 to 27
$E_8$	$D_6 \times A_1$	63	8 to 16
	$A_7$	72	8 to 15
$F_4$	$E_7 \times A_1$	120	14 to 28
	$D_8$	135	11 to 22
	$A_8$	1920	21 to 42
$I_3$	$B_4$	3	1 to 2
$I_3$	$H_2^5$	12	5 to 7, 10
	$A_1 \times A_1 \times A_1$	15	4 to 6
$I_4$	$I_3 \times A_1$	60	11 to 22
	$D_4$	75	9 to 16
	$H_2^5 \times H_2^5$	100	14 to 25
	$A_4$	120	11 to 20

subgroups the length of the longest coset leader will be the difference between the number of positive roots of the group and its subgroup. The number of positive roots for a given group, in turn, can be found in a table in Mittelholzer and Lahtonen [3], or computed directly.

Nonparabolic subgroups, on the other hand, involve generators of length greater than 1. There does not appear to be a general formula for the length of coset leaders for nonparabolic subgroups. For these cases, we wrote programs to determine the coset leaders for various subgroups using the method described in the next section.

In most cases, the lengths of the longest coset leader for a given type cover a range from  $k$  to  $2k$  or  $2k - 1$ , for various copies of the subgroup. Again, the groups  $H_2^n$ ,  $I_3$  and  $I_4$  do not follow this pattern exactly. As we shall see, using the subgroups with shorter lengths gives codes that can be more quickly decoded.

## IV. COSET LEADERS, GRAPHS AND NORMAL FORM

Now suppose we are given a sequence of reflection subgroups of a reflection group  $\mathbf{G}$ , say

$$\{I\} = \mathbf{G}_0 < \mathbf{G}_1 < \cdots < \mathbf{G}_{m-1} < \mathbf{G}_m = \mathbf{G}.$$

First we describe an algorithm to find a set of coset leaders for  $\mathbf{G}_i$  over  $\mathbf{G}_{i-1}$  consisting of minimal length terms, so that when combined they will yield a canonical expression for an arbitrary element of  $\mathbf{G}$  as a product of coset leaders. Then we describe how to construct a graph  $\Gamma_i$  for each extension  $\mathbf{G}_i$  over  $\mathbf{G}_{i-1}$ . These

graphs encode the algebraic structure of  $\mathbf{G}$  in terms of the sequence of subgroups, and will be crucial to our encoding and decoding algorithms. The graphs are an adaptation of a well-known tool for describing the structure of reflection groups; see Björner and Brenti [10].

Of course, for theoretical purposes, we need only consider a single subgroup  $\mathbf{H}$  of a group  $\mathbf{G}$ , and then use induction for the sequence.

Let  $\mathbf{G}$  be a finite reflection group, let  $A, B, C, \dots$  be the  $n$  fundamental reflections, and let  $\mathbf{H}$  be an arbitrary subgroup of  $\mathbf{G}$ . Define an *expression* to be a finite sequence of fundamental reflections. If you multiply the reflections in an expression, in sequence, you get a group element. More than one expression can result in the same group element. Recall that an expression is *reduced* if it has minimum length. Define the order for expressions as follows:  $E_1 < E_2$  if the number of factors in  $E_1$  is less than the number of factors in  $E_2$ , or if both have the same number of factors and  $E_1$  precedes  $E_2$  in dictionary order. Since the elements of  $\mathbf{G}$  are transformations, they are composed from right to left, and expressions should be read in the same way. Thus, for example,  $BAC < ABC$  in the dictionary order. The relation  $<$  on expressions is of course a linear order: you can compare any two expressions.

Consider the left cosets  $g\mathbf{H}$  of  $\mathbf{H}$  in  $\mathbf{G}$ . We can choose the *coset leader* of a coset to be the unique shortest element of the coset and use the unique lowest expression for that group element to represent that coset leader.

We will say that two expressions are in the same coset if they evaluate to group elements that are in the same coset. You can determine whether  $E_1$  and  $E_2$  are in the same coset by determining whether  $E_1^{-1}E_2$  is in  $\mathbf{H}$ . If  $\mathbf{G}$  is a reflection group, then you can do this efficiently by decoding  $E_1^{-1}E_2\mathbf{x}_0$  in  $\mathbf{G}$ , using the decoding algorithm described following Theorem 8, and seeing whether the result is the initial vector of  $\mathbf{G}$ .

Next we will define an algorithm for making a list of expressions. (We will prove that this is a list of all coset leaders.) Initially the list contains only the identity element, an expression with no elements. For each expression  $E$  in the list, in order, multiply it on the left by each fundamental reflection  $R$  in order, and then check whether the result is in the same coset as any expression already on the list. If it is not, then add  $RE$  to the end of the list. Note that if  $RE$  is in a coset already in the list, by Theorem 13. it is a coset with leader of length no less than  $\ell(E) - 1$  and it is not necessary to check whether it is in any shorter cosets than that.

*Lemma 17:* The set of all the expressions of length

$n$  on the list consists of all coset leaders of length  $n$  in order, and multiplying each of those by all fundamental reflections in order produces all coset leaders of length  $n + 1$  in order.

*Proof:* The proof is by induction on  $n$ . The statement is obviously true for  $n = 0$ . Assume that it is true for  $n = k$ . Let  $SE$  be a coset leader of length  $k + 1$ , where  $S$  is the first fundamental reflection in the lowest expression for  $SE$ . Then  $SSE = E$  which has length  $k$ . By Theorem 22,  $E$  is the coset leader of the coset that contains it, and it has length  $k$  and therefore by the induction hypothesis is already on the list. It follows that after we try all products of expressions of length  $k$  on the list by all fundamental reflections, we will have found all coset leaders of length  $k + 1$ . Furthermore, by the induction hypothesis, all the expressions on the list of length  $k$  are in order. Therefore the algorithm tests candidates for coset leaders of length  $k + 1$  in order, and the ones that pass the test will be added to the list in order. ■

*Theorem 18:* The list consists exactly of all coset leaders for  $\mathbf{G}$  over  $\mathbf{H}$  in order.

This follows immediately from the lemma.

Now let us construct the graph  $\Gamma$  of (left) coset leaders for  $\mathbf{G}$  over  $\mathbf{H}$ . The vertices of  $\Gamma$  are the coset leaders of  $\mathbf{G}$  over  $\mathbf{H}$ , say  $v_0, \dots, v_{k-1}$  where  $v_0 = I$  and  $k = |\mathbf{G}|/|\mathbf{H}|$ . In the graph, the coset leaders represent group elements, not just the corresponding minimal expressions. There is a directed edge from  $v_i$  to  $v_j$  if  $v_j = Rv_i$  for a fundamental reflection  $R$  and  $\ell(v_j) = \ell(v_i) + 1$ . We label such an edge by  $R$ , so that the transformation  $v_j$  can be reconstructed by tracing a path from  $v_0$  to  $v_j$  and reading the edge labels in order. Such a path need not be unique, due to the group relations between generators. Relations such as  $AB = BA$  or  $CDC = DCD$  can lead to multiple paths between the vertices. However, we have just seen that every coset leader has a unique minimal expression in the dictionary order, and this determines a canonical path from  $v_0$  to  $v_j$ . We formalize this as follows.

The *spanning tree*  $T$  for the coset leader graph  $\Gamma$  has again as vertices all the coset leaders for  $\mathbf{G}$  over  $\mathbf{H}$ . There is a directed edge from  $v_i$  to  $v_j$  in  $T$  if the minimal expression for  $v_j$  is  $RE$ , where  $R$  is a fundamental reflection and  $E$  is the minimal expression for  $v_i$ . It is easy to see that this is indeed a spanning tree in the graph theoretic sense, and thus determines a unique path from  $v_0$  to each  $v_j$ .

Within the spanning tree, the alphabetic order also determines an order on the edges emanating from a given vertex. That is, if the canonical expression for  $v$  is  $E$  and  $R, S$  are fundamental reflections with  $R < S$ , then  $RE$

precedes  $SE$  whenever both those coset leaders are in the tree  $T$ . We will refer to this as the *branch order* of  $T$ .

We will use spanning trees for navigating through the group in encoding and decoding. Now we observe that the spanning tree has a technical property that is crucial to the decoding algorithm.

*Lemma 19:* Let  $u$  and  $w$  be vertices in  $\Gamma$ . If there is a path in  $\Gamma$  going from  $u$  to  $w$ , then the (unique) path from  $u$  to  $w$  in the spanning tree  $T$  goes through the successor of  $u$  that is least in the branch order and lies on some path from  $u$  to  $w$ .

In other words, if there is a reduced expression for the coset leader  $w$  of the form  $ERU$ , where  $U$  is the minimal expression for  $u$  and  $R$  is a fundamental reflection, then the minimal expression for  $w$  will be the one of this form with  $R$  least in the alphabetical order. Thus the path from  $u$  to  $w$  will go through the successor  $v$  with minimal expression  $RU$ .

In fact, it is not necessary to use the alphabetical order to determine the spanning tree and the branch order. Any spanning tree and branch order with the property of Lemma 19 will work for the decoding algorithm. However, using the alphabetical order seems to be a very natural way to attain this.

The next result is quite general. We state it for sequences of subgroups, as that is how it will be used.

*Theorem 20:* Let  $\mathbf{G}$  be a group, and let

$$\{I\} = \mathbf{G}_0 < \mathbf{G}_1 < \cdots < \mathbf{G}_{m-1} < \mathbf{G}_m = \mathbf{G}$$

be a sequence of subgroups. Choose (left) coset leaders for each  $\mathbf{G}_i$  over  $\mathbf{G}_{i-1}$ , with  $I$  as the coset leader for  $\mathbf{G}_{i-1}$ . Then every element of  $\mathbf{G}$  has a unique expression as a product of coset leaders,  $g = c_m \cdots c_1$ , with each  $c_i$  a coset leader for  $\mathbf{G}_i$  over  $\mathbf{G}_{i-1}$ .

*Proof:* If  $g \in \mathbf{G}$ , then  $g \in \mathbf{G}_k$  for some minimal  $k \leq m$ . The claim is proved by induction on  $k$ . If  $k = 0$ , then  $g = I$  and we take  $c_j = I$  for all  $j$ . So, assuming that every  $h \in \mathbf{G}_{k-1}$  has a unique expression as a product of coset leaders, let  $g \in \mathbf{G}_k$ . Then  $g = c_k h$  for some unique coset leader  $c_k$  and unique  $h = c_k^{-1} g$  in  $\mathbf{G}_{k-1}$ . For larger indices  $\ell > k$  we have  $g \in \mathbf{G}_{\ell-1}$ , so we must choose  $c_\ell = I$ , and thus the whole expression is uniquely determined. ■

The results in this section apply to an arbitrary subgroup  $\mathbf{H}$  of a finite group  $\mathbf{G}$  with a given linearly ordered set of generators  $A, B, C, \dots$ . They can even be extended to finitely generated infinite groups so long as we have algorithms to determine whether two expressions evaluate to the same element of  $\mathbf{G}$ , and whether an expression evaluates to an element of the subgroup  $\mathbf{H}$ .

## V. A COMMENT ON LENGTHS

The length of an element  $g$  in a reflection group  $\mathbf{G}$  is the minimum number of fundamental reflections in an expression for  $g$ . These fundamental reflections are of the form  $S_\alpha$  where  $\alpha$  is a fundamental root of  $\mathbf{G}$ .

If  $\mathbf{H}$  is a reflection subgroup of  $\mathbf{G}$  and  $g \in \mathbf{H}$ , then the length of a minimal expression for  $g$  as a product of fundamental reflections of  $\mathbf{H}$  may differ from  $\ell_{\mathbf{G}}(g)$ . In fact, we always have  $\ell_{\mathbf{H}}(g) \leq \ell_{\mathbf{G}}(g)$ . For by Theorem 8(2), the length of  $g$  is the number of reflecting planes separating say  $\mathbf{x}_0$  and  $g\mathbf{x}_0$ , and for a reflection subgroup, the reflecting planes of  $\mathbf{H}$  are a subset of those of  $\mathbf{G}$ .

Now consider the situation where we have a sequence of reflection subgroups

$$\{I\} = \mathbf{G}_0 < \mathbf{G}_1 < \mathbf{G}_2 < \cdots < \mathbf{G}_m = \mathbf{G}.$$

In view of Theorem 20, we regard  $g$  as a product of coset leaders,  $g = c_m \cdots c_1$ , with each  $c_i$  a coset leader for  $\mathbf{G}_i$  over  $\mathbf{G}_{i-1}$ . But each  $c_i$  is written as a product of fundamental reflections for  $\mathbf{G}_i$ , rather than  $\mathbf{G}$ . The natural length function associated with this decomposition is

$$\ell^*(g) = \sum_{i=1}^m \ell_{\mathbf{G}_i}(c_i)$$

and it is really  $\ell^*(g)$  that we should be using. However, the distinction between  $\ell(g)$  and  $\ell^*(g)$  does not come directly into play as long as we are working with coset leaders for a group  $\mathbf{G}$  over a subgroup  $\mathbf{H}$ , and then applying this to  $\mathbf{G}_i$  over  $\mathbf{G}_{i-1}$  inductively. For a sequence of parabolic subgroups,  $\ell^*(g) = \ell(g)$  holds as a consequence of Corollary 27 below, but otherwise they may differ.

The notion of a *reduced* expression should also be adjusted accordingly, so that a reduced expression for a group element  $g$  has  $\ell^*(g)$  minimal.

## VI. WHO IS MY NEIGHBOR?

Our group coding scheme is based on the premise that there is a direct connection between Euclidean distance and word length. The group code consists of  $\{g\mathbf{x}_0 : g \in \mathbf{G}\}$ . For the message associated with the group element  $g$ , we have opted to send the vector  $g^{-1}\mathbf{x}_0$ . Then we decode by finding  $g$  in terms of its left coset decomposition.

The initial vector, always denoted  $\mathbf{x}_0$ , can be chosen to be any unit vector in the fundamental region of  $\mathbf{G}$ . By a *closest neighbor* of a code point  $u\mathbf{x}_0$  we mean a vector  $v\mathbf{x}_0$  that lies in a region adjoining that of  $u\mathbf{x}_0$ , so that  $u\mathbf{x}_0$  and  $v\mathbf{x}_0$  are separated by only one reflecting

plane. If we use the optimum initial vector as given by Mittelholzer and Lahtonen [3], which we would normally do in practice, then the neighbors of  $u\mathbf{x}_0$  will all be the same distance from  $u\mathbf{x}_0$ , which is called the *minimum distance* of  $\mathbf{G}$ ; see the proof of Theorem 21.

Now the labelling of a given group element as  $g$  versus  $g^{-1}$  is arbitrary. Of course,  $\|g\mathbf{x}_0 - \mathbf{x}_0\| = \|\mathbf{x}_0 - g^{-1}\mathbf{x}_0\|$  since  $g$  is an isometry, and  $\ell(g) = \ell(g^{-1})$  since they are products of the same reflections in opposite order. Indeed, the left coset decomposition of  $g$  corresponds to to the right coset decomposition of  $g^{-1}$ . Once having made our choice of notation, though, it behooves us to keep the notation straight, and to state our results in the form in which they will be used.

In those terms, the motivating problem for this section is this: *Given a group element  $u \in \mathbf{G}$ , find the closest neighbors of  $u^{-1}\mathbf{x}_0$  in  $\{g\mathbf{x}_0 : g \in \mathbf{G}\}$ .* More particularly, we would like to know the left coset decomposition of those elements  $v \in \mathbf{G}$  such that  $v^{-1}\mathbf{x}_0$  is a closest neighbor of  $u^{-1}\mathbf{x}_0$ . This investigation requires a careful analysis of the action of coset leaders on the roots of  $\mathbf{G}$ , and on the fundamental regions of subgroups.

The basic answer to our question, ignoring canonical form, is straightforward.

*Theorem 21:* Let  $\mathbf{G}$  be a reflection group, let  $\mathbf{x}_0$  be an initial vector for  $\mathbf{G}$ , and let  $u \in \mathbf{G}$ . Then  $v^{-1}\mathbf{x}_0$  is a closest neighbor of  $u^{-1}\mathbf{x}_0$  in  $\{g\mathbf{x}_0 : g \in \mathbf{G}\}$  if and only if  $v = S_\alpha u$  for some fundamental root  $\alpha$  of  $\mathbf{G}$ .

*Proof:* The nearest neighbors of  $\mathbf{x}_0$  are the vectors  $S_\alpha\mathbf{x}_0$  for  $\alpha$  a fundamental reflection. As  $\mathbf{G}$  consists of isometries, the neighbors of  $u^{-1}\mathbf{x}_0$  are the vectors  $u^{-1}S_\alpha\mathbf{x}_0 = (S_\alpha u)^{-1}\mathbf{x}_0$ . Moreover,

$$\begin{aligned} \|S_\alpha\mathbf{x}_0 - \mathbf{x}_0\| &= \|u^{-1}S_\alpha\mathbf{x}_0 - u^{-1}\mathbf{x}_0\| \\ &= \|(S_\alpha u)^{-1}\mathbf{x}_0 - \mathbf{x}_0\| \end{aligned}$$

as desired.  $\blacksquare$

It turns out that the second neighbors of  $\mathbf{x}_0$ , i.e., those vectors  $g\mathbf{x}_0$  separated from  $\mathbf{x}_0$  by two reflecting planes, need not be equidistant even for the optimum choice of  $\mathbf{x}_0$ . In fact, for longer terms, it is possible to have  $\|g\mathbf{x}_0 - \mathbf{x}_0\| > \|g'\mathbf{x}_0 - \mathbf{x}_0\|$  even though  $g$  is shorter than  $g'$ . However, the property of the next result suffices for coding purposes.

*Theorem 22:* Let  $\mathbf{H}$  be a subgroup of a reflection group  $\mathbf{G}$ , and let  $\mathbf{x}_0$  be an initial vector for  $\mathbf{G}$ . Consider two elements  $g$  and  $g'$  of  $\mathbf{G}$  with reduced expressions  $g = Lh$  and  $g' = L'h$ , where  $L, L'$  are coset leaders and  $h \in \mathbf{H}$ . If  $L$  precedes  $L'$  in the graph  $\Gamma$  of coset leaders for  $\mathbf{G}$  over  $\mathbf{H}$ , i.e., the reduced expression for  $L'$  is of the form  $L' = KL$ , then  $\|g\mathbf{x}_0 - \mathbf{x}_0\| < \|g'\mathbf{x}_0 - \mathbf{x}_0\|$ .

*Proof:* By the remark following Theorem 6, we know that

$$\begin{aligned} \|S_\alpha g\mathbf{x}_0 - \mathbf{x}_0\| &> \|g\mathbf{x}_0 - \mathbf{x}_0\| \quad \text{iff } (\alpha, g\mathbf{x}_0) > 0 \\ &\quad \text{iff } \ell(S_\alpha g) = \ell(g) + 1. \end{aligned}$$

The length condition certainly applies when  $L$  is an immediate predecessor of  $L' = S_\alpha L$  in  $\Gamma$ , and the statement of the theorem follows by induction.  $\blacksquare$

The task before us now is to find the canonical expression for the neighbors of  $u^{-1}\mathbf{x}_0$  as a product of coset leaders, in terms of the canonical expression for  $u$ . This will be important for both the efficiency of the algorithm, and making an assignment of binary messages to group elements so that neighbors are assigned bit-strings that do not much differ.

Combining Theorems 13 and 21, and using induction, we obtain the desired characterization of the neighbors of  $u^{-1}\mathbf{x}_0$  in terms of canonical form.

*Theorem 23:* Let  $\mathbf{G}$  be a reflection group with initial vector  $\mathbf{x}_0$ . Fix a sequence of reflection subgroups

$$\{I\} = \mathbf{G}_0 < \mathbf{G}_1 < \cdots < \mathbf{G}_{m-1} < \mathbf{G}_m = \mathbf{G}.$$

If the canonical form of  $u$  as a product of coset leaders is  $c_m \dots c_1$  and  $v^{-1}\mathbf{x}_0$  is a closest neighbor of  $u^{-1}\mathbf{x}_0$ , then the canonical form of  $v$  is  $c'_m \dots c'_1$  where  $c'_i = c_i$  for all but one  $i$ , and for that index  $c'_j$  is an immediate predecessor or successor of  $c_j$  in the coset leader graph  $\Gamma_j$ .

*Proof:* If  $u$  and  $v$  are as in the theorem, then by Theorem 21,  $v = S_\alpha u$  for some fundamental reflection  $S_\alpha$ . If  $c'_m = S_\alpha c_m$ , then it has length either one more or less than the length of  $c_m$ , and the conclusion follows. But if  $c'_m = c_m$ , then by the last case of the proof of Theorem 13,  $S_\alpha c_m = c_m S_\beta$  for some fundamental reflection  $S_\beta$  of  $\mathbf{G}_{m-1}$ . The conclusion follows by induction.  $\blacksquare$

It is useful to view the decoding process in terms of fundamental regions. Recall that if  $\mathbf{H}$  is a reflection subgroup of  $\mathbf{G}$ , then  $\text{FR}(\mathbf{G}) \subseteq \text{FR}(\mathbf{H})$ . By convention,  $\text{FR}(\{I\})$  is the whole space  $\mathfrak{R}^n$ . The following result is then crucial to our algorithm.

*Theorem 24:* Let  $\mathbf{H}$  be a reflection subgroup of the reflection group  $\mathbf{G}$ . If  $\mathbf{x}$  is a vector in the fundamental region of  $\mathbf{H}$ , but not on a reflecting plane of  $\mathbf{G}$ , then there is a unique coset leader  $g$  such that  $g\mathbf{x}$  is in the fundamental region of  $\mathbf{G}$ . If  $\mathbf{x}$  is in the closure of the fundamental region of  $\mathbf{H}$  on a reflecting plane, then there is at least one coset leader  $g$  such that  $g\mathbf{x}$  is in the closure of the fundamental region of  $\mathbf{G}$ .

*Proof:* First we note that since  $\mathbf{G}$  acts faithfully and transitively on its regions, as long as  $\mathbf{x}$  is not on a

reflecting plane, then there is a unique  $g \in \mathbf{G}$  such that  $g\mathbf{x}$  is in the fundamental region of  $\mathbf{G}$ . If  $\mathbf{x}$  is on the reflecting plane  $N_\alpha$ , then  $\mathbf{x}$  is fixed by  $S_\alpha$ , and hence there will be more than one such group element  $g$ .

Now, assuming that  $\mathbf{x} \in \text{FR}(\mathbf{H})$  and  $g\mathbf{x} \in \text{FR}(\mathbf{G})$ , we will show that  $g$  is a coset leader. Let  $\beta$  be any positive root of  $\mathbf{H}$ . Then  $(g\beta, g\mathbf{x}) = (\beta, \mathbf{x}) > 0$ , and thus  $g\beta$  is a positive root of  $\mathbf{G}$ . Since  $\beta$  was arbitrary, it follows that  $\Delta_{\mathbf{H}}(g) = \emptyset$ , whence  $g$  is a coset leader. ■

It may be tempting to also conjecture that if  $\mathbf{x} \in \text{FR}(\mathbf{H})$  and  $g$  is a coset leader for  $\mathbf{G}$  over  $\mathbf{H}$ , then  $g\mathbf{x} \in \text{FR}(\mathbf{H})$ . This, however, is seldom the case.

## VII. SIMPLIFICATIONS FOR PARABOLIC SUBGROUPS

Let us note some respects in which parabolic subgroups are more well-behaved than arbitrary reflection subgroups.

*Lemma 25:* Let  $\alpha$  be a fundamental root of the parabolic subgroup  $\mathbf{H}$  and  $S_\alpha$  the corresponding reflection. Then  $S_\alpha$

- (a) reverses the sign of  $\alpha$ ,
- (b) permutes all the other positive roots of  $\mathbf{H}$  among themselves, and
- (c) permutes the positive roots of  $\mathbf{G}$  that are not roots of  $\mathbf{H}$  among themselves.

*Proof:* Items (a) and (b) follow from Theorem 4 and the fact that  $\mathbf{H}$  is a group. Because  $\mathbf{G}$  is a group, all its positive roots except  $\alpha$  are permuted, but if every root of  $\mathbf{H}$  is carried into another root of  $\mathbf{H}$ , each root of  $\mathbf{G}$  that is not a root of  $\mathbf{H}$  must be carried into another root that is not a root of  $\mathbf{H}$ . ■

*Theorem 26:* If  $h$  is an element of the parabolic subgroup  $\mathbf{H}$ , then  $h$  permutes the positive roots of  $\mathbf{G}$  that are not roots of  $\mathbf{H}$  among themselves. Thus  $|\Delta_{\mathbf{H}}^*(h)| = 0$ .

*Proof:* If  $\alpha$  is a root and  $S_\alpha$  a reflection of  $\mathbf{H}$ , then by the lemma above, the roots of  $\mathbf{G}$  that are not in  $\mathbf{H}$  are permuted, and therefore  $|\Delta_{\mathbf{H}}^*(h)| = |\Delta_{\mathbf{H}}^*(S_\alpha h)|$ . Every element of  $\mathbf{H}$  can be written as a product of fundamental reflections of  $\mathbf{H}$ , while  $|\Delta_{\mathbf{H}}^*(I)| = 0$ . The theorem follows by induction on the number of factors in  $h$ . ■

*Corollary 27:* Let  $\mathbf{G}$  be a reflection group and  $\mathbf{H}$  a parabolic subgroup, and let  $g$  be a coset leader and  $h \in \mathbf{H}$ . Then  $\ell(gh) = \ell(g) + \ell(h)$ .

*Proof:* In general, for an element  $x \in \mathbf{G}$  we have  $\ell(x) = |\Delta_{\mathbf{G}}(x)| = |\Delta_{\mathbf{H}}(x)| + |\Delta_{\mathbf{H}}^*(x)|$  by Theorem 8. Now we consider the action of  $gh$  on the roots of  $\mathbf{G}$ , combining the characterizations of Theorems 12 and 26, the latter of which is valid only for parabolic subgroups. Because  $g$  is a coset leader, we have  $|\Delta_{\mathbf{H}}(g)| = 0$ ,

whence  $|\Delta_{\mathbf{H}}(gh)| = |\Delta_{\mathbf{H}}(h)| = |\Delta_{\mathbf{G}}(h)|$ . On the other hand,  $|\Delta_{\mathbf{H}}^*(gh)| = |\Delta_{\mathbf{H}}^*(g)| = |\Delta_{\mathbf{G}}(g)|$ . Therefore

$$\begin{aligned} \ell(gh) &= |\Delta_{\mathbf{H}}(gh)| + |\Delta_{\mathbf{H}}^*(gh)| = |\Delta_{\mathbf{G}}(h)| + |\Delta_{\mathbf{G}}(g)| \\ &= \ell(h) + \ell(g) \end{aligned}$$

as claimed. ■

## VIII. BINARY SCHEMES

An important factor in a good group code is the correspondence between messages and group elements. For the sake of simplicity, let us take the messages to be binary strings of a fixed length. In this section, we show how to efficiently map these binary strings onto group elements. This mapping reveals useful not only for encoding, but also for decoding and retrieving the original bit stream from the decoded group element. It can be noted that while the decoding methods of [1], [2], [3] allow one to efficiently determine the most likely group element from a noisy received vector, they do not consider the demapping problem and often, a lookup table approach to solve this problem may become as costly as the decoding itself. It can also be observed that in a communication system, modulation based on group codes is likely to be used in conjunction with channel coding. Hence it seems appropriate to consider group codes as signal sets of concatenated codes. Such approaches are beyond the scope of this paper, which in this context, still remains relevant to bit interleaved coded modulation.

In the fixed-length case, the correspondence takes bit strings of length  $x$ , where  $x \leq \lfloor \log_2 |\mathbf{G}| \rfloor$ , and maps them one-to-one to a subset consisting of some  $2^x$  elements of  $\mathbf{G}$ . Let  $\gamma : 2^x \rightarrow \mathbf{G}$  denote the assignment map, which will be used for encoding. For decoding, we select a left inverse map  $\gamma^{-1} : \mathbf{G} \rightarrow 2^x$  so that  $\gamma^{-1}\gamma(\mathbf{m}) = \mathbf{m}$  for all  $\mathbf{m} \in 2^x$ .

A good correspondence should have three basic properties.

- 1) If  $u^{-1}\mathbf{x}_0$  and  $v^{-1}\mathbf{x}_0$  are closest neighbors in  $\{g\mathbf{x}_0 : g \in \mathbf{G}\}$ , then  $\gamma^{-1}(u)$  and  $\gamma^{-1}(v)$  should differ by as few bits as possible. It may not be possible to arrange that the strings corresponding to closest neighbors always differ by one bit, but in fact it is practical to get the average bit-difference for neighbors fairly close to one bit.
- 2) In general,  $\gamma\gamma^{-1}(g) = g$  will hold only if  $g$  is in the range of  $\gamma$ . If it is not, then  $\gamma^{-1}(g)$  should be chosen so that  $\gamma\gamma^{-1}(g)$  is close to  $g$  geometrically. Ideally,  $\gamma^{-1}(g)$  would be  $\gamma^{-1}(g')$  for a  $g'$  in the range of  $\gamma$  chosen to minimize  $\|(g')^{-1}\mathbf{x}_0 - g^{-1}\mathbf{x}_0\|$ .

- 3) Both  $\gamma$  and  $\gamma^{-1}$  should be relatively easy to implement. These will be implemented repeatedly in coding and decoding, respectively. Trying to satisfy the first two properties can create some complications, but an effort should be made to keep the process efficient. For example, we want to avoid extensive look-up tables.

The motivation for the first two conditions is that we will be sending messages in the form of vectors  $g^{-1}\mathbf{x}_0$  with  $g \in \mathbf{G}$ . Decoding will be done by finding the transformation  $g' \in \mathbf{G}$  that moves the received vector  $\mathbf{r}$  closest to the initial vector  $\mathbf{x}_0$ , i.e., the  $g'$  that minimizes  $\|g'\mathbf{r} - \mathbf{x}_0\|$ . The most likely error will be to incorrectly decode  $u^{-1}\mathbf{x}_0$  as one of its neighbors  $v^{-1}\mathbf{x}_0$ . If the corresponding bit strings differ by only a few bits, then super-imposing an error-correcting code will correct most of these errors.

To construct the correspondence, we take our sequence of reflection subgroups

$$\{I\} = \mathbf{G}_0 < \mathbf{G}_1 < \mathbf{G}_2 < \dots < \mathbf{G}_m = \mathbf{G}$$

and write each element  $T$  of  $\mathbf{G}$  as a product of coset leaders. Now geometric neighbors are described by Theorem 23; their expressions differ by exactly one coset leader, and the differing pair are neighbors in some  $\Gamma_j$ . So the initial plan would be to partition the bit strings into sections corresponding to the coset leader graphs  $\Gamma_1, \dots, \Gamma_m$ . If we do this directly, then we can encode only bit strings of length  $\sum_{j=1}^m \lceil \log_2 |\Gamma_j| \rceil$ , which is generally less than the capacity of the group  $\lceil \log_2 |\mathbf{G}| \rceil$ . In order to gain extra bits, we will combine various pairs of indices. That is, for pairs of graphs  $\Gamma_i$  and  $\Gamma_j$  with say  $2^s < |\Gamma_i| < 2^{s+1}$  and  $2^t < |\Gamma_j| < 2^{t+1}$ , but  $|\Gamma_i| \times |\Gamma_j| > 2^{s+t+1}$ , we can make the assignment using the direct product graph  $\Gamma_i \times \Gamma_j$ , thereby gaining one bit of information.

We do not have a general scheme for finding good binary correspondences, but in practice it was not hard to come up with fairly good *ad hoc* assignments for various specific cases. Note that we must use a sequence of *reflection* subgroups in making the binary correspondence. Theorems 13 and 23 are only valid for reflection subgroups. Other intermediate subgroups can and will be used in the process of encoding and decoding, but some care is required, and the coset leader graphs for non-reflection subgroups should not be used in the binary assignment.

## IX. DESCRIPTION OF THE ALGORITHM

The outline of our group coding algorithm is straightforward. The setup involves selecting a particular reflection group  $\mathbf{G}$  and a sequence of its subgroups. The coset

leader graph and its spanning tree are determined, and a binary correspondence is established. All these things appear as subroutines or parameters in the implementation.

The message  $\mathbf{m}$  to be sent corresponds to a group element  $g = \gamma(\mathbf{m})$ . The element  $g$  has a canonical expression  $g = c_m \dots c_1$  as a product of coset leaders. We transmit the vector

$$\mathbf{x} = g^{-1}\mathbf{x}_0 = c_1^{-1} \dots c_m^{-1}\mathbf{x}_0,$$

The expression for each  $c_i$  as a product of reflections is obtained from the coset leader tree  $\Gamma_i$ , and these combine to give the expression for  $g$ . The expression for  $c_i^{-1}$  is obtained by going backwards through the coset leader tree, and these are combined in reverse order to give  $g^{-1}$ .

The received vector has the form  $\mathbf{r} = \mathbf{x} + \mathbf{n}$  where  $\mathbf{n}$  represents channel noise. Hopefully,  $\mathbf{r}$  will be in the same region as the transmitted vector  $\mathbf{x}$ , or if not, in a neighboring region. We decode by finding the sequence of coset leaders  $d_1, \dots, d_m$  such that  $d_i \dots d_1 \mathbf{r}$  is in the fundamental region of the subgroup  $\mathbf{G}_i$ . This in turn is done by going through the coset leader trees and applying reflections, so that at each step the vector obtained is closer to  $\mathbf{x}_0$ , and hence to the fundamental region of  $\mathbf{G}$ , than the preceding vector. Thus the final vector  $d_m \dots d_1 \mathbf{x}$  is in the fundamental region of  $\mathbf{G}$ . We decode by taking  $g' = d_m \dots d_1$  and the received message as  $\mathbf{m}' = \gamma^{-1}(g')$ .

For the remainder of this section, we discuss some of the details of implementation.

### A. Setup

In order to implement encoding and decoding for a specific group  $\mathbf{G}$  acting on  $\mathbb{R}^n$  with a given subgroup sequence, we begin by choosing a fundamental root system  $\rho_1, \dots, \rho_n$  for  $\mathbf{G}$ . If there are intermediate subgroups, then we also need to identify vectors that are fundamental roots of those subgroups, say  $\sigma_1, \dots, \sigma_k$ . Assume these vectors are normalized to unit length.

At this point we also write the procedures to implement the reflections  $S_\alpha(\mathbf{x}) = \mathbf{x} - 2(\mathbf{x}, \alpha)\alpha$  for  $\alpha = \rho_1, \dots, \rho_n, \sigma_1, \dots, \sigma_k$ . These transformations are given by orthogonal matrices, but most of times they will be permutation matrices or permutations with sign changes. It is clearly advantageous to apply those transformations to the vector directly, reserving matrix multiplication for those few cases where the matrix is not sparse.

Next we find the optimum initial vector  $\mathbf{x}_0$ , lying in the middle of a fundamental region. An algorithm for this process is described in Mittelholzer and Lahtonen [3].

First form the matrix  $R$  whose rows are the root vectors  $\rho_i$ . Find the inverse matrix  $R^{-1}$ , and let  $\mathbf{v}$  be the vector whose  $j$ -th entry is the sum of the  $j$ -th row of  $R^{-1}$ . Then normalize,  $\mathbf{x}_0 = \frac{\mathbf{v}}{\|\mathbf{v}\|}$ .

Now, using the algorithm of section IV, find the spanning tree of coset leaders for each successive pair of subgroups in the sequence. While the whole coset leader graph is very useful for understanding the structure of the group, its spanning tree is all we need for encoding and decoding. Indeed, the simplest way we know to find the whole graph is to decode all the terms  $S_\alpha g$  with  $\alpha$  a fundamental root and  $g$  a coset leader.

### B. Encoding: message to coset leaders

Now assume that we are given a binary message  $\mathbf{m} = m_1 \dots m_x$  of length  $x$ . Using our predetermined scheme, we invoke a procedure to convert  $\mathbf{m}$  to a string of coset leaders  $c_m \dots c_1$ , which in turn corresponds uniquely to an element  $g$  of  $\mathbf{G}$ . In the simple case where we are not combining subgroups, just break  $\mathbf{m}$  into blocks so that

$$\begin{aligned} m_1 \dots m_r &\mapsto c_1 \\ m_{r+1} \dots m_s &\mapsto c_2 \\ &\dots \\ m_{t+1} \dots m_x &\mapsto c_m. \end{aligned}$$

If we are combining subgroups, then some substrings will correspond to pairs  $\langle c_i, c_j \rangle$ , and the procedure may become fairly involved. The output is the string  $\mathbf{c}$  of coset leaders.

### C. Encoding: coset leaders to vector

We want to transmit the vector  $\mathbf{x} = g^{-1}\mathbf{x}_0 = c_1^{-1} \dots c_m^{-1}\mathbf{x}_0$ . This is done by working our way backwards through the spanning trees of the coset leader graphs for  $\mathbf{G}_i$  over  $\mathbf{G}_{i-1}$ .

In practice, then, to encode we first apply to  $\mathbf{x}_0$  the reflection corresponding to the vertex  $c_m$  in the graph  $\Gamma_m$ , and then move to its predecessor in the spanning tree for  $\Gamma_m$  and apply its reflection to the vector we have, continuing until we reach the root of the tree, which corresponds to the identity  $I$ . Then we move to the vertex  $c_{m-1}$  in the graph  $\Gamma_{m-1}$  and apply its reflection and move to its predecessor, and so on until we reach the root of  $\Gamma_1$ . Then we transmit the vector  $\mathbf{x}$  thus obtained.

### D. Noise

Presumably, at this point the vector goes through a noisy channel, so that the received vector is  $\mathbf{r} = \mathbf{x} + \mathbf{n}$ . In testing, we simulate this by adding to each component

of  $\mathbf{x}$  separately a value generated by pseudo-random generator for a Gaussian distribution with mean 0 and variance  $s^2$ , for various values of  $s$ .

### E. Decoding: vector to coset leaders

Upon receiving the vector  $\mathbf{r}$ , we want to find the transformation  $g'$  that will take  $\mathbf{r}$  into the fundamental region of  $\mathbf{G}$ , or equivalently, to find  $g'$  that will minimize  $\|g'\mathbf{r} - \mathbf{x}_0\|$ . Recall that since

$$\{I\} = \mathbf{G}_0 < \mathbf{G}_1 < \dots < \mathbf{G}_m = \mathbf{G}$$

we have

$$\mathfrak{R}^n = \text{FR}(I) \supseteq \text{FR}(\mathbf{G}_1) \supseteq \dots \supseteq \text{FR}(\mathbf{G}).$$

So we want recursively to call a procedure that will take a vector in the fundamental region of one subgroup, and in view of Theorem 24, find the coset leader that moves it into the fundamental region of the next. Let us describe the procedure in those terms.

So let  $\mathbf{H}$  be a reflection subgroup of a reflection group  $\mathbf{G}$ , and let  $T$  be the spanning tree for the coset leader graph  $\Gamma$  of  $\mathbf{G}$  over  $\mathbf{H}$ . We are given a vector  $\mathbf{x}$  in the fundamental region of  $\mathbf{H}$ , and we want to find the coset leader  $g$  such that  $\mathbf{z} = g\mathbf{x}$  is in the fundamental region of  $\mathbf{G}$ . Unless  $\mathbf{x}$  is on a reflecting plane, this  $g$  is unique. If  $\mathbf{x}$  is on a reflecting plane, then the procedure yields a minimum-length  $g$  such that  $g\mathbf{x}$  is on the boundary of the fundamental region.

At each stage of the procedure, we have a pair  $\langle u, \mathbf{y} \rangle$  where  $u$  is a coset leader, corresponding to a vertex of  $T$ , and  $\mathbf{y} = u\mathbf{x}$ . We start at the root of the tree with the pair  $\langle I, \mathbf{x} \rangle$ . Suppose we have reached the vertex  $u$  in  $T$ , so that we have the pair  $\langle u, \mathbf{y} \rangle$ . Test whether there is a successor  $v = S_\alpha u$  of  $u$  in  $T$  for which  $(\alpha, \mathbf{y}) < 0$ . If there is such a successor  $v$ , choose the one that is least in the branch order, and replace  $\langle u, \mathbf{y} \rangle$  by  $\langle v, S_\alpha \mathbf{y} \rangle$ . Note that  $S_\alpha \mathbf{y} = S_\alpha u \mathbf{x} = v\mathbf{x}$ . If there is no such successor (either because  $u$  has no successor in  $T$  or because the successors fail the property), take  $g = u$  and  $\mathbf{z} = \mathbf{y}$ , and terminate the procedure.

In section XI, we will prove that the procedure works as intended.

Applying the procedure to  $\mathbf{G}_1$  over  $\mathbf{G}_0$  with  $\mathbf{x}$ , we obtain  $d_1\mathbf{x}$  in the fundamental region of  $\mathbf{G}_1$ . Then applying it to  $\mathbf{G}_2$  over  $\mathbf{G}_1$  starting with  $d_1\mathbf{x}$ , we obtain  $d_2d_1\mathbf{x}$  in the fundamental region of  $\mathbf{G}_2$ . After  $m$  applications, we have  $d_m \dots d_1\mathbf{x}$  in the fundamental region of  $\mathbf{G}_m = \mathbf{G}$ , as desired. Then we take  $g' = d_m \dots d_1$ .

The preceding discussion depends on Theorem 22, which is valid only for reflection subgroups. Nor is there an obvious modification that applies to arbitrary

intermediate subgroups. However, there is one important situation, which arises repeatedly, that we can handle. This involves an intermediate subgroup generated by a set of reflections and an element  $R$  satisfying  $R^2 = I$ . In this case, there is no root corresponding to  $R$  so we check instead whether  $\|R\mathbf{y} - \mathbf{x}_0\| < \|\mathbf{y} - \mathbf{x}_0\|$ , which is in itself no problem. It can happen, though, that there are transformations  $T \in \mathbf{G}$  such that  $\|RT\mathbf{x}_0 - \mathbf{x}_0\| = \|T\mathbf{x}_0 - \mathbf{x}_0\|$ , making it impossible to decide which coset leader to choose. The situation can be resolved by allowing two expressions to represent certain coset leaders. Rather than discuss this situation abstractly, we will illustrate it in Section XIII for the groups  $D_n$ , which are typical.

#### F. Decoding: coset leaders to received message

Then we reverse the encoding process, using the functions  $\gamma^{-1}$  to obtain the received message  $\mathbf{m}' = m'_1 \dots m'_x$  from the coset leaders  $d_m \dots d_1$ . The details of implementation can get complicated, but in principle this is straightforward.

### X. CODES WITH ISOTROPY GROUPS

Modifications are required to deal systematically with the case when the initial vector  $\mathbf{x}_0$  has a nontrivial stabilizer in the reflection group  $\mathbf{G}$ . In general, the set of elements  $g$  of a group  $\mathbf{G}$  such that  $g\mathbf{x}_0 = \mathbf{x}_0$  is a subgroup of  $\mathbf{G}$  called the *isotropy group* or *stabilizer group* of  $\mathbf{x}_0$ , denoted  $\text{Stab}(\mathbf{x}_0)$ . Mittelholzer and Lahtonen, and also Slepian, considered codes with nontrivial stabilizers. Let us begin with some of the analysis found in Mittelholzer and Lahtonen, then develop the theory generalizing the case when the stabilizer is  $\{I\}$ .

If  $\mathbf{x}_0$  lies in the interior of one of the regions of  $\mathbf{G}$ , then its stabilizer is trivial. The isotropy group will be nontrivial exactly when  $\mathbf{x}_0$  lies on one or more of the reflecting planes  $N_\alpha$ . Since  $\mathbf{G}$  is transitive on its regions, we may choose the fundamental region so that  $\mathbf{x}_0$  is in the closure of the fundamental region. In that case, Theorem 3 of [3] shows that  $\text{Stab}(\mathbf{x}_0)$  is the parabolic subgroup generated by the fundamental reflections  $S_\alpha$  such that  $\mathbf{x}_0$  is in the reflecting plane  $N_\alpha$ .

Now the roots of  $\mathbf{G}$  divide into three classes:

$$\begin{aligned} \Delta_{\mathbf{G}}^+ &= \{\alpha \in \Delta_{\mathbf{G}} : (\alpha, \mathbf{x}_0) > 0\} \\ \Delta_{\mathbf{G}}^0 &= \{\beta \in \Delta_{\mathbf{G}} : (\beta, \mathbf{x}_0) = 0\} \\ \Delta_{\mathbf{G}}^- &= \{\gamma \in \Delta_{\mathbf{G}} : (\gamma, \mathbf{x}_0) < 0\}. \end{aligned}$$

The fundamental root system  $\Sigma$  is contained in the former two sets. Theorem 5 of [3] is that the maximum likelihood decoding region of  $\mathbf{x}_0$  is given by

$$\text{MLDR}(\mathbf{x}_0) = \{\mathbf{y} : (\mathbf{y}, \alpha) \geq 0 \text{ for all } \alpha \in \Sigma\}.$$

With our standard assumption that  $\mathbf{x}_0$  is in the closure of the fundamental region of  $\mathbf{G}$ ,

$$\text{MLDR}_{\mathbf{G}}(\mathbf{x}_0) = \bigcup_{h \in \text{Stab}(\mathbf{x}_0)} h(\overline{\text{FR}(\mathbf{G})}).$$

In general, this will not be the fundamental region of a subgroup, so adjustments must be made in both the theory and the encoding/decoding algorithms.

For the purposes of induction, we also need the decoding region for subgroups:

$$\text{MLDR}_{\mathbf{H}}(\mathbf{x}_0) = \bigcup_{h \in \text{Stab}(\mathbf{x}_0)} h(\overline{\text{FR}(\mathbf{H})}).$$

These are the fundamental regions when the stabilizer subgroup is trivial, and play the analogous role in the more general case.

#### A. Description of the algorithm

The setup involves selecting a particular reflection group  $\mathbf{G}$ , an initial vector  $\mathbf{x}_0$ , and a sequence of its subgroups  $\{I\} = \mathbf{G}_0 \leq \mathbf{G}_1 < \dots < \mathbf{G}_m = \mathbf{G}$  with  $\mathbf{G}_1 = \text{Stab}(\mathbf{x}_0)$ . Find the coset leader graphs and their spanning trees, this time for the *right* cosets  $\mathbf{H}g$  of the subgroups in the sequence. (Either side can be used when the isotropy group is trivial, but now it makes a difference.) Set up a correspondence  $\gamma$  between messages and elements of  $\mathbf{G}$  that is constant on right cosets of  $\mathbf{G}_1$ .

The message  $\mathbf{m}$  to be sent corresponds to a right coset  $\mathbf{G}_1g$  of the stabilizer. Any such representative element  $g$  has a canonical expression  $g = c_1 \dots c_m$  as a product of right coset leaders. We transmit the vector

$$\begin{aligned} \mathbf{x} &= g^{-1}\mathbf{x}_0 = c_m^{-1} \dots c_2^{-1} c_1^{-1} \mathbf{x}_0 \\ &= c_m^{-1} \dots c_2^{-1} \mathbf{x}_0. \end{aligned}$$

The expression for each  $c_i$  as a product of reflections is obtained from the coset leader trees  $\Gamma_i$ , and the expression for  $g^{-1}$  is obtained by going backwards through the coset leader trees.

The received vector has the form  $\mathbf{r} = \mathbf{x} + \mathbf{n}$ . Hopefully,  $\mathbf{r}$  will be in the same decoding region as the transmitted vector  $\mathbf{x}$ , or if not, in a neighboring region. We decode by finding the sequence of coset leaders  $d_m, \dots, d_2$  such that  $d_i \dots d_m \mathbf{r}$  is in the decoding region of the subgroup  $\mathbf{G}_i$ . This in turn is done by going through the coset leader trees and applying reflections, so that at each step the vector obtained is closer to  $\mathbf{x}_0$  than the preceding vector. Thus the final vector  $\mathbf{z} = d_2 \dots d_m \mathbf{x}$  is in the decoding region of  $\mathbf{G}$ . Note that, for any element  $h$  of the stabilizer,  $\|h\mathbf{z} - \mathbf{x}_0\| = \|\mathbf{z} - \mathbf{x}_0\|$ . We decode by taking  $g' = d_2 \dots d_m$  and the received message as  $\mathbf{m}' = \gamma^{-1}(g')$ .

## B. Analysis

The trouble with this algorithm is that it doesn't work as stated. The fact that  $\|\mathbf{S}_\alpha \mathbf{z} - \mathbf{x}_0\| = \|\mathbf{z} - \mathbf{x}_0\|$  whenever  $S_\alpha$  is in the isotropy group means that one cannot navigate those edges of the coset leader graphs: distance cannot be used to determine coset leaders. This problem shows up already in the six-element group  $A_2$  with a two-element stabilizer. There seem to be two straightforward options for dealing with the situation.

(1) Choose a set of codewords  $g^{-1}\mathbf{x}_0$  whose decoding avoids such problem edges. This is essentially what Mittelholzer and Lahtonen did, and it yields some interesting codes.

(2) Alternately, one can choose an arbitrary parabolic subgroup  $\mathbf{H}$  of  $\mathbf{G}$ , and choose  $\mathbf{x}_0$  in the fundamental region  $\text{FR}(\mathbf{G})$ . In that case, there are no problem edges in the sense above. Select a subgroup sequence  $\{I\} = \mathbf{G}_0 \leq \mathbf{G}_1 < \dots < \mathbf{G}_m = \mathbf{G}$  with  $\mathbf{G}_1 = \mathbf{H}$ , and proceed as above, just encoding and decoding from  $\mathbf{G}_1$  to  $\mathbf{G}$ . The maximum likelihood decoding region is still

$$\text{MLDR}_{\mathbf{G}}(\mathbf{x}_0) = \bigcup_{h \in \mathbf{H}} h(\overline{\text{FR}(\mathbf{G})})$$

which, for a parabolic subgroup  $\mathbf{H}$ , is a union of adjacent regions. These types of codes would be a good topic for further investigation.

## XI. A PROOF THAT THE DECODING ALGORITHM ACHIEVES MAXIMUM LIKELIHOOD DECODING

In this section, let us show that the decoding procedure described in subsection IX-E actually works. Given are a reflection subgroup  $\mathbf{H}$  of a reflection group  $\mathbf{G}$ , the spanning tree  $T$  for the coset leader graph, and a vector  $\mathbf{x}$  in the closure of the fundamental region of  $\mathbf{H}$ . By Theorem 24, there is a coset leader  $g$  such that  $g\mathbf{x}$  is in the closure of the fundamental region of  $\mathbf{G}$ . The coset leader  $g$  is unique if  $\mathbf{x}$  is not on a reflecting plane. If perchance  $\mathbf{x}$  is on a reflecting plane, we choose  $g$  to be such an element of minimal length.

Now we know that there is a unique path to  $g$  in the tree  $T$ , and that if there are multiple paths to  $g$  in the graph  $\Gamma$ , then at each option the path in  $T$  takes the edge leading towards  $g$  that is least in the branch order (Lemma 19). We want to show that the procedure follows this path to the vertex  $g$ .

Suppose we are at the vertex (coset leader)  $u$  in the procedure. Note that it begins at the vertex  $I$  with  $\ell(I) = 0$ . Thus we may assume inductively that  $\ell(u) \leq \ell(g)$  and that  $\ell(ug^{-1}) = \ell(g) - \ell(u)$ . It must be shown that, unless  $u = g$ , there is a fundamental root  $\alpha$  such that  $v = S_\alpha u$  is a successor of  $u$  in  $T$  and  $\ell(vg^{-1}) = \ell(g) - \ell(u) - 1$ .

Assuming  $u \neq g$ , then  $u\mathbf{x}$  is not in the closure of the fundamental region, and hence  $(\alpha, u\mathbf{x}) < 0$  for some fundamental root  $\alpha$ . Let us first show that  $S_\alpha u$  is indeed a coset leader for any such  $\alpha$ . By Theorem 13, the other possibility is that  $S_\alpha u \in u\mathbf{H}$ , whence  $u^{-1}S_\alpha u = S_{u^{-1}\alpha}$  is in  $\mathbf{H}$ . That would make  $u^{-1}\alpha$  a root of  $\mathbf{H}$ . Now  $(u^{-1}\alpha, \mathbf{x}) = (\alpha, u\mathbf{x}) < 0$  and  $\mathbf{x}$  is in the fundamental region of  $\mathbf{H}$ , so  $u^{-1}\alpha$  is a negative root of  $\mathbf{H}$ . But  $u$  carries  $u^{-1}\alpha$  into  $\alpha$ , which is a positive root. Thus  $\Delta_{\mathbf{H}}(u)$  is nonempty, which contradicts its being a coset leader. We conclude that  $S_\alpha u$  must itself be a coset leader.

To compute  $\ell(S_\alpha u g^{-1})$ , we apply Lemma 5. If  $g\mathbf{x}$  is in the interior of the fundamental region, use the initial vector  $\mathbf{x}_0 = g\mathbf{x}$ , while if  $g\mathbf{x}$  is on the boundary choose  $\mathbf{x}_0$  in the interior very close to  $g\mathbf{x}$ . Since  $(\alpha, u\mathbf{x}) \approx (\alpha, ug^{-1}\mathbf{x}_0) < 0$ , we have  $\ell(S_\alpha u g^{-1}) = \ell(ug^{-1}) - 1$ . Thus  $S_\alpha u$  is closer to  $g$  in the graph  $\Gamma$  than  $u$  is, so that in going from  $u$  to  $S_\alpha u$  we are moving towards  $g$ . Moreover, it follows that  $v = S_\alpha u$  is a successor of  $u$ , i.e. that  $\ell(v) = \ell(u) + 1$ , since  $\ell(vg^{-1}) \geq \ell(g) - \ell(v)$ .

The edge from  $u$  to  $v = S_\alpha u$  may not be in the spanning tree  $T$  for every fundamental root with  $(\alpha, u\mathbf{x}) < 0$ , but it is so for the edge used in the canonical expression for  $g$ , by Lemma 19. Among the candidates for  $v = S_\alpha u$ , both the canonical path and the procedure choose the one that it is least in the branch (alphabetical) order.

Finally, the algorithm does indeed terminate if  $u = g$ , for in that case  $u\mathbf{x}$  is in the closure of the fundamental region of  $\mathbf{G}$ , and hence  $(\alpha, u\mathbf{x}) \geq 0$  for every fundamental root  $\alpha$ . Note that  $(\alpha, g\mathbf{x}) = 0$  occurs when  $g\mathbf{x}$  is on the reflecting plane  $N_\alpha$ ; otherwise  $(\alpha, g\mathbf{x}) > 0$ .

Thus at each step, the algorithm moves through the spanning tree towards the desired coset leader  $g$ , and terminates there.

## XII. COUNTING COMPARISONS IN THE ALGORITHM

The basic step iterated in the decoding algorithm is to test whether  $\|S_\alpha \mathbf{y} - \mathbf{x}_0\| < \|\mathbf{y} - \mathbf{x}_0\|$ , or equivalently, whether  $(\mathbf{y}, \alpha) < 0$ . So a good measure of the efficiency of the algorithm, for a given group and sequence of subgroups, is to count the average number of such comparisons in decoding. We also count the maximum number of comparisons for various group coding schemes.

In this section, the method of counting comparisons will be established. This will be applied in the next section, where we consider encoding and decoding with various specific reflection groups. These concrete cases allow us to illustrate and analyze the algorithm in some detail. In particular, we will see that a good choice

of the subgroup sequence can significantly improve the efficiency of decoding.

Our counts are based on two simple mathematical observations. The first gives the method of counting, while the second just says that averages add appropriately.

*Lemma 28:* Let  $\mathbf{H}$  be a subgroup of a reflection group  $\mathbf{G}$ . Let  $\Gamma$  be the coset leader graph for  $\mathbf{H}$  over  $\mathbf{H}$ , with spanning tree  $T$ . For an element  $g = ch$  of  $\mathbf{G}$ , with  $c \in \Gamma$  a coset leader and  $h \in \mathbf{H}$ , let  $\nu(h)$  be the number of comparisons to determine  $h$ ,  $\lambda(c)$  the number of comparisons to determine  $c$ , and  $\tau(g)$  the total number of comparisons to determine  $g$ . Then  $\tau(g) = \nu(h) + \lambda(c)$ . Moreover,  $\lambda(c)$  can be determined recursively in  $T$  as follows.

- 1)  $\lambda(I)$  is the number of successors of  $I$  in  $T$ .
- 2) If  $b$  is the predecessor of  $c$  in  $T$ , and  $b$  has  $m$  successors, of which  $c$  is  $\ell$ -th in the branch order, and  $c$  has  $s$  successors, then  $\lambda(c) = \lambda(b) - m + \ell + s$ .

*Lemma 29:* Let  $\mathbf{H}$  be a subgroup of a reflection group  $\mathbf{G}$ . Let

$$\begin{aligned}\bar{\nu} &= \frac{1}{|\mathbf{H}|} \sum_{h \in \mathbf{H}} \nu(h) \\ \bar{\lambda} &= \frac{|\mathbf{H}|}{|\mathbf{G}|} \sum_{c \in \Gamma} \lambda(c) \\ \bar{\tau} &= \frac{1}{|\mathbf{G}|} \sum_{g \in \mathbf{G}} \tau(g)\end{aligned}$$

be the average values of these parameters. Then  $\bar{\tau} = \bar{\nu} + \bar{\lambda}$ .

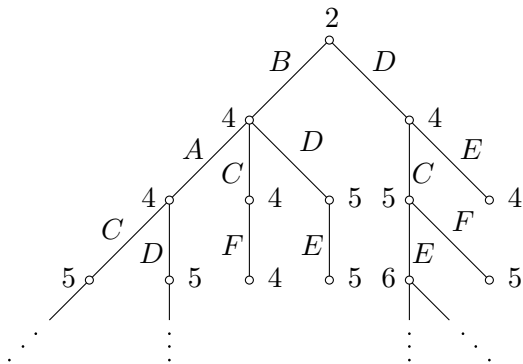


Fig. 1. Top of the coset leader tree for  $E_6$  over  $D_5$

The method of Lemma 28 is illustrated in Figure 1, which gives the top few layers of the spanning tree for the coset leader graph of  $E_6$  over a subgroup isomorphic to  $D_5$ . (There are several such reflection subgroups.) The edge labels are included to establish the branch order, alphabetically. Each vertex  $c$  is labelled with  $\lambda(c)$ , the number of comparisons required to establish whether the coset leader is  $c$ .

### XIII. SPECIFIC GROUPS

In this section, we show how the algorithm works with various specific groups. Subgroup sequences and coset leader graphs are given for most of the finite irreducible reflection groups, the graphs for  $I_4$ ,  $E_7$  and  $E_8$  being too large to fit here. Our analysis will show that the algorithm can be made more efficient by an appropriate choice of the subgroup sequence.

#### A. $A_n$

The groups  $A_n$  have a particularly simple structure, making them a good example with which to begin.  $A_n$  is isomorphic to the symmetric group on  $n+1$  letters, and it is convenient to represent it as acting on the subspace  $\sum x_i = 0$  of  $\mathbb{R}^{n+1}$ . In particular,  $A_n$  has size  $(n+1)!$ .

For a concrete example, we consider  $A_4$ , which has order  $5! = 120$ . The Coxeter graph for  $A_4$  is given in Figure 2.

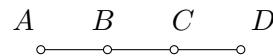


Fig. 2. Coxeter graph for  $A_4$

Let us begin with the parabolic subgroup sequence

$$I < A_1 < A_2 < A_3 < A_4$$

where  $A_1$  is the subgroup generated by  $\{A\}$ , the subgroup  $A_2$  is generated by  $\{A, B\}$ , etc. Note that our ordering of the parabolic subgroups is implicit in the labelling of the Coxeter graph, and this choice is not unique.

The coset leader graphs  $\Gamma_1$  to  $\Gamma_4$  are given in Figure 3. The vertices of  $\Gamma_k$  are labelled  $0, \dots, k$ ; to avoid cluttering the figure we have labelled only  $\Gamma_4$ . Also it is understood that the edges are always directed downward, from vertex  $j$  to  $j+1$ .

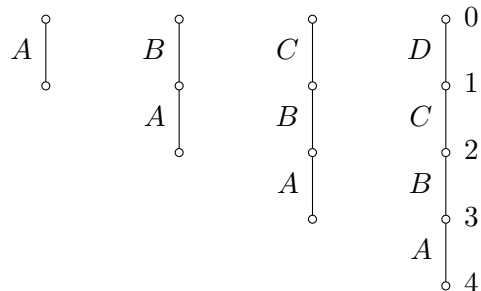


Fig. 3. Coset leader graphs  $\Gamma_k$  for  $A_4$

The elements of the group  $A_n$  are then identified by sequences  $\mathbf{d} = (d_1, d_2, d_3, d_4)$  with each  $d_i$  a vertex in  $\Gamma_i$ . For example, the sequence  $(1, 0, 2, 2)$  corresponds to the product (from right to left)  $CD \cdot BC \cdot I \cdot A = CDBCA$ .

Now we assign binary sequences to the first  $2^x$  vertices of  $\Gamma_k$ , where  $x = \lfloor \log_2(|\Gamma_k|) \rfloor = \lfloor \log_2(k+1) \rfloor$ , which is the first two vertices of  $\Gamma_1$  and  $\Gamma_2$ , and the first four vertices of  $\Gamma_3$  and  $\Gamma_4$ . If we do this using a Gray map, so that neighboring vertices have binary sequences differing in only one bit, we obtain the assignment shown in Figure 4.

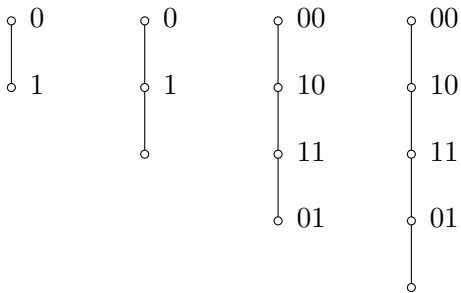


Fig. 4. Binary sequence assignments for  $A_4$

Then we can use 64 elements of our 120-element group to encode six-bit messages. We think of a message as being broken into four parts

$$m_1 \quad m_2 \quad m_3 m_4 \quad m_5 m_6$$

associated with the four graph labellings. For example, the message  $m = 101111$  corresponds to the vertex sequence  $(1, 0, 2, 2)$ , which as we saw above represents the element  $CDBCA$ .

Of course, the choice  $n = 4$  gives us a rather small example, but in practice it is not much harder to do larger  $n$ . The only complication is that in order to encode at or near the maximum number of bits  $\lfloor \log_2(|\mathbf{G}|) \rfloor$ , we must combine the binary assignments for pairs of graphs, as mentioned earlier. But for the permutation groups  $A_n$ ,  $B_n$  and  $D_n$  this process is quite straightforward, and we have written the programs for  $n$  a power of 2 up to  $n = 2^5 = 32$ , without much difficulty.

Counting the number of comparisons in decoding  $A_n$ , using the parabolic subgroup sequence, is a straightforward application of Lemmas 28 and 29. It is somewhat simpler to work with  $\sum_{g \in \mathbf{G}} \tau(g)$  and divide by  $\|\mathbf{G}\|$  later. Let  $a_n = \sum_{g \in A_n} \tau(g)$  using the parabolic subgroup sequence described above. The recursion for the totals is

$$a_1 = 2$$

$$a_n = n! \frac{n^2 + 3n}{2} + (n+1)a_{n-1}$$

which in turn yields a recursion for the averages:

$$\bar{a}_1 = 1$$

$$\bar{a}_n = \frac{n^2 + 3n}{2n+2} + \bar{a}_{n-1}.$$

Consequently  $\bar{a}_n$  is asymptotically  $\frac{n^2}{4}$ . The maximum number of comparisons in decoding  $A_n$  this way is  $\frac{n^2+n}{2}$ .

However, it turns out that we can decode much more efficiently if we choose a different sequence of reflection subgroups. The subgroups used are still isomorphic to  $A_k$  for  $k \leq n$ , but we choose the copies of those subgroups that have short coset leader trees. Let us illustrate this method with  $n = 6$ , the generalization being straightforward.

The following observations will be applied recursively.

*Lemma 30:* Assume that  $X, Y, Z$  satisfy the generating relations for  $A_3$ :  $X^2 = Y^2 = Z^2 = (XY)^3 = (YZ)^3 = (XZ)^2 = I$ . Let  $Y' = ZYZ = YZY$ . Then  $Y'^2 = (XY')^3 = I$ , so that  $X, Y'$  generate a subgroup isomorphic to  $A_2$ .

*Corollary 31:* Assume that  $X, Y, Z, T$  satisfy the generating relations for  $A_4$ , adding  $T^2 = (ZT)^3 = (XT)^2 = (YT)^2 = I$  to the relations of Lemma 30. Let  $Y' = ZYZ = YZY$ . Then  $X, Y', T$  generate a subgroup isomorphic to  $A_3$ .

These can be proved by straightforward calculations, or by noting that the subgroups in question are conjugates of parabolic subgroups.

Now consider  $A_6$  with the standard generators  $A, B, C, D, E, F$ . Form the following sequence of subgroups, given in terms of their generators:

$$\mathbf{G}_0 = \{I\}$$

$$\mathbf{G}_1 = \text{Sg}(N)$$

$$\mathbf{G}_2 = \text{Sg}(A, M)$$

$$\mathbf{G}_3 = \text{Sg}(A, L, F)$$

$$\mathbf{G}_4 = \text{Sg}(A, B, K, F)$$

$$\mathbf{G}_5 = \text{Sg}(A, B, J, E, F)$$

$$\mathbf{G}_6 = \text{Sg}(A, B, C, D, E, F) = A_6$$

where

$$J = CDC = DCD$$

$$K = JEJ = EJE$$

$$L = BKB = KBK$$

$$M = LFL = FLF$$

$$N = AMA = MAM.$$

By induction using Lemma 30 and Corollary 31, each  $G_k$  is isomorphic to  $A_k$ . (Alternatively, each of these subgroups is a conjugate of a parabolic subgroup.) The

coset leader graphs, however, are different and shorter. These are given in Figure 5. The indicated branch order, from left to right, in this case is not always alphabetical.

To implement the algorithm with this subgroup sequence, we would need the positive roots corresponding to these reflections. These are easily found using Theorem 2.

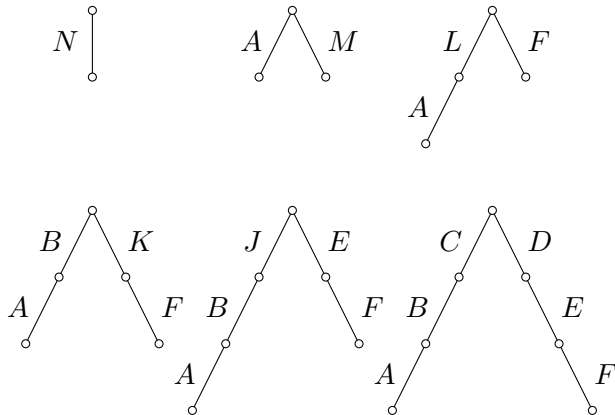


Fig. 5. Coset leader trees for a better  $A_6$

Now let us count the number of comparisons in decoding  $A_n$  using this subgroup sequence. Let  $k = \lfloor \frac{n}{2} \rfloor$ , so that  $n = 2k$  or  $2k + 1$ . The recursion for the average number of comparisons is

$$\begin{aligned} \bar{a}'_1 &= 1 \\ \bar{a}'_n &= \bar{a}'_{n-1} + \frac{k^2 + 4k}{n+1} \quad \text{for } n \text{ even} \\ \bar{a}'_n &= \bar{a}'_{n-1} + \frac{k^2 + 5k + 2}{n+1} \quad \text{for } n \text{ odd.} \end{aligned}$$

Asymptotically, the average number of comparisons tends to  $\frac{n^2}{8}$ , which is half the number for the parabolic subgroup sequence. The maximum number of comparisons this way is  $\lfloor \frac{n^2+4n}{4} \rfloor$ , again roughly half.

Thus there is a significant improvement in decoding efficiency to be had by choosing the subgroup sequence with shorter coset leader trees. This phenomenon will hold for decoding other groups as well. Table II gives some values for  $\bar{a}_n$ ,  $\bar{a}'_n$  and, for comparison, the theoretical lower bound for the average,  $\log_2(n+1)!$  (see Knuth [11], p. 194).

## B. $B_n$

The groups  $B_n$  have larger size for the same dimension ( $2^n n!$  versus  $(n+1)!$  for  $A_n$ ) and a refined subgroup sequence, making them a better candidate for group coding, especially for larger  $n$ . Again, to make a specific example we choose  $n = 4$ . The Coxeter graph for  $B_4$  is given in Figure 6.

TABLE II  
AVERAGE NUMBER OF COMPARISONS TO DECODE  $A_n$

$n$	$\bar{a}_n$	$\bar{a}'_n$	$\log_2(n+1)!$
4	7.7	7.1	6.9
8	24.2	19.5	18.4
16	81.6	57.4	48.3
32	292.9	182.2	122.7

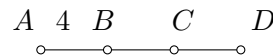


Fig. 6. Coxeter graph for  $B_4$

The coset leader graph for the sequence of parabolic subgroups

$$I < B_1 < B_2 < B_3 < B_4$$

is given in Figure 7. This provides a reasonably good group coding scheme.

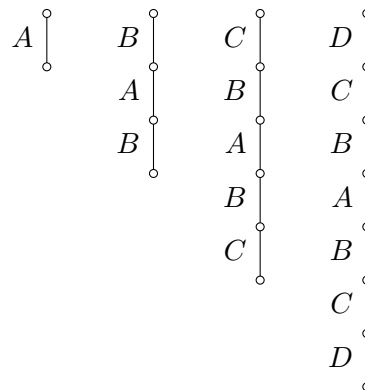


Fig. 7. Coset leader graphs  $\Gamma_k$  for  $B_4$

We can do much better, however, by using the refined subgroup sequence

$$I < B_1 < M_2 < B_2 < M_3 < B_3 < M_4 < B_4$$

where  $M_k$  is the subgroup generated by  $B_{k-1}$  and the longest coset leader  $R_k$  for  $B_k$  over  $B_{k-1}$ . In fact,  $R_k$  is a reflection, so  $R_k^2 = I$ , and it commutes with the elements of  $B_{k-1}$ , whence  $M_k \cong B_{k-1} \times A_1$ .

Using the reflections  $R_2 = BAB$ ,  $R_3 = CBABC$  and  $R_4 = DCBABCD$  the coset leader graph for the refined sequence is given in Figure 8. The coset leaders for  $B_k$  over  $M_k$  correspond to building terms from  $I$ , or reducing them from  $R_k$ , depending on whether their length is more or less than half that of the longest expression.

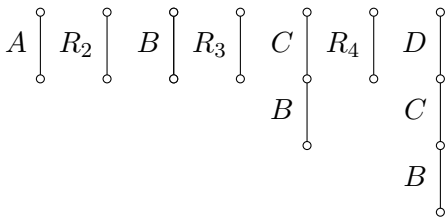


Fig. 8. Coset leader graphs for the refined sequence of  $B_4$

Now one can apply the techniques used on  $A_n$  to this sequence, to obtain a similar sequence with shorter coset leader trees, and hence improved decoding efficiency.

Consider the following subgroups given in terms of their generators:

$$\begin{aligned} \mathbf{G}_0 &= \{I\} \\ \mathbf{G}_1 &= \text{Sg}(A) \\ \mathbf{N}_2 &= \text{Sg}(A, R'_2) \\ \mathbf{G}_2 &= \text{Sg}(A, K) \\ \mathbf{N}_3 &= \text{Sg}(A, K, R'_3) \\ \mathbf{G}_3 &= \text{Sg}(A, B, J) \\ \mathbf{N}_4 &= \text{Sg}(A, B, J, R'_4) \\ \mathbf{G}_4 &= \text{Sg}(A, B, C, D) \end{aligned}$$

where

$$\begin{aligned} J &= CDC = DCD \\ K &= BJB = JBJ \\ R'_2 &= KAK \\ R'_3 &= JKAKJ \\ R'_4 &= DJKAKJD. \end{aligned}$$

Calculations in the vein of Lemma 30 show that we still have  $G_k \cong B_k$  and  $N_k \cong B_{k-1} \times A_1$ , but now the coset leader graphs are those given in Figure 9.

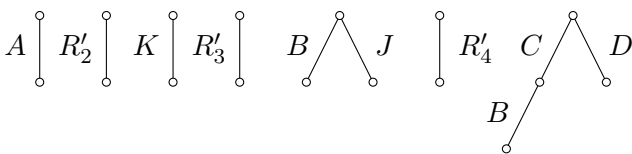


Fig. 9. Best coset leader graphs for  $B_4$

It is straightforward to see how this scheme can be extended for larger  $n$ . For  $n \geq 6$  we will again want to combine the cosets in assigning the bit-strings. Since the complexity of encoding and decoding is roughly proportional to the sum of the lengths of the coset leader graphs, for larger  $n$  we start to realize a noticeable gain in efficiency by using the intermediate subgroups and the shorter trees.

TABLE III  
AVERAGE NUMBER OF COMPARISONS TO DECODE  $B_n$

$n$	$\bar{b}_n$	$\bar{b}'_n$	$\bar{b}''_n$	$n + \log_2 n!$
4	11.0	8.9	8.7	8.6
8	38.6	27.3	24.0	23.3
16	142.3	88.6	67.7	60.3
32	542.0	307.9	204.5	149.7

To quantify this, we again count the number of comparisons in decoding. The recursions for the average number of comparisons used in the three different methods are

$$\begin{aligned} \bar{b}_1 &= \bar{b}'_1 = \bar{b}''_1 = 1 \\ \bar{b}_n &= \frac{2n^2 + n - 1}{2n} + \bar{b}_{n-1} \\ \bar{b}'_n &= \frac{n^2 + 3n - 2}{2n} + \bar{b}'_{n-1} \\ \bar{b}''_n &= \bar{a}'_{n-1} + n. \end{aligned}$$

Some values for these averages, along with the theoretical lower bound, are given in Table III. Asymptotically, the average number of comparisons to decode  $B_n$  with the three methods are  $\frac{n^2}{2}$ ,  $\frac{n^2}{4}$  and  $\frac{n^2}{8}$ , respectively. The maximum numbers of comparisons are  $n^2$ ,  $\frac{n^2+n}{2}$  and  $\lfloor \frac{n^2+2n-3}{4} \rfloor + n$ , respectively.

### C. $D_n$

The groups  $D_n$  are in many ways similar, but introduce a few new features. They are also useful as building blocks for the groups  $E_6$ ,  $E_7$  and  $E_8$ . The cardinality of  $D_n$  is  $2^{n-1}n!$ , so for the same dimension we lose a bit and gain a little in terms of distance. The structure of  $D_4$  is atypically symmetric, so we illustrate with  $D_5$ . The Coxeter graph for  $D_5$  is given in Figure 10.

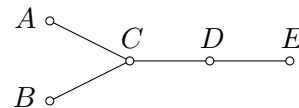
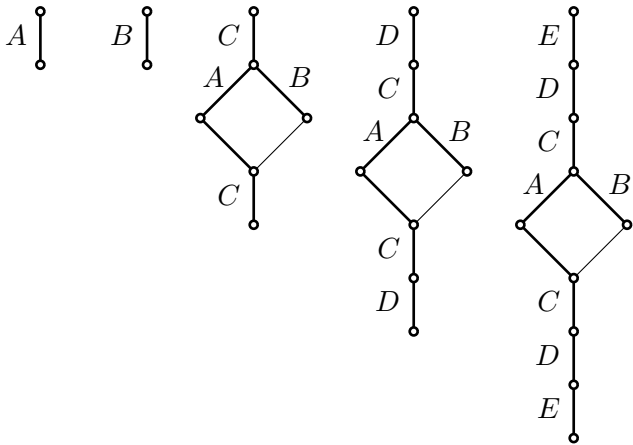


Fig. 10. Coxeter graph for  $D_5$

The corresponding coset leader graphs for the sequence of parabolic subgroups is in Figure 11. The fact that  $AB = BA$  makes the graph nonlinear. In this event, we do not label the lower edges, which carry the same value as their opposite (transposed) edge. The spanning trees  $T_k$  are indicated by darker lines.

Similar to the case with  $B_n$ , the group  $D_n$  has a refining subgroup sequence. For  $D_5$  this is

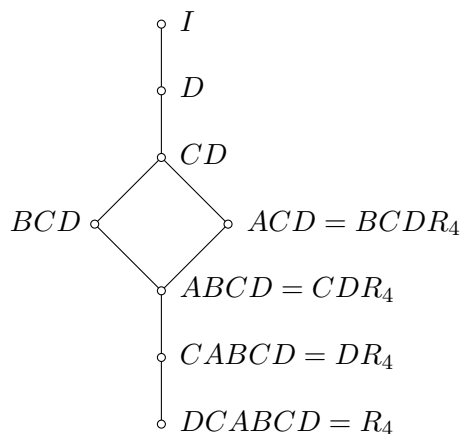
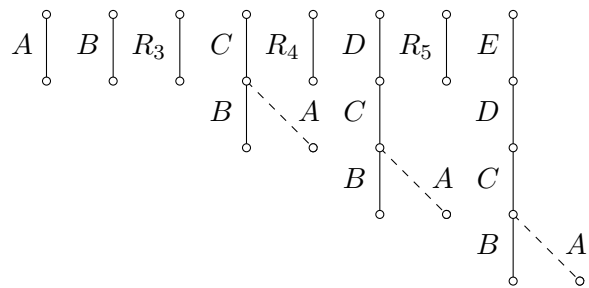
$$I < D_1 < D_2 < N_3 < D_3 < N_4 < D_4 < N_5 < D_5$$

Fig. 11. Coset leader graphs  $\Gamma_k$  for  $D_5$ 

where  $N_k$  is the subgroup generated by  $D_{k-1}$  and the longest coset leader  $R_k$  for  $D_k$  over  $D_{k-1}$ . Since  $A$  and  $B$  commute, the refinement starts with dimension 3. The longest coset leaders  $R_k$  in  $\Gamma_k$  for our case are  $R_3 = CAB$ ,  $R_4 = DCABCD$ , and  $R_5 = EDCABCDE$ . They satisfy  $R_k^2 = I$ , but they are rotations rather than reflections, being the product of an even number of reflections. Nor do they commute with the elements of  $D_{k-1}$ . For example,

$$AR_3 = ACABC = CACBC = CABCB = R_3B.$$

So  $N_k$  is not a direct product. Nonetheless, they work fine for encoding and decoding purposes, once we make a minor adjustment. Figure 12, which labels the vertices rather than the edges of  $\Gamma_4$ , illustrates pretty well what is going on. Figure 13 gives the coset leaders for the refined sequence, ignoring the dashed lines temporarily.

Fig. 12. Example with  $\Gamma_4$  for  $D_4$ Fig. 13. Coset leader graphs for the refined sequence of  $D_5$ 

Consider the vector  $\mathbf{x} = DC\mathbf{A}\mathbf{x}_0 = (ACD)^{-1}\mathbf{x}_0$  which we would use to encode the element  $ACD = BCD R_4$ , the latter being the representation as a product of coset leaders of the refined sequence. However,

$$\begin{aligned} \|\mathbf{x} - \mathbf{x}_0\| &= \|DC\mathbf{A}\mathbf{x}_0 - \mathbf{x}_0\| \\ &= \|DCB\mathbf{x}_0 - \mathbf{x}_0\| \text{ by symmetry} \\ &= \|R_4DC\mathbf{A}\mathbf{x}_0 - \mathbf{x}_0\| \\ &= \|R_4\mathbf{x} - \mathbf{x}_0\|. \end{aligned}$$

So in decoding  $\mathbf{x}$ , even with exact arithmetic, you wouldn't know whether to take  $R_4$  as part of the coset leader or not. Indeed, the scheme indicated by Figure 13 would use  $BCD$  and  $BCDR_4$  for elements that could also be represented as  $ACDR_4$  and  $ACD$ , respectively. Adding roundoff error and noise just make the situation worse. The easy solution is to allow both representations for all such middle elements. Thus we have added the *faux* coset leaders, indicated by dashed lines.

With this adjustment, the only differences between the coset leader graphs for the refined sequence of  $D_n$  and that of  $B_n$  are the one extra subgroup  $M_2$  for  $B_n$ , and the duplicate coset leaders in  $D_n$ . Then, with exactly the same type of substitutions, this sequence can be replaced by a sequence of isomorphic subgroups with short coset leader trees. The number of comparisons used for these methods of decoding  $D_n$  are thus just one less than the corresponding counts for decoding  $B_n$ , ignoring the duplicate coset leaders which add a negligible amount.

#### D. Note

For the permutation groups  $A_n$ ,  $B_n$  and  $D_n$  we can compare our method of group coding (using subgroups and coset leaders) with Slepian's method (using sorting methods for the permutation).

For the group  $A_n$ , the simplest implementation is to choose the generators so that  $A$  interchanges the first two elements of a vector,  $B$  interchanges the second and third elements, etc. (all in  $\mathfrak{R}^{n+1}$ ). Then our original algorithm implements a standard insertion sort, and the improved

version is a modified insertion sort that works from the middle of the list outward.

With the group  $B_n$ , the standard choice of generators is such that  $A$  changes the sign of the first entry,  $B$  interchanges the first two entries,  $C$  interchanges the second and third entries, etc. Then  $R_k$  ( $k \geq 2$ ) changes the sign of the  $k$ -th entry, and these sign changes are super-imposed on the insertion sort.

The group  $D_n$  is similar, except that  $A$  changes the sign of the first two entries, and  $R_k$  ( $k \geq 3$ ) changes the signs of the first and  $k$ -th entries.

In general, with permutation group codes we have the option of using fast sorting algorithms. On the other hand, the coset leaders give us a natural way to represent and keep track of the permutations, which for large  $n$  is a real issue. These ideas could be combined: other insertion-type sorts can be implemented by navigating through the coset leader graph in a different order. This is straightforward for the coset leader graphs that are linear. Thus one can adapt these ideas to use a binary insertion sort for the decoding of permutation groups. The average number of comparisons required to sort  $A_n$  this way is very close to the lower bound  $\log_2(n+1)!$ . For large  $n$  this is a significant gain, though for smaller  $n$ , Table II shows that the simpler algorithm is reasonably close to the bound.

*E.  $E_6$ ,  $E_7$  and  $E_8$*

The group  $E_6$  has cardinality  $2^7 3^4 5 = 51,840$ . Its Coxeter graph is given in Figure 14. The group  $E_6$  can be considered to be an extension of either  $D_5$  or  $A_5 \times A_1$ . Let us analyze and compare these possibilities.

The parabolic subgroup  $P = \text{Sg}(B, C, D, E, F)$  is isomorphic to  $D_5$ , and has index 27. The coset leader graph for this subgroup is given in Figure 15. In the figure, the darker lines again indicate a spanning tree (based on the order  $F < B < C < D < E < A$ ).

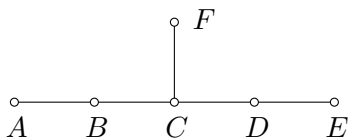


Fig. 14. Coxeter graph for  $E_6$

The coset leader graph for  $E_6$  over  $P$  has length 16. The various conjugate subgroups  $gPg^{-1}$  of  $P$  are of course all isomorphic to  $D_5$ , but the coset leader graphs for  $E_6$  over these conjugates have lengths varying from 8 to 16. Three of these have length 8, including the conjugate by  $g_0 = DCFEDCBA$ . The coset leader

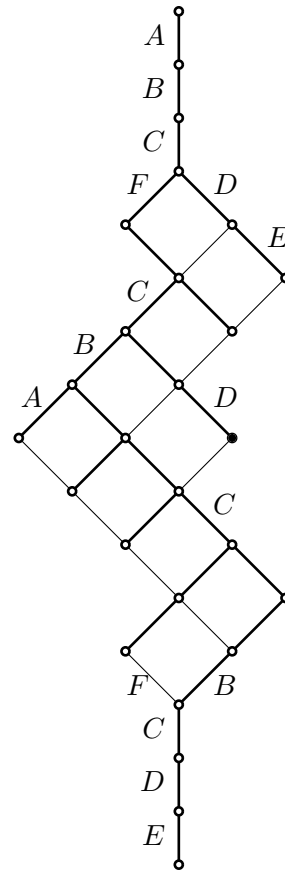


Fig. 15. Coset leader graph for  $E_6$  over a parabolic subgroup  $D_5$

graph for  $E_6$  over  $g_0Pg_0^{-1}$  is shown in Figure 16. This can be interpreted as folding the original coset leader graph at the vertex  $g_0$ , indicated by a solid circle in Figure 15.

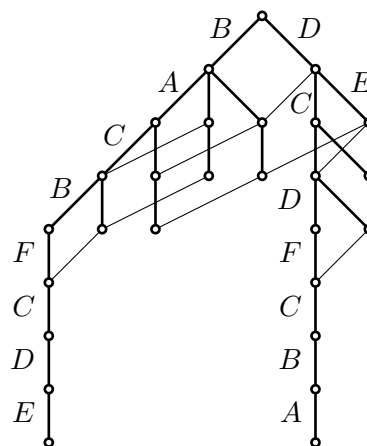


Fig. 16. Shorter coset leader graph for  $E_6$  over a conjugate of  $D_5$

Navigating the graph of Figure 16 requires an average of 5.6 comparisons, with a maximum of 8 comparisons. Adding these values to the optimum ones for  $D_5$ , we see that  $E_6$  can be decoded with an average of 16.7

TABLE IV  
COMPARISON COUNTS FOR DECODING  $E_6, E_7, E_8$

Group	Subgroup	Index	Length	Average	Maximum
$E_6$	$D_5$	27	8	16.7	20
	$A_5 \times A_1$	36	5	16.2	19
$E_7$	$E_6$	56	14	24.2	35
	$D_6 \times A_1$	63	8	22.6	26
	$A_7$	72	8	22.6	29
$E_8$	$D_8$	135	11	31.3	39

comparisons, and a maximum of 20 with this method.

The parabolic subgroup  $Q = \text{Sg}(A, B, C, D, E)$  of  $E_6$  is isomorphic to  $A_5$ . It has a unique longest coset leader

$$R = \text{FCBDCFE DCBABCDEF CDBCF}$$

and the subgroup  $U = \text{Sg}(A, B, C, D, E, R)$  is isomorphic to  $A_5 \times A_1$ . This is easily verified using the roots for these expressions.

The subgroup  $U$  has index 36, and the coset leader graph for  $E_6$  over  $U$  has length 10. There are 36 conjugates  $gUg^{-1}$  of  $U$ , and the length of the coset leader graphs for  $E_6$  over these conjugates varies from 5 to 10. Exactly one of these has length 5, that being the subgroup  $g_1Ug_1^{-1}$  with  $g_1 = \text{BCDEF CDBCF}$ .

Navigating the coset leader graph for  $E_6$  over  $g_1Ug_1^{-1}$  requires an average of 5.4 comparisons, with a maximum of 7 comparisons. Adding these values to the optimum ones for  $A_5 \times A_1$  yields the result that  $E_6$  can be decoded with an average of 16.2 comparisons, and a maximum of 19 with this method. These numbers are marginally better than those for decoding using  $D_5$ . Also, the indices for this sequence of subgroups are slightly better suited for the binary assignments. The coset leader graph for this case is rather messy, and can be found on our website: [www.math.hawaii.edu/~jb/codes.html](http://www.math.hawaii.edu/~jb/codes.html).

The group  $E_7$  has size  $2^{10}3^45 \cdot 7 = 2,903,040$ . It has  $E_6$  as a parabolic subgroup of index 57. The Coxeter graph for  $E_7$  over  $E_6$ , with the labelling we used, is given in Figure 17, and the resulting coset leader graph is given in Figure 18. The longest coset leader for the parabolic subgroup has length 22, but there are conjugate subgroups of type  $E_6$  with coset leader graphs of length 11. One of these is given in Figure 19.

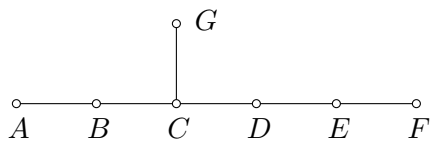


Fig. 17. Coxeter graph for  $E_7$  over  $E_6$

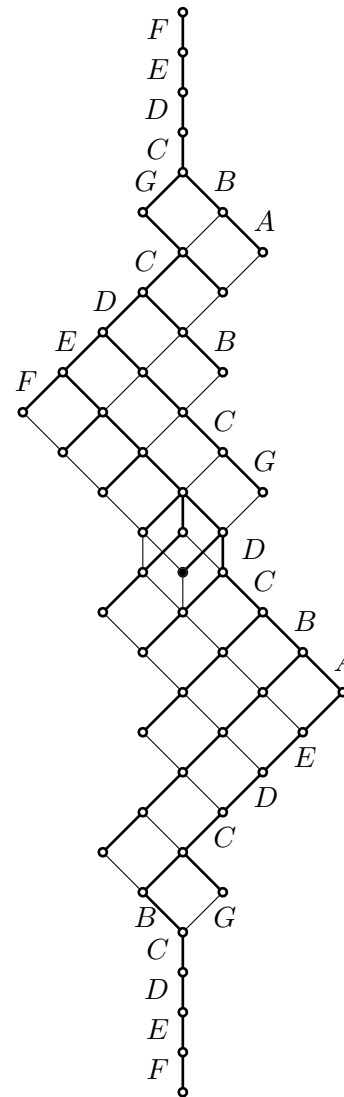


Fig. 18. Coset leader graph for  $E_7$  over  $E_6$

Implementation of coding and decoding for  $E_7$  using the subgroup  $E_6$  was straightforward. In retrospect, Table IV shows that subgroups of type  $A_7$  should yield the best results for decoding  $E_7$ . The coset leader graph for this case is also on our website.

In principle, the group  $E_8$  with cardinality  $2^{14}3^55^27 = 696,729,600$  should be only slightly more complicated, but we have not yet attended to the details. The main problem is to find a good binary assignment for the coset leader graph of  $E_8$  over  $D_8$ , say, which seems to be the most promising subgroup to use. The program output to construct the coset leader graph is again on our website. (Jiajia Seffrood helped with our analysis of  $E_8$  over  $E_7 \times A_1$ , a less likely candidate for coding.)

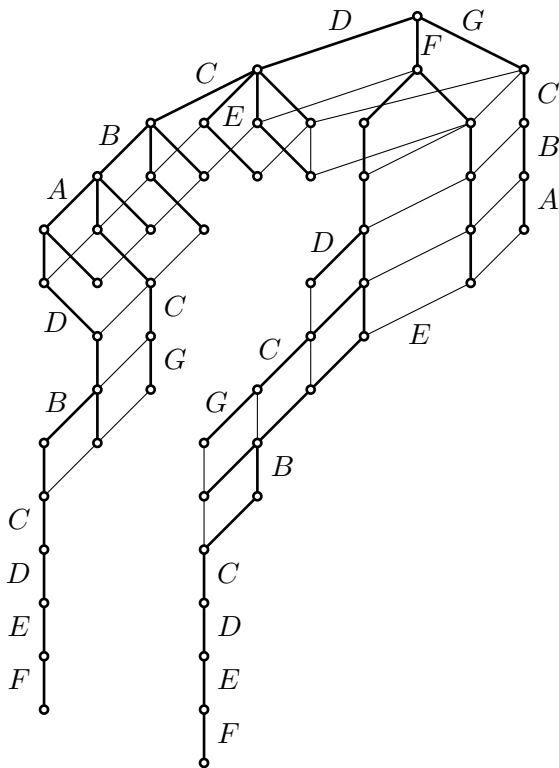


Fig. 19. Shorter coset leader graph for  $E_7$  over  $E_6$

#### F. $F_4$

The exceptional group  $F_4$  has size  $2^7 3^2 = 1152$ , with the Coxeter graph given in Figure 20. It is an extension of  $B_4$  by a group of order three. This fact is realized concretely in the sequence of subgroups used for coding:

$$I < B_1 < M_2 < B_2 < M_3 < B_3 < M_4 < B_4 < F_4$$

where

$$\begin{aligned} B_1 &= \text{Sg}(A) \\ M_2 &= \text{Sg}(A, R_2) \\ B_2 &= \text{Sg}(A, B) \\ M_3 &= \text{Sg}(A, B, R_3) \\ B_3 &= \text{Sg}(A, B, C) \\ M_4 &= \text{Sg}(A, B, C, X) \\ B_4 &= \text{Sg}(A, B, C, Y) \\ F_4 &= \text{Sg}(A, B, C, D) \end{aligned}$$

using the reflections

$$\begin{aligned} R_2 &= BAB \\ R_3 &= CBABC \\ X &= DABACBADABCABAD \\ Y &= DABAD. \end{aligned}$$

Note that  $X$  and  $Y$  are conjugates of  $D$  and  $B$ , respectively, and thus indeed reflections.

For encoding and decoding we use the simple coset leader graphs for the intermediate subgroups, shown in Figure 21. With this scheme, decoding requires an average of 10.6 comparisons, and a maximum of 12. An almost negligible improvement can be obtained by using conjugate subgroups.

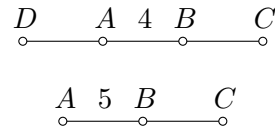


Fig. 20. Coxeter graphs for  $F_4$  (top) and  $I_3$  (bottom)

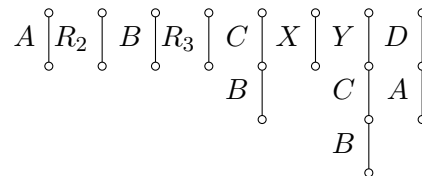


Fig. 21. Coset leaders for  $F_4$  over  $B_3$

#### G. $I_3$ and $I_4$

The exceptional group  $I_3$  has size 120. Like  $F_4$ , it has an interesting subgroup structure. In particular,  $I_3$  contains the dihedral group  $H_2^5$  as a subgroup of index 12. The Coxeter graph for  $I_3$  is given in Figure 20. For encoding and decoding  $I_3$  we can use a sequence of subgroups

$$I < A_1 < H_2^5 < K_3 < L_3 < I_3$$

where

$$\begin{aligned} A_1 &= \text{Sg}(A) \\ H_2^5 &= \text{Sg}(A, B) \\ K_3 &= \text{Sg}(A, B, R) \\ L_3 &= \text{Sg}(A, B, S) \\ I_3 &= \text{Sg}(A, B, C) \end{aligned}$$

where  $R$  and  $S$  are given by

$$\begin{aligned} R &= CBABACBABC \\ S &= CBABC. \end{aligned}$$

Note that  $S$  is a reflection, but  $R$  is not, and we have the same sort of coset leader ambiguity as occurs in  $D_n$ , which is resolved similarly. (The coset leader graph for

$I_3$  over  $H_2^5$  resembles that for  $D_6$  over  $D_5$ .) The coset leader graphs used for encoding and decoding  $I_3$  with this sequence of subgroups are given in Figure 22.

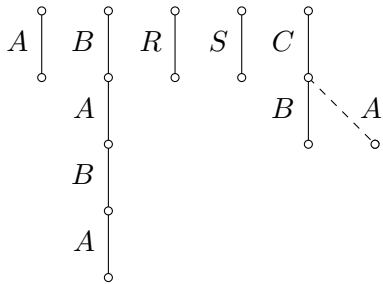


Fig. 22. Coset leader graphs for the refined sequence of  $I_3$

The coset leader graph for  $I_4$  over  $I_3 \times A_1$  would be of similar complexity to that of  $E_7$  over  $E_6$ , say, but we have not done that case yet.

#### REFERENCES

- [1] D. Slepian, *Group codes for the Gaussian channel*, Bell Syst. Tech. J. **47** (1968), 575–602.
- [2] D. Slepian, *Permutation modulation*, Proc. IEEE **53** (1965), 228–236.
- [3] T. Mittelholzer and J. Lahtonen, *Group codes generated by finite reflection groups*, IEEE Trans. on Information Theory, **42** (1996), 519–528.
- [4] T. Niyomsataya, A. Miri and M. Nevins, *Affine reflection group codes*, IEEE Trans. on Information Theory, **54** (2008), 441–454.
- [5] L.C. Grove and C.T. Benson, *Finite Reflection Groups* (GTM 99), New York, Springer-Verlag, 1985.
- [6] J.E. Humphreys, *Reflection Groups and Coxeter Groups* (Cambridge Studies in Advanced Mathematics, vol. 29), Cambridge, UK, Cambridge Univ. Press, 1990. Springer-Verlag, 1985.
- [7] R. Kane, *Reflection Groups and Invariant Theory*, CMS Books in Mathematics, New York, Springer-Verlag, 2001.
- [8] M. Dyer, *On parabolic closures in Coxeter groups*, available on the website [www.nd.edu/~dyer/papers/parclos.pdf](http://www.nd.edu/~dyer/papers/parclos.pdf).
- [9] J. Lahtonen, *The group codes from the finite reflection groups and their decoding*, Livres des Resumes, EUROCODE '94 at the Abbey of La Bussiere, France, 49–54.
- [10] A. Björner and F. Brenti, *Combinatorics of Reflection Groups* (GTM 231), New York, Springer-Verlag, 2005.
- [11] D. Knuth, *Searching and Sorting, The Art of Computer Programming*, vol. 3, Reading, MA, Addison-Wesley, 1973.