Discrete Mathematics 309 (2009) 4253–4265

Contents lists available at ScienceDirect

Discrete Mathematics

journal homepage: www.elsevier.com/locate/disc

Error-correcting codes from permutation groups

Robert F. Bailey

School of Mathematics and Statistics, Carleton University, 1125 Colonel By Drive, Ottawa, Ontario K1S 5B6, Canada

ARTICLE INFO

Article history:
Received 1 March 2006
Received in revised form 29 May 2008
Accepted 30 December 2008
Available online 29 January 2009

Keywords:
Error-correcting code
Permutation group
Base
Covering design
Graph decomposition

ABSTRACT

We replace the usual setting for error-correcting codes (i.e. vector spaces over finite fields) with that of permutation groups. We give an algorithm which uses a combinatorial structure which we call an *uncovering-by-bases*, related to covering designs, and construct some examples of these. We also analyse the complexity of the algorithm.

We then formulate a conjecture about uncoverings-by-bases, for which we give some supporting evidence and prove for some special cases. In particular, we consider the case of the symmetric group in its action on 2-subsets, where we make use of the theory of graph decompositions. Finally, we discuss the implications this conjecture has for the complexity of the decoding algorithm.

1. Introduction: Groups as codes

In this paper, we discuss the possible use of permutation groups as error-correcting codes, where the codewords are permutations written in list form and with the usual Hamming distance. The use of sets (rather than groups) of permutations in coding theory has been studied since the 1970s (see Blake, Cohen and Deza (1979) [7] for instance); often sets of permutations are referred to as *permutation arrays* in this context. Permutation arrays have attracted recent interest, partly due to a potential application to so-called "powerline communications", where electrical power cables are used to transmit data as well as electricity. The 2004 paper by Chu, Colbourn and Dukes [11] gives a description of this, and some constructions for permutation arrays suitable for this purpose, while the 2006 paper by Huczynska [15] gives an introductory survey.

Groups have received less attention; however, the algebraic structure of the group is there to be exploited, for instance in determining properties of the group when viewed as a code. The main focus of this paper is to present a decoding algorithm which works for arbitrary permutation groups when used as codes in this manner; as a result, this paper takes a rather broad viewpoint. We also analyse the complexity of this algorithm, then give a conjecture which (if true) helps to bound this complexity. We conclude by giving some supporting evidence for the conjecture, and by proving some special cases. The results in this paper are taken from the author's Ph.D. thesis [1].

Recall that the *minimum distance* of a code C is

$$d(C) = \min_{\substack{x,y\in C\\ x\neq y}} d_H(x, y),$$

i.e. the least value of d_H over all pairs of words in C. Consequently, if a received word contains at most $r = \lfloor \frac{d(C)-1}{2} \rfloor$ errors, there will be a unique nearest neighbour in C and we can decode correctly. We call this parameter r the *correction capability* of C.

If the code C is a set of permutations of $\{1, \ldots, n\}$, then it is well known that the Hamming distance between permutations $g, h \in C \subseteq S_n$ is simply

$$d_H(g, h) = n - \left|\mathrm{Fix}(gh^{-1})\right|,$$

E-mail address: robertb@math.carleton.ca.

0012-365X/$ – see front matter
doi:10.1016/j.disc.2008.12.027

where $\mathrm{Fix}(g)$ denotes the set of fixed points of g. (By applying h^{-1} to both g and h, $d_H(g, h) = d_H(gh^{-1}, 1)$ is the number of places gh^{-1} differs from the identity, which is $n - \left|\mathrm{Fix}(gh^{-1})\right|$.) Thus the minimum distance of such a set C is equal to

$$\min_{\substack{g,h\in C\\ g\neq h}} n - \left|\mathrm{Fix}(gh^{-1})\right| = n - \max_{\substack{g,h\in C\\ g\neq h}} \left|\mathrm{Fix}(gh^{-1})\right|.$$

When a set of permutations forms a group G, this becomes

$$n - \max_{\substack{g\in G\\ g\neq 1}} |\mathrm{Fix}(g)|,$$

which is known to group theorists as the *minimum degree* of G. There is an analogy here to the theory of linear codes, in that the minimum distance of a linear code is equal to its minimum weight, i.e. the distance from the all-zero codeword, which plays the role of the identity permutation in that setting.

The family of groups that appear in Blake's original paper [6] are the *sharply k-transitive* groups. A group G acting on a set Ω is sharply k-transitive if for any two ordered k-tuples of distinct elements of Ω, there is a unique group element mapping the first to the second. These were an obvious starting point for this theory, as the minimum distance of such a group is easy to calculate. As Blake observed in [6], the minimum distance is simply $n - k + 1$ (only the identity element can fix k points, so the maximum number of fixed points of a non-identity element is $k - 1$).

The symmetric group S_k is both sharply k-transitive and sharply $(k - 1)$-transitive, while the alternating group A_k is sharply $(k - 2)$-transitive. Thus the minimum distance of S_n is $n - (n - 1) + 1 = 2$, and the minimum distance of A_n is $n - (n - 2) + 1 = 3$. Consequently, the correction capability of the symmetric group is 0, and that of the alternating group is 1. So S_n is of no use as an error-correcting code, but A_n is a 1-error-correcting code. For $k > 5$, there are no others, but there are infinite families for $k = 2$ and 3, as well as the Mathieu groups M_{11} (for $k = 4$) and M_{12} (for $k = 5$): Bray and the author [3] consider M_{12} viewed as a code in detail. The method we present in the next section is completely general.

2. A decoding algorithm: Uncoverings-by-bases

In order to use a permutation group as an error-correcting code, it is necessary to have a suitable decoding algorithm. To this end, the following definition, originally due to Sims [19], is fundamental.

Definition 1. Let G be a group acting on a finite set Ω. A *base* for G in this action is a sequence of points $(x_1, \ldots, x_b)$ from Ω such that $G_{(x_1,\ldots,x_b)} = \langle 1\rangle$, i.e. the pointwise stabiliser is the identity. An *irredundant* base is a base where $G_{(x_1,\ldots,x_i,x_{i+1})} \neq G_{(x_1,\ldots,x_i)}$ for $i = 1, \ldots, b - 1$.

Example 2. Suppose G is sharply k-transitive. Then any sequence of k points forms a base, as the stabiliser of any k points is trivial.

Example 3. Suppose G is the general linear group $\mathrm{GL}(n, q)$ acting on $\mathbb{F}_q^n \setminus \{0\}$. Then any basis for the vector space $\mathbb{F}_q^n$ is a base for G.

Bases have the following property.

Proposition 4. *For any group G, the action of an element $g \in G$ on a base $(x_1, \ldots, x_b)$ uniquely determines that element; that is, if $(x_1, \ldots, x_b)^g = (x_1, \ldots, x_b)^h$, then $g = h$.*

Proof. Suppose $g, h \in G$, $(x_1, \ldots, x_b)$ is a base for G and that $x_i^g = x_i^h$ for each i. Then $x_i^{gh^{-1}} = x_i$ for each i, that is $gh^{-1} \in G_{(x_1,\ldots,x_b)} = \langle 1\rangle$. Hence $gh^{-1} = 1$, i.e. $g = h$. □

This is not only a theoretical result: there exist algorithms in computational group theory which will actually compute $g \in G$ from the image of a base; see Butler [9], Chapter 10, for details. So, if a group G is to be used as a code, if the received word contains errors in positions outside those labelled by a base, we can decode successfully. However, as it is possible for r errors to lie in any r positions, we need the following.

Definition 5. Suppose G is a group acting on Ω, where $|\Omega| = n$, with correction capability r. Then an *uncovering-by-bases* for G is a set of bases for G such that any r-subset of Ω is disjoint from at least one base.

In the case where G is sharply k-transitive, this reduces to a set of k-subsets of Ω, which we call an (n, k, r)-*uncovering*. This is equivalent to finding a set of $(n - k)$-subsets of Ω, with the property that any r-subset is *contained* in at least one $(n - k)$-set, which is precisely the definition of an $(n, n - k, r)$ *covering design*. Consequently, the extensive literature on covering designs is of use to us in finding uncoverings. For instance, there is a large internet database of covering designs with small parameters maintained by Gordon, the "La Jolla Covering Repository" [13]. Many of the constructions featured in the database are described in the paper of Gordon, Kuperberg and Patashnik [14], while a more general survey can be found in Mills and Mullin [18].

Example 6. For the sharply 3-transitive group PGL(2, 7), we have $n = 8$, $k = 3$, $r = \lfloor \frac{8-3}{2} \rfloor = 2$, so we need an (8, 3, 2)-uncovering, as shown below.

1 2 3
4 5 6
2 3 7
1 7 8

The above example was obtained from an (8, 5, 2) covering design in Gordon's database [13]. It is small enough for the "uncovering" property to be verified easily by hand.

While it is trivial that (n, k, r)-uncoverings always exist (by taking the set of all k-subsets), it is not immediately obvious that for an arbitrary group in a given action an uncovering-by-bases should exist. However, we are saved by the next result.

Proposition 7. *For any finite group G acting on a set Ω with $|\Omega| = n$, there always exists an uncovering-by-bases.*

Proof. Let d be the minimum distance of G in this action, so $r = \lfloor \frac{d-1}{2} \rfloor$. We show that for an arbitrary r-subset of Ω, there exists a base for G disjoint from it, arguing by contradiction.

Suppose there exists an r-subset $R \subseteq \Omega$ that meets every base for G. Then the pointwise stabiliser of $\bar{R} = \Omega \setminus R$ is non-trivial, as $\bar{R}$ does not contain a base. Therefore there exists a non-identity element g that fixes $\bar{R}$ pointwise, so $|\mathrm{Fix}(g)| \geq |\bar{R}| = n - r$. But the maximum number of fixed points of a non-identity element is $n - d < n - r$, giving a contradiction. □

We remark that the definition of uncovering-by-bases, and indeed the proof of Proposition 7, is vacuous in the case $r = 0$. Although groups with zero correction capability are, of course, useless as error-correcting codes, an uncovering-by-bases for such a group consists of a single base only.

Once an uncovering-by-bases for a given group in a given action has been obtained, we can then use it with the following decoding algorithm.

Algorithm 1. Suppose we have a permutation group G and an associated uncovering-by-bases $\mathcal{U} = \{B_1, \ldots, B_u\}$, and that we have transmitted the permutation $g = g_1 g_2 \cdots g_n \in G$ and received the word $w = w_1 w_2 \cdots w_n$, which is assumed to have at most r errors. Set $i := 1$.

The iterative step is as follows. Take B_i, and look at the entries w_j for $j \in B_i$. If there are no repeated symbols in those positions, then we can determine if there is an element $g' \in G$ (i.e. a codeword) agreeing with w in those positions. If not, then set $i := i + 1$ and repeat. If g' does exist, we compute $d_H(w, g')$; if this is at most the correction capability r, we must have that $g' = g$ and return g'. If the distance is more than r, we set $i := i + 1$ and repeat.

We make some remarks about this algorithm. First, the fact that $\mathcal{U}$ is an uncovering-by-bases guarantees that the algorithm will succeed. Next, we know that we can ignore cases where there are repeated symbols, as we know there must be an error among them. The method for finding g' is described in Algorithm 2 below; if G is sharply k-transitive, the element g' is guaranteed to exist, so this step will always succeed. Finally, we remark that the algorithm can be implemented in the computer system GAP [12] (see [1] for details).

Example 8. We continue with the example of PGL(2, 7), which is generated by the permutations (3 8 7 6 5 4) and (1 2 6)(3 4 8) (in disjoint cycle form). As a code, the word length is 8, minimum distance 6, correction capability 2, and there are 336 codewords. Suppose we transmit the permutation

$g =$ 1 2 3 4 5 6 7 8

and that the following word is received:

$w =$ 4 2 3 6 5 6 7 8.

(This has two errors, in positions 1 and 4.) Using the uncovering in Example 6, we first identify the element of PGL(2, 7) which maps the base (1, 2, 3) to (4, 2, 3), which is

4 2 3 6 8 7 5 1.

This is distance 4 from w, so is rejected. Then we look at the next base in the uncovering, which is (4, 5, 6). In w, these positions contain the symbols (6, 5, 6), so clearly no permutation can exist here. So we move to the next 3-tuple, (2, 3, 7); these positions contain the symbols (2, 3, 7). Thus we find the element which maps the first to the second, which is

1 2 3 4 5 6 7 8.

This is distance 2 from w, so is accepted, and the algorithm terminates.

In the author's paper [2], uncoverings-by-bases are constructed for certain families of *base-transitive* groups. These are groups which act transitively on their irredundant bases, so the fact that all irredundant bases have the same structure simplifies the construction of an uncovering-by-bases. Some examples of groups which are not base-transitive are considered in Sections 7 and 8 below. In another paper [5], Prellberg and the author describe an alternative (and faster) algorithm which works for the family of groups $C_m \wr S_n$ in their imprimitive action on mn points.

3. Parallels with linear codes

In this section we mention some of the analogies between linear codes (i.e. vector spaces over finite fields) and permutation groups when viewed as codes. To begin with, we compare their rates.

The *rate* of an error-correcting code is defined as $\frac{1}{n}\log_q M$, where n is the word length, q the alphabet size, and M the number of codewords. For a linear code of length n and dimension k over $\mathbb{F}_q$, this works out simply as $\frac{k}{n}$. For a permutation group of degree n, we have $\frac{1}{n}\log_n |G|$. Now, if G has an irredundant base of length k, it is known that $2^k \leq |G| \leq n^k$ (see [10], Section 4.13); the closest to reaching the upper bound are the sharply k-transitive groups of order $n(n-1)\cdots(n-k+1)$. Consequently, an upper bound on the rate of $\frac{k}{n}$ is obtained. Since the base size is the permutation group analogue of the dimension of a vector space, this is not a coincidence.

It is thought that for primitive permutation groups, the order tends to be much closer to the upper bound than the lower. Consequently, for many permutation groups (should the assertion be true) the rates of such groups (viewed as codes) would be comparable with linear codes.

Next, we consider distance enumerators. For any code, we can define a polynomial which "counts" the number of codewords at each distance from a fixed codeword, which is as follows.

Definition 9. Let C be a code of fixed length, and choose $c \in C$. The *distance enumerator*, is defined to be

$$\Delta_c(x) = \sum_{w \in C} x^{d_H(c,w)}.$$

Clearly, the coefficient of x^i gives the number of codewords at distance i from c. In the case where C is a permutation group, we obtain the same polynomial regardless of the choice of c (because each codeword can be mapped to any other by a permutation), so we can take c to be the identity permutation and call the polynomial $\Delta(x)$. In the case where C is a linear code, we take c to be the zero vector and obtain the *weight enumerator*. Weight enumerators for linear codes have been studied extensively (any text on coding theory should mention them), while the distance enumerator for permutation groups is studied by Dixon and the author in [4].

Finally, we mention that our decoding algorithm is similar to the method of *permutation decoding*. This method, which is attributed to F.J. MacWilliams, is a decoding method used for linear codes, and involves finding a subset of the automorphism group of the code to move any set of errors out of the "information positions". It is also related to covering designs, with the analogue of an uncovering-by-bases being known as a *PD-set*. A full description is given in the survey article by Huffman [16] in the *Handbook of Coding Theory*.

4. Complexity issues

An important part of the decoding algorithm (Algorithm 1) is where a group element is reconstructed from a set of base images. In order to determine the complexity of that algorithm, we need to know exactly what this entails. Suppose we have a group G acting on a set Ω of size n, and suppose that $(x_1, \ldots, x_b)$ is a base for G in this action. Let G_i denote the pointwise stabiliser in G of $(x_1, \ldots, x_i)$, with the convention that $G_0 = G$. The following definition is due to Sims [19].

Definition 10. The set $S = S_1 \cup S_2 \cup \cdots \cup S_b$, where S_i is a set of coset representatives for G_i in G_{i-1}, is called a *strong generating set* for G.

This set is indeed a set of generators for G; furthermore, any element of G can be written uniquely as a product $s_b s_{b-1} \cdots s_1$, where each $s_i \in S_i$ (see Butler [9], Chapter 10). This enables us to use the following algorithm.

Algorithm 2. Suppose that we have a group G which we are using as an error-correcting code. Suppose that $B = (x_1, \ldots, x_b)$ is a base for G and that S is a corresponding strong generating set, and that we have a received word w which has symbols $(y_1, \ldots, y_b)$ in the positions labelled by B. Then we want an answer to the following question:

Does there exist $g \in G$ with $x_i^g = y_i$ for all i, and if yes, what is it?

At the first stage, we have that the set $\{x_1^s \mid s \in S_1\}$ is the G_0-orbit on x_1, and see if y_1 appears in it. If not, then no such g can exist, and the algorithm stops. If it does appear, we let s_1 be the element that maps x_1 to y_1, then replace $(y_1, \ldots, y_b)$ with $(y_1^{s_1^{-1}}, \ldots, y_b^{s_1^{-1}})$, and iterate as follows.

At step i, we check if there exists some $s_i \in S_i$ such that $x_i^{s_i} = y_i^{s_{i-1}^{-1}\cdots s_1^{-1}}$. If not, then the algorithm stops; if some s_i does exist, we replace $(y_1^{s_{i-1}^{-1}\cdots s_1^{-1}}, \ldots, y_b^{s_{i-1}^{-1}\cdots s_1^{-1}})$ with $(y_1^{s_i^{-1}\cdots s_1^{-1}}, \ldots, y_b^{s_i^{-1}\cdots s_1^{-1}})$, then repeat the iteration.

When we reach step b, if we succeed we take the element $s_b s_{b-1} \cdots s_1$ to be our required element g. This works because for each i we have

$$\begin{aligned} x_i^g &= x_i^{s_b \cdots s_1} \\ &= x_i^{s_i \cdots s_1} \quad (\text{since } s_b, s_{b-1}, \ldots, s_{i+1} \text{ all lie in } G_i) \\ &= (y_i^{s_1^{-1}\cdots s_{i-1}^{-1}})^{s_{i-1}\cdots s_1} \\ &= y_i. \end{aligned}$$

Calculating the element $g = s_b s_{b-1} \cdots s_1$ concludes the algorithm.

Now that we know what the procedure is, we are able to answer the questions, "how long does it take?" and "how much space is required?". The latter question has two parts, as there are two kinds of space needed: storage space for any look-up tables (ROM), and space needed for performing the actual computation (RAM). For the sake of simplicity, we make the following assumptions:

- finding the image of a point under a permutation takes one unit of time;
- the composition of two permutations of length n takes n units of time;
- the storage of a single symbol requires one unit of space.

We also use the convention where $g(n) = \mathrm{O}(f(n))$ means that $g(n)$ is bounded above by some constant multiple of $f(n)$.

Lemma 11. *The time required by the element reconstruction algorithm (Algorithm 2) is* $\mathrm{O}(bn)$.

Proof. At step i we look through $\{x_i^{s_i} \mid s_i \in S_i\}$ to see if $y_i^{s_1^{-1}\cdots s_{i-1}^{-1}}$ appears there. Since $|S_i| = |G_{i-1} : G_i| \leq n$, there are at most n operations to be made here. If we succeed here, we replace b symbols with their images under s_i^{-1}, and as acting on a point by a permutation requires one operation, this gives b operations, so there are at most $n + b$ operations per step. As there are at most b steps, this gives a maximum of $b(n + b)$.

If all b steps are completed successfully, we then have to compose b permutations, which requires $(b - 1)n$ operations. Overall, the maximum number of operations required will be $b(n + b) + (b - 1)n$, so this is $\mathrm{O}(bn)$. □

Lemma 12. *The storage space required by the element reconstruction algorithm (Algorithm 2) is* $\mathrm{O}(bn^2)$*, and the space required to perform the algorithm is* $\mathrm{O}(n)$.

Proof. With the convention that a single symbol requires one unit of space, a permutation needs n units. Our look-up table comprises a strong generating set and the corresponding set of inverses. Now, a strong generating set has size bounded by bn, as each S_i has size at most n. As we also need to store the inverse of each element, the overall number of storage units required is at most $2bn^2$, which is $\mathrm{O}(bn^2)$.

When performing the algorithm, at each stage i we need to store the position in the look-up table of the element s_i, which requires one unit of space, giving us a total of b units. Also, when performing the composition of permutations at the end, we need a further n units for this. So we have a total of $n + b$ units, which is $\mathrm{O}(n)$. □

Recall that our decoding algorithm (Algorithm 1) works by working through a set of bases and applying the element reconstruction algorithm repeatedly until the correct permutation is obtained. Let $\mathcal{U}$ be the uncovering-by-bases being used. Then we have the following results.

Theorem 13. *The time required by the decoding algorithm (Algorithm 1) is* $|\mathcal{U}|\mathrm{O}(bn)$.

Proof. We apply the element reconstruction algorithm (Algorithm 2), which by Lemma 11 needs at most $b(n+b)+bn$ time units. After this, we check the Hamming distance between the reconstructed permutation and the received word to see if it is within the correction capability, so there are n checks here. We then may have to repeat this procedure until it has been carried out $|\mathcal{U}|$ times, so the total number of steps is bounded by $|\mathcal{U}|(b(n + b) + bn + n)$, which is $|\mathcal{U}|\mathrm{O}(bn)$. □

Theorem 14. *The storage space required by the decoding algorithm (Algorithm 1) is* $|\mathcal{U}|\mathrm{O}(bn^2)$*, and the space required to perform the algorithm is* $\mathrm{O}(n)$.

Group	Degree	Correction capability	Size of UBB
A_n	n	1	$\lceil \frac{n}{3} \rceil$
$H \wr S_n$ (H regular, $\|H\| = m$)	mn	$\lfloor \frac{m-1}{2} \rfloor$	$\lfloor \frac{m-1}{2} \rfloor + 1$
Base-transitive, rank 2	$n = km$	$\lfloor \frac{(k-1)m-1}{2} \rfloor$	$\lfloor \frac{(k-1)m-1}{2} \rfloor + 1$
Sharply 3-transitive	n	$\lfloor \frac{n-3}{2} \rfloor$	n or $n-3$
GL(3, q)	q^3-1	$\frac{q^3-q^2}{2} - 1$	q^3-1 or q^3-4
AGL(2, q)	q^2	$\frac{q^2-q}{2} - 1$	q^2 or q^2-3
A_7 on 15 points	15	5	9
M_{11}	11	3	8
M_{12}	12	3	11
S_m on 2-subsets (Section 7)	$\binom{m}{2}$	$m-3$	Between $\frac{3}{2}(m-2)$ and $2(m-1)$
"Dihedral-like" (Section 8)	n	$\lfloor \frac{n-2}{2} \rfloor$	n or $\frac{n}{2}$

Fig. 1. Parameters of various groups when viewed as codes.

Proof. For the look-up table, each base in $\mathcal{U}$ requires its own strong generating set, so by Lemma 12, we will need $2|\mathcal{U}|bn^2$ storage units, which is $|\mathcal{U}|O(bn^2)$. To perform the algorithm, whilst applying the reconstruction algorithm we need $b + n$ units of space. The same n units are then used for the comparison with the received word, meaning that the space required here is still $O(n)$. □

So, ultimately, both the time complexity and the amount of storage space required are dependent on the size of $\mathcal{U}$. In Section 9, we investigate some possible bounds for this.

5. The single-orbit conjecture

We have already seen (in Proposition 7) that, for any permutation group G of degree n and minimum distance (i.e. minimum degree) d and correction capability r, there exists an uncovering-by-bases for G. In [2], the author constructs uncoverings-by-bases for certain base-transitive groups, so in that situation it follows that an uncovering-by-bases contains irredundant bases from one orbit only. It is this particular property that interests us in this section.

Definition 15. We say that a permutation group G has the *single-orbit property* if there exists an orbit on irredundant bases for G that contains an uncovering-by-bases.

Furthermore, we make the following conjecture.

Conjecture 16 (*The Single-orbit Conjecture*). *Any permutation group has the single-orbit property.*

While we do not offer a proof of this conjecture, we have various pieces of evidence that it should be true, such as the following.

- It holds trivially for base-transitive groups (as there is only one orbit) and for groups with $r = 0$ (as we only need one base).
- The single-orbit property is preserved by taking direct and wreath products: see Section 6.
- The conjecture holds for the action of S_m on 2-subsets (Section 7) and for some further examples of groups (Section 8).
- Computer searches (see below) show that the conjecture holds for transitive groups of degree at most 19, and primitive groups of degree at most 30.

Using the GAP libraries of transitive and primitive groups, it is relatively easy to verify the conjecture for groups of low degree. For each group, one has to determine the correction capability, then for a given base construct the orbit of the group on that base, and then check that this orbit forms an uncovering. GAP programs were used to test transitive groups of degree at most 19 and primitive groups of degrees 20 to 30 for the single-orbit property, and did not find any counterexample.

In practice, it is important to know not just that an uncovering-by-bases exists, but also what size it should be. Thus we make the following, stronger, conjecture.

Conjecture 17. *Let G be a permutation group of degree n. The G has the single-orbit property, and furthermore, there exists such an uncovering-by-bases with size polynomial in n.*

This conjecture is likely to be much harder to prove than the single-orbit conjecture. However, the evidence we have suggests that it should be true. Consider Fig. 1 below, and in particular compare the degrees with the sizes of the uncoverings-by-bases (UBBs). (With the exception of the last two examples, constructions for the others appear in [2].)

In each case, we observe that the size of the uncovering-by-bases is bounded above by the degree of the group. This leads us to the next conjecture, possibly the most optimistic so far.

Conjecture 18. *Let G be a permutation group of degree n. The G has the single-orbit property, and furthermore, the uncovering-by-bases obtained has size bounded above by n.*

A computer search (using GAP) has demonstrated that Conjecture 18 holds for all transitive groups of degree up to 15. This computer search involves, for each group, randomly constructing uncoverings-by-bases until one of size less than the degree is found. (For most groups, only one attempt was necessary.)

The various versions of the single-orbit conjecture are not entirely unmotivated; they have implications for the time and space complexity of the decoding algorithm, as discussed in Section 9 below.

6. The single-orbit property for direct and wreath products

In this section, we show that the single-orbit property is preserved by taking direct products (in the intransitive action) and wreath products (in the imprimitive action).

Theorem 19. *Suppose that G acting on Ω and H acting on Δ have the single-orbit property. Then $G \times H$ acting on $\Omega \dot\cup \Delta$ also has this property.*

Proof. We take suitable orbits for G and H, then use them to form an orbit for $G \times H$, and show that this satisfies our requirements. Suppose G acting on Ω has degree n, minimum distance d and correction capability $r = \lfloor \frac{d-1}{2} \rfloor$, and that $\mathbf{x} = (x_1, \ldots, x_k)$ is a base for G in this action, such that the orbit $\mathbf{x}^G$ forms an uncovering-by-bases. Suppose also that H acting on Δ has degree m, minimum distance e and correction capability s, and that $\mathbf{y} = (y_1, \ldots, y_l)$ is a base such that $\mathbf{y}^H$ forms an uncovering-by-bases.

Now consider the action of $G \times H$ acting on $\Omega \dot\cup \Delta$. Clearly $|\Omega \dot\cup \Delta| = n + m$, and the minimum distance of $G \times H$ is $\min\{d, e\}$. Consequently, the correction capability of $G \times H$ is $\min\{r, s\}$.

To construct a base for $G \times H$, we define $\mathbf{z} = (x_1, \ldots, x_k, y_1, \ldots, y_l)$. Since G acts on Ω only and H acts on Δ only, we have

$$\mathbf{z}^{(g,h)} = (x_1^{(g,h)}, \ldots, x_k^{(g,h)}, y_1^{(g,h)}, \ldots, y_l^{(g,h)}) = (x_1^g, \ldots, x_k^g, y_1^h, \ldots, y_l^h)$$

for any $(g, h) \in G \times H$. In particular, if $\mathbf{z}^{(g,h)} = \mathbf{z}$ (i.e. if $(g, h) \in \mathrm{Stab}_{G\times H}(\mathbf{z})$), we have that $\mathbf{x}^g = \mathbf{x}$ and $\mathbf{y}^h = \mathbf{y}$, so because $\mathbf{x}$ and $\mathbf{y}$ are bases we have $g = 1_G$ and $h = 1_H$, so $(g, h) = 1_{G\times H}$. Hence the pointwise stabiliser of $\mathbf{z}$ is trivial, so $\mathbf{z}$ forms a base for $G \times H$.

Now, we have that $\mathbf{z}^{G\times H} = \{(x_1^g, \ldots, x_k^g, y_1^h, \ldots, y_l^h) \mid g \in G, h \in H\}$. We show that $\mathbf{z}^{G\times H}$ forms an uncovering-by-bases for $G \times H$. Suppose without loss of generality that $r \leq s$. We need to show that given an arbitrary r-subset $R \subset \Omega \dot\cup \Delta$, there exists an element of the orbit $\mathbf{z}^{G\times H}$ that is disjoint from R. Suppose that $R = A \dot\cup B$, where $A \subset \Omega$, $B \subset \Delta$ and $|A| + |B| = r$ (note that one of A, B may be empty). Since $|A| \leq r$, there exists a base $(x_1^g, \ldots, x_k^g)$ for G disjoint from A. Similarly, since $|B| \leq r \leq s$, there exists a base $(y_1^h, \ldots, y_l^h)$ for H disjoint from B. Thus the base $(x_1^g, \ldots, x_k^g, y_1^h, \ldots, y_l^h) \in \mathbf{z}^{G\times H}$ is disjoint from $R = A \dot\cup B$, so we are done. □

Theorem 20. *Suppose G acting on Ω has the single-orbit property, and that H is an arbitrary permutation group of degree m. Then $G \wr H$ acting on m copies of Ω also has the single-orbit property.*

Proof. As in Theorem 19 above, we take a suitable orbit for G and use it to find a suitable orbit for $G \wr H$. Suppose G acting on Ω has degree n, minimum distance d and correction capability r. Recall that $G \wr H$ has the form $G^m \rtimes H$ (often written as $G^m : H$), where G^m is the direct product of m copies of G.

Now, a non-identity element of $G \wr H$ with the maximum number of fixed points will be an element of G^m, as any non-trivial action of H will reduce the number of fixed points. But a non-identity element of G^m with the maximum number of fixed points will fix $m - 1$ copies of Ω, and have $n - d$ fixed points in the remaining copy. Hence it has $(n - 1)m + (n - d)$ fixed points in total, and so the minimum distance of $G \wr H$ is d. Consequently the correction capability of $G \wr H$ is r.

Next, we need a base for $G \wr H$. Suppose $\mathbf{x}$ is an irredundant base for G such that $\mathbf{x}^G$ forms an uncovering-by-bases. Now, by the proof of Theorem 19 above (and a straightforward induction), m copies of $\mathbf{x}$ (one from each copy of Ω) forms a base for G^m. We denote this base by $m\mathbf{x}$. Clearly $m\mathbf{x}$ is also a base for $G \wr H$, as the only elements of $G \wr H$ that fix each block blockwise are elements of G^m.

Finally, we observe that the orbit $(m\mathbf{x})^{G^m}$ is contained in the orbit $(m\mathbf{x})^{G\wr H}$. By Theorem 19 above, $(m\mathbf{x})^{G^m}$ forms an uncovering-by-bases, so therefore $(m\mathbf{x})^{G\wr H}$ also does. □

7. The single-orbit property for S_m acting on 2-subsets

Consider the symmetric group S_m acting on the 2-subsets of $\{1, \ldots, m\}$. In this section we demonstrate that the single-orbit conjecture holds for this group in this action. Throughout, we assume $m \geq 4$, as the cases $m = 1$ and 2 are meaningless, and for $m = 3$ we have the usual action of S_3. In this action the group is acting on a set of size $\binom{m}{2}$, so as a code the codewords have length $\binom{m}{2}$, and clearly there are $m!$ codewords. The minimum distance and correction capability are as follows.

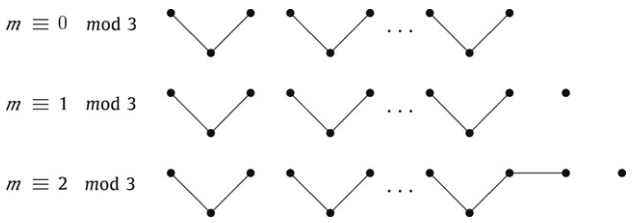


Fig. 2. Minimal bases for S_m acting on the edges of K_m.

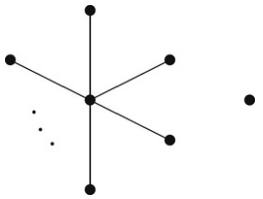


Fig. 3. An irredundant base of a different size.

Proposition 21. *The minimum distance of S_m acting on 2-subsets is $2(m-2)$ and the correction capability is $r = m-3$.*

Proof. The maximum number of fixed points is $\binom{m-2}{2}+1$, corresponding, for example, to a transposition $(1\ 2)$, which will fix the remaining 2-sets chosen from $\{3,\ldots,m\}$ and also the pair $\{1,2\}$. Thus the minimum distance is

$$\binom{m}{2} - \binom{m-2}{2} - 1 = 2(m-2).$$

Consequently the correction capability is $r = \lfloor \frac{2(m-2)-1}{2} \rfloor = m-3$. □

The 2-subsets of $\{1,\ldots,m\}$ can be thought of as the edge set of the complete graph K_m. Thinking in this way enables us to use graph-theoretic methods, which we shall exploit to construct uncoverings-by-bases. For instance, a base for S_m in this action will consist of a subset of these edges.

Lemma 22. *A base for S_m acting on the edge set of K_m consists of the edges of a spanning subgraph of K_m which has* (i) *at most one isolated vertex and* (ii) *no isolated edges.*

Proof. Let Γ denote such a spanning subgraph. To show that Γ is base, we have to show that the edgewise stabiliser, $G_{(E\Gamma)}$, of Γ in $G = S_m$ is trivial. First we suppose that Γ contains no isolated vertex. Let $e_1 = \{i,j\}$ be an edge in Γ. Since e_1 is not isolated, there exists another edge e_2 that is incident with e_1. We can suppose without loss of generality that $e_2 = \{j,k\}$. Now choose some $g \in G_{(E\Gamma)}$, so g fixes e_1 and e_2, i.e.

$$\{i,j\}^g = \{i,j\} \quad \text{and} \quad \{j,k\}^g = \{j,k\}.$$

The only way this can happen is if g fixes the vertex j, which then forces g to fix both i and k as well. But since Γ is a spanning subgraph, and there are no isolated vertices, every vertex must lie in such a configuration, so must be fixed by g. Hence all m vertices are fixed by g, and so $g = 1$.

If there *is* a single isolated vertex, we have by the same argument as above that the remaining $m-1$ vertices are fixed, which forces the remaining vertex to be fixed. So in this case we also have $g = 1$.

The converse is easy: clearly a graph violating condition (i) or (ii) cannot be base, as the edgewise stabiliser would not trivial. □

A minimal base for S_m in this action obtained from the bases described above will be such a graph with the least number of edges. Thus a graph of one of the three forms shown in Fig. 2 will form a minimal base for this action (according to congruence classes modulo 3).

We call a base of this form a *V-graph*. Clearly such a base is irredundant, as removing any edge will leave a graph with an isolated edge or two isolated vertices, which will have non-trivial edge stabiliser. We note that S_m acts transitively on V-graphs, so they form a single orbit. There are, however, other graphs which form irredundant bases, such as Fig. 3. The V-graphs are irredundant bases of size $\sim \frac{2}{3}m$, but the "star" bases are irredundant bases of size $m-2$.

Recall that a *Hamilton circuit* in a graph Γ is a circuit in Γ containing each vertex exactly once, and that Γ is said to be *Hamiltonian* if it contains such a circuit. Now, we observe that any V-graph is contained inside a Hamilton circuit of K_m, as is shown in Fig. 4 (for $m \equiv 0 \mod 3$). From a V-graph, by adding extra edges (shown as ●═══●) we can obtain a Hamilton circuit; conversely, if we have a Hamilton cycle we can remove those edges to obtain a V-graph.

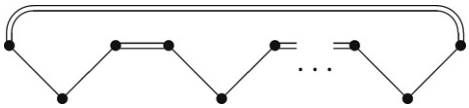


Fig. 4. A V-graph inside a Hamilton circuit.

In order to prove that S_m in this action has the single-orbit property (where the orbit on bases is the set of V-graphs), we show that for an arbitrary r-subset of edges, there exists a V-graph disjoint from it. To do this, we need the following result on Hamiltonicity. (The notation $\deg(v)$ denotes the degree of the vertex v.)

Theorem 23 (*Ore's Theorem, 1960*). *If Γ is a simple graph with $m \geq 3$ vertices, and where $\deg(v) + \deg(w) \geq m$ for all pairs of non-adjacent vertices v, w, then Γ is Hamiltonian.*

Proof. See Wilson [20], Theorem 7.1. □

We apply this in the following theorem.

Theorem 24. *There exists an uncovering-by-bases for the action of S_m on 2-subsets, contained in a single-orbit on irredundant bases.*

Proof. Let R denote an arbitrary r-set of edges of K_m. Choose two vertices v, w which are non-adjacent in $\Gamma = K_m \setminus R$, so therefore the edge $e = \{v, w\} \in R$. Suppose s further edges of R are incident with v and t further edges are incident with w, so we have $0 \leq s+t \leq r-1$. In K_m, both v and w have degree $m-1$, so in Γ, we have $\deg(v) = (m-1)-(s+1) = m-s-2$ and $\deg(w) = m-t-2$. Consequently, we have

$$\begin{aligned}\deg(v) + \deg(w) &= (m-s-2) + (m-t-2)\\ &\geq 2(m-2) - (r-1)\\ &= 2(m-2) - (m-4)\\ &= m,\end{aligned}$$

so by Ore's Theorem (Theorem 23), Γ is Hamiltonian. Therefore Γ contains a V-graph (so K_m contains a V-graph disjoint from R), and we are done. □

Of course, this is only an existence proof: it does not give us a way of constructing an uncovering-by-bases, or give us an idea of its size. We resolve this problem below.

7.1. A construction of uncoverings-by-bases

We continue with our graph-theoretic approach from above. A *decomposition* of a graph Γ is a partition of the edge-set of Γ; a *Hamiltonian decomposition* of Γ is a decomposition into disjoint Hamilton circuits. In the 1890s, Walecki showed that when m is odd, K_m has a Hamiltonian decomposition. Clearly if m is even, this cannot happen; however, in this case K_m can be decomposed into Hamilton circuits and a 1-factor (i.e. perfect matching). This is also due to Walecki: see Lucas [17] or the survey by Bryant [8] for details. We will use Walecki's decompositions to construct uncoverings-by-bases for the action of S_m on the edges of K_m, as follows.

Construction 25. For m odd, take a Hamiltonian decomposition of K_m. Then in each Hamilton circuit, take the set of all V-graphs contained in it. For m even, we take the Hamilton circuits in the decomposition described above, and for each of those take the set of all V-graphs contained in that.

In order to prove that these constructions do indeed yield uncoverings-by-bases, we need to show that for an arbitrary set R of r edges, there is a V-graph disjoint from R which is contained in one of the specified Hamilton circuits. First we need two lemmata.

Lemma 26. *Let π be a partition of the integer n into k parts. Then if $n < 2k$, π contains a part of size 1.*

Proof. Suppose not, i.e. suppose that π has k parts, each of which has size at least 2. Then clearly $n \geq 2k$. □

Lemma 27. *Suppose that R contains a single edge e within a Hamilton circuit C. Then there is a V-graph contained in C which avoids e.*

Proof. Recall from Fig. 4 how a V-graph, say B, is obtained from a Hamilton circuit. Then by choosing B such that e is one of the edges in $C \setminus B$, we have that B avoids e. □

We can now proceed with the proof. Not surprisingly, we consider the cases m odd and m even separately.

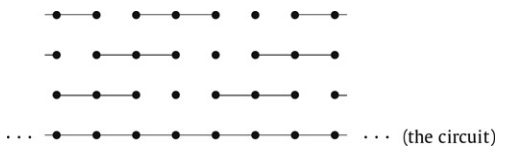


Fig. 5. Arranging V-graphs for $m \equiv 1 \bmod 3$.

Theorem 28. *Let $\mathcal{H}$ denote a Hamiltonian decomposition of K_m, for m odd. Let $\mathcal{U}$ denote the set of V-graphs contained in the Hamilton circuits of $\mathcal{H}$. Then $\mathcal{U}$ avoids any r-edge subset of the edges of K_m.*

Proof. Recall that $r = m - 3$ from Proposition 21. Also note that $\mathcal{H}$ contains $c = \frac{1}{2}(m-1)$ Hamilton circuits.

Let R denote an arbitrary set of r edges of K_m. This may contain edges from many Hamilton circuits. However, if it meets strictly less than c of them, we are done, as we can choose a V-graph from one of the remaining circuits. So we assume that R meets every circuit in $\mathcal{H}$.

If this is so, then there exists a circuit which contains only one edge of R for the following reason. We have r edges, partitioned into c parts (one for each circuit). Now, we have $r = m - 3$ and $c = \frac{1}{2}(m-1)$, so therefore $r = 2c - 2$. By Lemma 26, a partition of $2c - 2$ into c parts must have a part of size 1, so there must be a circuit containing just one edge of R. Let C denote such a circuit, containing a single edge $e \in R$. Then by Lemma 27, there exists a V-graph contained in C which avoids e and therefore avoids the rest of R. □

The case where m is even works similarly.

Theorem 29. *Let $\mathcal{F}$ denote a decomposition of K_m, m even, into Hamilton circuits and a 1-factor. Let $\mathcal{U}$ denote the set of V-graphs formed from the Hamilton circuits in $\mathcal{F}$. Then $\mathcal{U}$ avoids any r-edge subset of the edges of K_m.*

Proof. Again, we recall that $r = m-3$. We regard the parts of the decomposition $\mathcal{F}$ as *colour classes*; we have $c = \frac{1}{2}(m-2)$ *Hamiltonian* colour classes (corresponding to the Hamilton circuits) and a single distinguished colour class (corresponding to the 1-factor), which we label as "black".

Let R denote an arbitrary set of r edges of K_m. If it contains edges from strictly less than c of the Hamiltonian colour classes, then we are done, as we can choose a V-graph from inside the remaining classes. So we assume that R meets every Hamiltonian colour class.

Suppose that this happens, and that R also contains $b \geq 0$ "black" edges. Ignoring these b edges, we have a partition of $r-b$ edges into c colour classes. Now, since $r = m-3$ and $c = \frac{1}{2}(m-2)$, we have that $r = 2c-1$, so $r-b = 2c-b-1 < 2c$. By Lemma 26, this partition must contain a part of size 1. Hence there is a Hamiltonian colour class (i.e. Hamilton circuit) which contains a single edge $e \in R$. Then by Lemma 27, there exists a V-graph contained in C which avoids e and therefore avoids the rest of R. □

In fact, in the cases where $m \not\equiv 0 \bmod 3$, it is possible to refine our construction in order to reduce the size: it is not actually necessary to take *all* V-graphs from inside each Hamilton circuit. Recall from Lemma 27 that for each edge e in a given Hamilton circuit C, we need to provide a V-graph contained in C that is disjoint from e. The next result tells us how many V-graphs we need inside each circuit.

Lemma 30. *Let $m \geq 6$. Then the number of V-graphs inside a Hamilton circuit of length m needed to avoid any single edge is 3 when $m \equiv 0 \bmod 3$ or $m \equiv 1 \bmod 3$, or 4 when $m \equiv 2 \bmod 3$.*

Proof. When $m \equiv 0 \bmod 3$, a V-graph has every third edge of the circuit removed, so that circuit only contains three V-graphs. For $m \equiv 1 \bmod 3$ (with $m \geq 7$), again we have a circuit with every third edge removed, except at one point where we remove two edges and leave an isolated vertex. Now, we can write $m = 3s + 7$. Over the $3s$ edges, we arrange three V-graphs missing every third edge, so that all these edges are avoided by one of these. This leaves us with seven edges remaining, over which we arrange our three V-graphs as shown in Fig. 5. As can be seen from this, all seven edges are avoided by at least one of the three V-graphs.

Now we consider $m \equiv 2 \bmod 3$ (where $m \geq 8$). This time, the V-graphs have the repeating pattern of every third edge omitted, except at the end, where there is a path of length 3 then two missing edges. This time, we can write $m = 3s + 8$. Over $3s$ edges of the circuit, we arrange three V-graphs as before, so that all these edges are avoided. We then add a fourth V-graph which is a duplicate of one of the first three. Over the remaining eight edges, we arrange the V-graphs as shown in Fig. 6. As can be seen, each of the eight edges is avoided by at least one V-graph. □

This proof leaves us with the cases $m = 4$ and $m = 5$ outstanding, which can both be handled easily. K_4 can be decomposed into one Hamilton circuit and a 1-factor, while a V-graph consists of two adjacent edges, and we have $r = 1$. So we split the Hamilton circuit into two V-graphs, and we are done. K_5 has a Hamiltonian decomposition into two Hamilton circuits, a V-graph is a path of length three and $r = 2$. In each of the two circuits, we arrange three V-graphs as shown in Fig. 7, giving us a total of six V-graphs.

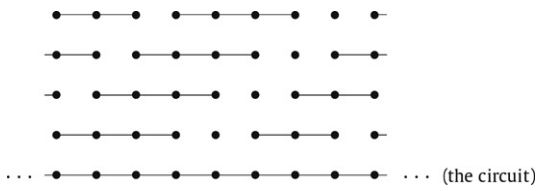


Fig. 6. Arranging V-graphs for $m \equiv 2 \bmod 3$.

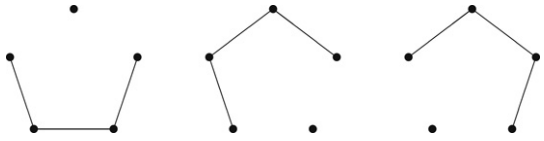


Fig. 7. Arranging V-graphs for $m = 5$.

We can now determine the sizes of these uncoverings. Observing that we are using $\frac{1}{2}(m-1)$ Hamilton circuits for m odd, and $\frac{1}{2}(m-2)$ Hamilton circuits for m even, and combining this with the result of Lemma 30 above, we obtain the following result. As the size is dependent on congruence classes modulo 3 *and* on whether m is odd or even, we can phrase this in terms of congruence classes modulo 6.

Theorem 31. *Let $m \geq 6$. Then the sizes of an uncovering-by-bases for the action of S_m on 2-subsets, as described above, are as follows:*

$$\begin{aligned}
m \equiv 0 \bmod 6 &: \quad \frac{3}{2}(m-2)\\
m \equiv 1 \bmod 6 &: \quad \frac{3}{2}(m-1)\\
m \equiv 2 \bmod 6 &: \quad 2(m-2)\\
m \equiv 3 \bmod 6 &: \quad \frac{3}{2}(m-1)\\
m \equiv 4 \bmod 6 &: \quad \frac{3}{2}(m-2)\\
m \equiv 5 \bmod 6 &: \quad 2(m-1).
\end{aligned}$$

We conclude this section with an example demonstrating our construction.

Example 32. Consider the symmetric group S_7 acting on the edges of the complete graph K_7. This graph has a Hamiltonian decomposition into three Hamilton circuits, as shown in Fig. 8(a). Applying our construction to these three Hamilton circuits, we obtain the nine V-graphs in Fig. 8(b).

8. The single-orbit property for some "dihedral-like" groups

Let G be a finite permutation group with a transitive, abelian normal subgroup A and an irredundant base of size 2. For instance, G could be a transitive subgroup of a sharply 2-transitive group, or could be a dihedral group. (For this reason we describe such groups as "dihedral-like".) Now, a transitive, abelian group is regular (see Cameron [10], Exercise 1.5), so G is acting on A and therefore has degree $|A| = n$. As A is abelian we write it additively.

Suppose $\{0, a\}$ is a base for G, where $a \in A$. We have the following theorem.

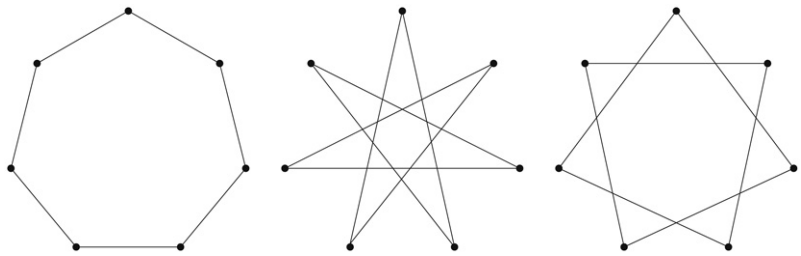
Theorem 33. *For G as above, the orbit of G on $\{0, a\}$ contains an uncovering-by-bases for G, and thus G has the single-orbit property.*

Proof. The set

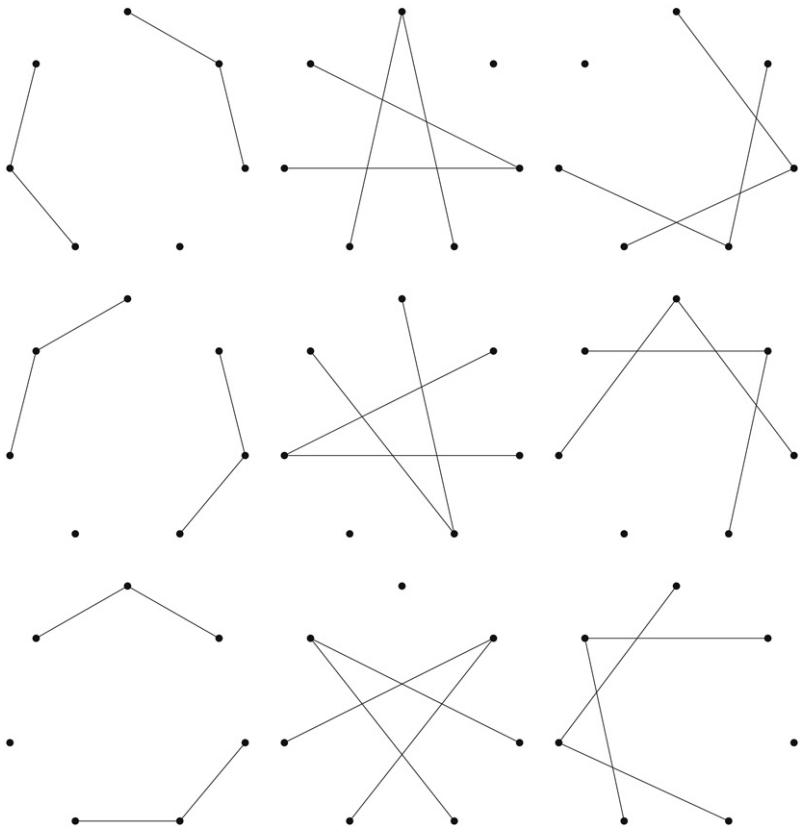
$$\mathcal{U} = \{\{x, x+a\} \mid x \in A\}$$

is the A-orbit on $\{0, a\}$, so therefore is contained in the G-orbit on $\{0, a\}$. Suppose that a has order m. We have two cases to consider.

First, suppose $m = 2$. Then $\{x, x+a\} = \{x+a, x+2a\}$, so $\mathcal{U}$ consists of $\frac{1}{2}n$ disjoint bases, which cover all n points. Since the correction capability of G is $r < \frac{1}{2}n$, we have that $\mathcal{U}$ forms an uncovering-by-bases.



(a) A Hamiltonian decomposition of K_7.



(b) An uncovering-by-bases for S_7 acting on the edges of K_7.

Fig. 8.

Second, suppose $m > 2$. Then $\mathcal{U}$ is formed of k disjoint m-cycles (where $n = km$), such as

$$\{x, x + a\}, \{x + a, x + 2a\}, \ldots, \{x + (m - 1)a, x\}.$$

So $|\mathcal{U}| = n$, and all n points are covered twice. Now choose R to be an arbitrary r-subset of the n points. These will be contained in at most $2r$ bases. But $2r < n$, so there exists a base in $\mathcal{U}$ disjoint from R, so $\mathcal{U}$ forms an uncovering-by-bases.

In both cases, $\mathcal{U}$ is contained in a single-orbit of G on irredundant bases, so G has the single-orbit property. □

In the second case of the above proof, when m is even we can take every other base in each m-cycle to form an uncovering, as this gives us $\frac{1}{2}n$ disjoint bases which is quite sufficient.

Example 34. Let G be the dihedral group of order 12, acting on six points. This contains a cyclic group of order 6, which we will regard as $(\mathbb{Z}_6, +)$. Then $\{0, 1\}$ is a base for G, and as 1 has order 6 in $\mathbb{Z}_6$, we can take every second translate of $\{0, 1\}$. Thus

$$\{0, 1\}, \{2, 3\}, \{4, 5\}$$

forms an uncovering-by-bases for G.

9. Implications for complexity

In this section we suggest some improvements which will reduce the complexity from that described in Section 4. It is here that the single-orbit conjecture from Section 5 comes into its own.

The benefit gained from having an uncovering-by-bases contained within a single orbit is that the space complexity is reduced. Instead of storing a list of all the bases and, more importantly, a list of several strong generating sets, we only need to store one base and one strong generating set, along with a list of group elements that map the first base to each of the others, and to the corresponding strong generating set. As such, we have the following.

Theorem 35. *Assuming the truth of Conjecture 16, then the storage space required by Algorithm 1 is* $O(bn^2) + |\mathcal{U}|O(n)$.

Proof. Recall from Theorem 14 that without the single-orbit property, the space required is $|\mathcal{U}|O(bn^2)$. This was because each strong generating set required $O(bn^2)$, and we had to store $|\mathcal{U}|$ of them. Now, if we assume Conjecture 16, we only need to store one, so the factor of $|\mathcal{U}|$ can be removed. However, we now need to store a list of $|\mathcal{U}|$ permutations, which map the first base to each of the others. As a permutation requires n units, and we have $|\mathcal{U}|$ permutations, this gives us $|\mathcal{U}|O(n)$. So altogether we have a space complexity of $O(bn^2) + |\mathcal{U}|O(n)$. □

This is a reduction in complexity, although by how much is dependent on the size of $|\mathcal{U}|$ when compared with $O(bn)$. However, Conjectures 17 and 18 assist with this, as they assert the existence of bounds on $|\mathcal{U}|$.

Theorem 36. *Assuming the truth of Conjecture 17, then the storage space required by Algorithm 1 is* $O(bn^2)+O(n^{k+1})$. *Assuming the truth of Conjecture 18, then the required space is* $O(bn^2)$.

Proof. Conjecture 17 asserts that $|\mathcal{U}| = O(n^k)$ for some k, while the stronger version (Conjecture 18) asserts that $|\mathcal{U}| = O(n)$. Substituting these results into Theorem 35 completes the proof. □

Note that if the size of the uncovering-by-bases is $O(n^2)$, then this gives the same space complexity as having an uncovering-by-bases of size $O(n)$.

In its weakest form, the single-orbit conjecture does not affect the time complexity. Recall that the amount of time needed to perform Algorithm 1 is bounded by $|\mathcal{U}|O(bn)$. Now, by using the modified version described above (which assumes Conjecture 17), at the final step we must compose the reconstruct group element with the 'base change' element; this is another composition of permutations, so requires another n steps. This then needs a total time of $|\mathcal{U}|(O(bn)+n) = |\mathcal{U}|O(bn)$, so there is no change. However, if the stronger conjectures are true, we have the following.

Theorem 37. *Assuming the truth of Conjecture 17, then the time required by Algorithm 1 is* $O(bn^{k+1})$. *Assuming the truth of Conjecture 18, then the required time is* $O(bn^2)$.

Proof. By the above discussion, the time required is $|\mathcal{U}|O(bn)$. Conjectures 17 and 18 give sizes for $|\mathcal{U}|$; substituting these sizes gives the required results. □

So, as far as time is concerned, the single-orbit property on its own does not affect the time complexity, but the stronger versions which include a bound on the size of the uncovering-by-bases reduces it.

Acknowledgement

The author was supported by an EPSRC CASE studentship, sponsored by the UK Government Communications Headquarters (GCHQ), while at Queen Mary, University of London.

References

[1] R.F. Bailey, Permutation groups, error-correcting codes and uncoverings, Ph.D. Thesis, University of London, 2006.
[2] R.F. Bailey, Uncoverings-by-bases for base-transitive permutation groups, Des. Codes Cryptogr. 41 (2006) 153–176.
[3] R.F. Bailey, J.N. Bray, Decoding the Mathieu group M_{12}, Adv. Math. Commun. 1 (2007) 477–487.
[4] R.F. Bailey, J.P. Dixon, Distance enumerators for permutation groups, Comm. Algebra 35 (2007) 3045–3051.
[5] R.F. Bailey, T. Prellberg, Decoding generalised hyperoctahedral groups and asymptotic analysis of correctible error patterns (in preparation).
[6] I.F. Blake, Permutation codes for discrete channels, IEEE Trans. Inform. Theory 20 (1974) 138–140.
[7] I.F. Blake, G. Cohen, M. Deza, Coding with permutations, Inf. Control 43 (1979) 1–19.
[8] D.E. Bryant, Cycle decompositions of complete graphs, in: A.J.W. Hilton, J. Talbot (Eds.), Surveys in Combinatorics 2007, in: London Mathematical Society Lecture Note Series, 346, Cambridge University Press, Cambridge, 2007.
[9] G. Butler, Fundamental Algorithms for Permutation Groups, in: Lecture Notes in Computer Science, 559, Springer-Verlag, Berlin, 1991.
[10] P.J. Cameron, Permutation Groups, in: London Mathematical Society Student Texts, 45, Cambridge University Press, Cambridge, 1999.
[11] W. Chu, C.J. Colbourn, P. Dukes, Constructions for permutation codes in powerline communications, Des. Codes Cryptogr. 32 (2004) 51–64.
[12] The GAPGroup, GAP — Groups Algorithms, and Programming Version 4.4, 2004. http://www.gap-system.org.
[13] D.M. Gordon, La Jolla Covering Repository. http://www.ccrwest.org/cover.html.
[14] D.M. Gordon, G. Kuperberg, O. Patashnik, New constructions for covering designs, J. Comb. Des. 3 (1995) 269–284.
[15] S. Huczynska, Powerline communications and the 36 officers problem, Phil. Trans. Royal Soc. A 364 (2006) 3199–3214.
[16] W.C. Huffman, Codes and groups, in: V.S. Pless, W.C. Huffman (Eds.), Handbook of Coding Theory, Elsevier, Amsterdam, 1998.
[17] E. Lucas, Récreations Mathématiqués, vol II, Paris, 1892.
[18] W.H. Mills, R.C. Mullin, Coverings and packings, in: J.H. Dinitz, D.R. Stinson (Eds.), Contemporary Design Theory: A Collection of Surveys, John Wiley & Sons, New York, 1992.
[19] C.C. Sims, Determining the conjugacy classes of a permutation group, in: G. Birkhoff, M. Hall (Eds.), Computers in Algebra and Number Theory, American Mathematical Society, Providence, 1971.
[20] R.J. Wilson, Introduction to Graph Theory, 4th ed., Prentice Hall, Harlow, 1996.