, , 1–8 ()

Enumeration, construction and random generation of block codes

HARALD FRIPERTINGER * harald.fripertinger@kfunigraz.ac.at
Institut für Mathematik
Karl-Franzens-Universität Graz
Heinrichstraße 36
A-8010 Graz

Editor:

Abstract. We discuss some methods for the enumeration, construction and random generation of isometry classes of block codes using methods from algebraic combinatorics.

Keywords: Block codes, isometry classes, enumeration, construction, random generation, discrete structures.

1. Introduction

The methods and results presented in this paper are interesting in the framework of *classification of discrete structures* [12, 13]. Very often discrete structures can be described as equivalence classes of certain objects. If these equivalence classes can be expressed as orbits under a group G acting on a set X — i. e. there is a mapping $G \times X \to X$, $(g, x) \mapsto gx$ such that $g_1(g_2x) = (g_1g_2)x$ and $1x = x$ — then there exist some combinatorial and algebraic methods for the classification of these structures. For instance each element in the discrete structure of *unlabelled graphs* can be considered as the set of all possible labellings of a given graph which is the orbit of a labelled graph under the action of the symmetric group; or the *isometry classes of block-codes* are orbits of special wreath-products as we will see below. In a first step one can enumerate these structures by applying the *Cauchy-Frobenius-Lemma*, which says that the number of G-orbits is the average number of fixed points

$$\frac{1}{|G|}\sum_{g\in G}|X_g|\,,\quad X_g := \{x \in X \mid gx = x\}\,.$$

In a second step certain properties of these structures can be described by weight functions or by their stabilizer (the stabilizer of $x \in X$ is the subgroup $G_x := \{g \in G \mid gx = x\}$ of G) and the numbers of structures with these additional properties can be computed by the *Redfield-Pólya-de Bruijn-Theory* or by *Burnsides Lemma*. The more details of a structure are specified and prescribed by parameters the closer comes the enumeration procedure to the construction of all structures

* Supported by the Fonds zur Förderung der wissenschaftlichen Forschung P10189 - PHY

with given properties. The most ambitious task of course is the computation of complete lists of representatives of a discrete structure, which can be done by carefully arranging both algebraic and combinatorial algorithms. Having computed lists of representatives we can investigate each member of a list for its properties. When the numbers of representatives get too large in order to compute complete lists, it is useful, helpful and makes sense to compute representatives uniformly at random. This way we can produce unprejudiced lists of representatives which can be used to check hypotheses on them and afterwards we can try to prove the valid ones.

Let A be a finite alphabet, then an $[n, m]$ *block code* C over A is an m-subset of A^n. In order to describe the isometry classes of block codes we need the notion of *wreath products*. The wreath product $S_A \wr S_n$ is a group formed by a set $\left\{(\psi, \pi) \;\middle|\; \psi \in {S_A}^n, \pi \in S_n\right\}$ with multiplication $(\psi, \pi)(\psi', \pi') = (\psi\psi'_\pi, \pi\pi')$, where $\psi\psi'_\pi(i) := \psi(i)\psi'_\pi(i)$ and $\psi'_\pi(i) := \psi'(\pi^{-1}i)$. (For more details on group actions and wreath products cf. [11].)

Two $[n, m]$ codes C_1 and C_2 will be called *equivalent*, if and only if there is some (ψ, π) in the full monomial group $S_A \wr S_n$ such that

$$C_1 = (\psi, \pi)(C_2) := \left\{(\psi, \pi)f \;\middle|\; f \in C_2\right\},$$

where $(\psi, \pi)f(i) = \psi(i)f(\pi^{-1}i)$. (I. e. $S_A \wr S_n$ acts in form of the *exponentiation* on A^n which induces an action on the set of all subsets of A^n.) The equivalence classes under this group action are exactly the *isometry classes* of $[n, m]$ codes.

In previous papers [4, 5, 6, 3] we were dealing with the enumeration of isometry classes of linear (n, k) codes over a finite field $GF(q)$. In this situation we had to determine the number of isometry classes of k-dimensional subspaces of $GF(q)^n$, which can be described as orbits under the action of $GF(q)^* \wr S_n$. Of course, each linear (n, k)-code over $GF(q)$ is an $[n, q^k]$ block code.

2. Enumeration of block codes

Usually when enumerating or constructing under the action in form of the exponentiation we can apply *Lehmann's Lemma* ([14, 15]) which reduces the action of a wreath product $H \wr_X G$ on Y^X to the action of the group G on the set of all functions from X into the set of all orbits of H on Y. As a matter of fact it can't be applied in the present situation since $S_A \wr S_n$ acts on the set of all m-subsets or more generally on the powerset

$$2^{(A^n)}$$

of A^n. For enumerating the isometry classes of block codes each $[n, m]$ code C can be identified with its *characteristic function*

$$\chi_C \colon A^n \to \{0, 1\}, \qquad f \mapsto \begin{cases} 1 & \text{if } f \in C \\ 0 & \text{if } f \notin C, \end{cases}$$

which fulfils $\left|f^{-1}(\{1\})\right| = m$. The other way round, each function f from A^n to $\{0, 1\}$ with $\left|f^{-1}(\{1\})\right| = m$ is the characteristic function of an $[n, m]$ block code

over A. Using *Pólya's Theorem* [18] we can determine the number of classes of block codes:

THEOREM 1 *The number of classes of $[n, m]$ block codes over the alphabet A is the coefficient of x^m in the expansion of the substitution $x_i := 1 + x^i$ into the* cycle index[1] *of the exponentiation $S_A \wr S_n$. In short it is the coefficient of x^m in*

$$Z(S_A \wr S_n, A^{n)}|_{x_i:=1+x^i}.$$

It is well known how to compute the cycle index of the exponentiation from the cycle indices of S_A and S_n. See for instance [10, 16, 17]. Using the computer algebra system SYMMETRICA [22] the following tables were computed:

Table 1. Number of classes of $[n, m]$ block codes over an alphabet of size 2.

$m\backslash n$	1	2	3	4	5	6	7	8	9
0	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
2	1	2	3	4	5	6	7	8	9
3		1	3	6	10	16	23	32	43
4		1	6	19	47	103	203	373	649
5			3	27	131	497	1606	4647	12320
6			3	50	472	3253	18435	91028	404154
7			1	56	1326	19735	221778	2074059	16.957301
8			1	74	3779	120843	2773763	51.107344	805.174011
9				56	9013	681474	33.297380	1245.930065	38921.113842
10				50	19963	3.561696	375.158732	28900.653074	1.816451.773537

Table 2. Number of classes of $[n, m]$ block codes over an alphabet of size 3.

$m\backslash n$	1	2	3	4	5	6
0	1	1	1	1	1	1
1	1	1	1	1	1	1
2	1	2	3	4	5	6
3	1	4	10	20	35	57
4		5	34	144	490	1470
5		5	105	1245	11075	82918
6		4	321	12473	334678	7.194272
7		2	846	120213	10.274578	664.545445
8		1	1984	1067757	293.142769	57778.060974
9		1	4023	8.508432	7563.157341	4.570181.600483
10			7074	60.801152	176207.637611	327.615878.641570

3. Construction of block codes

Now let me draw your attention to the construction of transversals of block codes. For doing this it is convenient to identify the alphabet A with the set $\underline{a} := \{1, \ldots, a\}$. Then the elements $f = (f(0), \ldots, f(n-1)) \in \underline{a}^n$ can be arranged in the lexicographical order $(f_1 < f_2 < \ldots < f_{a^n})$, which can be used to define a lexicographical

Table 3. Number of classes of $[n, m]$ block codes over an alphabet of size 4.

$m\backslash n$	1	2	3	4	5	6
0	1	1	1	1	1	1
1	1	1	1	1	1	1
2	1	2	3	4	5	6
3	1	4	10	20	35	57
4	1	10	55	223	759	2309
5		13	254	3227	32970	292103
6		23	1643	77194	2877651	90.647411
7		26	10164	2097080	311.400852	37593.032352
8		32	63488	57.796870	34630.385050	16.429342.163157
9		26	364843	1502.295684	3.667889.498353	6925.787777.638463
10		23	1930906	36065.804158	360.865277.628727	2.729333.815881.686935

order on the set of all characteristic functions $\chi\colon \underline{a}^n \to \{0,1\}$, by identifying each function χ with a vector $(\chi(f_1),\ldots,\chi(f_{a^n}))$. (So each code word can be identified with a 0-1 vector $(\chi(f_1),\ldots,\chi(f_{a^n}))$.) Then we may choose the lexicographically smallest element in the orbit of a block code C (given by its characteristic function) as the *canonic representative* of this orbit. In order to apply the standard algorithm of *orderly generation* combined with *Read's method of recursion* [1, 19] and *learning techniques* [7] as described in [11] we have to compute the *Sims-chain* [20] of the operating group, which can be quite time consuming using a general algorithm, since $S_{\underline{a}} \wr S_n$ is of order $n!(a!)^n$ and of degree a^n. In the next paragraph we will see how to compute this Sims-chain which is given by coset representatives of subgroups of $S_{\underline{a}} \wr S_n$ occurring as pointwise stabilizers of certain subsets of $\underline{a}^n$.

The *stabilizer* of the first element $f_1 = (1,\ldots,1) \in \underline{a}^n$ is $S_{\underline{a}\backslash\underline{1}} \wr S_n$. So there are a^n *coset representatives* of $S_{\underline{a}} \wr S_n / S_{\underline{a}\backslash\underline{1}} \wr S_n$ given by (ψ, id), where ψ is a function from $\underline{n}$ to $\{\mathrm{id}, (1,2), (1,3), \ldots, (1,a)\}$. Having computed the pointwise stabilizer of the first a^i elements for $0 \le i < n$ (i. e. the set of all elements in $S_{\underline{a}} \wr S_n$ which stabilize each element in $\{f_1,\ldots,f_{a^i}\}$) we can compute the pointwise stabilizers of $\{f_1,\ldots,f_\ell\}$ for $\ell \in \{a^i+1,\ldots,a^{i+1}\}$ with the following method. The set $\{a^i+1,\ldots,a^{i+1}\}$ can be partitioned into sets $\{(j-1)a^i+1,\ldots,ja^i\}$ for $j = 2,\ldots,a$. For $\ell \in \{(j-1)a^i+1,\ldots,ja^i\}$ the ℓ-th element f_ℓ in $\underline{a}^n$ is of the form

$$f_\ell = (1,\ldots,1,j,\ldots)$$

starting with $n-i-1$ entries of 1 followed by j in the $(n-i)$-th position and an arbitrary sequence of length i. Depending on j we have: If $j = 2$, then the pointwise stabilizer of $\{f_1,\ldots,f_\ell\}$ can be expressed as a direct product

$$(S_{\underline{a}\backslash\underline{1}} \wr S_{n-(i+1)}) \times S_{\underline{a}\backslash\underline{2}} \times \langle \mathrm{id} \rangle^i.$$

So there are $(n-i)(a-1)$ coset representatives of

$$(S_{\underline{a}\backslash\underline{1}} \wr S_{n-i}) / \left((S_{\underline{a}\backslash\underline{1}} \wr S_{n-(i+1)}) \times S_{\underline{a}\backslash\underline{2}} \times \langle \mathrm{id} \rangle^i \right),$$

given by (ψ, π), where $\pi \in \{\mathrm{id}, (n-i,1), \ldots, (n-i, n-i-1)\}$, $\psi(k) = \mathrm{id}$ for $k \neq n-i$ and $\psi(n-i) \in \{\mathrm{id}, (2,3), (2,4), \ldots, (2,a)\}$.

For $3 \leq j \leq a$ the pointwise stabilizer of $\{f_1, \ldots, f_\ell\}$ is given by

$$(S_{\underline{a}\setminus\underline{1}} \wr S_{n-(i+1)}) \times S_{\underline{a}\setminus\underline{j}} \times \langle \mathrm{id} \rangle^i,$$

and the $a-j+1$ coset representatives of

$$\Big((S_{\underline{a}\setminus\underline{1}} \wr S_{n-(i+1)}) \times S_{\underline{a}\setminus\underline{j-1}} \times \langle \mathrm{id} \rangle^i\Big) \,/\, \Big((S_{\underline{a}\setminus\underline{1}} \wr S_{n-(i+1)}) \times S_{\underline{a}\setminus\underline{j}} \times \langle \mathrm{id} \rangle^i\Big)$$

are given in the form (ψ, id), where $\psi(n-i) \in \{\mathrm{id}, (j, j+1), \ldots, (j,a)\}$ and $\psi(k) = \mathrm{id}$ for $k \neq n-i$.

Because of the fact that the lexicographically smallest element of an orbit is chosen to be the canonic representative of it, the *minimal distance* of a code can be read from the canonic representative in a very comfortable way. It is the *Hamming distance* between the first and the second word in the code (with respect to the numbering of the elements of $\underline{a}^n$ given above). As a matter of fact the first code word is always f_1, and the second code word is of the form $(1, \ldots, 1, 2, \ldots, 2)$ with at least one occurrence of 2. So the minimal distance is the number of 2's in the second code word of the canonic representative. This fact is very useful for recursively constructing all $[n, m]$ block codes of given minimal distance.

In addition to this let me point out that in the case $a = 2$ the $S_2 \wr S_n$ classes of functions $f\colon \{0,1\}^n \to \{0,1\}$ correspond to classes of *boolean functions* or *switching circuits*. See for instance [21] or [9].

Harrison and High [8] counted classes of *Post-functions* under various group actions. These functions are functions $f\colon \{1, \ldots, a\}^n \to \{1, \ldots, a\}$. Even in the case when $S_a \wr S_n$ is acting on the domain of these functions, the number of representatives is growing rather fast. Using the *homomorphism principle* and the method of *surjective resolution* [11] it is possible to compute a transversal of Post functions, from a transversal of $S_a \wr S_n$-orbits on the set of all functions $f\colon \{1, \ldots, a\}^n \to \{1, \ldots, a-1\}$. Iterating this process we can start constructing the classes of Post functions from a transversal of block codes.

4. Random generation of block codes

For many parameter values n, m and a there are far too many representatives in order to compute complete lists. In these situations we can apply the so called *Dixon-Wilf algorithm* [2] for generating block codes *uniformly at random*. I. e. given the isometry class ω of a block code then the probability that a random-generated block code f lies in ω equals

$$p(f \in \omega) = \frac{1}{\alpha}$$

where α is the total number of isometry classes of $[n, m]$ block codes. The Dixon-Wilf algorithm says:

THEOREM 2 *In order to generate $[n,m]$ block codes over the alphabet A uniformly at random, first compute α, the number of isometry classes of $[n,m]$ block codes over A. Then choose a conjugacy class $\mathcal{C}$ of $S_A \wr S_n$ with probability*

$$p(\mathcal{C}) := \frac{|\mathcal{C}| \left| \binom{A^n}{m}_{(\psi,\pi)} \right|}{n!\,|A|^{n!}\,\alpha},$$

where

$$\binom{A^n}{m}_{(\psi,\pi)}$$

is the set of fixed points of an arbitrary element (ψ,π) of $\mathcal{C}$ acting on all m-sets of A^n. Finally construct a characteristic function $\chi : A^n \to \{0,1\}$ of an $[n,m]$ block code that takes values 0 or 1 on the cycles of $(\psi,\pi) \in \mathcal{C}$ which are distributed uniformly at random.

In a first step this algorithm has to compute the cycle index of $S_A \wr S_n$ as described above in order to compute α. Then it must determine the conjugacy classes of the acting group which is the complete monomial group of degree n over S_A. These conjugacy classes can be described by integer matrices $(a_{i,k})$ holding the *cycle types* of the *cycleproducts* of (ψ,π) ([11]). In other words, a matrix having n columns and as many rows as S_A has conjugacy classes corresponds to a conjugacy class of $S_A \wr S_n$ if and only if

$$\sum_{i,k} a_{ik} = n.$$

In the next step the probabilities of the conjugacy classes can be computed. Finally the construction of the characteristic functions of block codes which are fixed points of the chosen element (ψ,π) in the chosen conjugacy class $\mathcal{C}$ must be organized such that it produces only functions of weight m.

In order to minimize the amount of work before the algorithm actually starts to generate block codes it is useful to start the generation at once after having computed the information on the first conjugacy class, and evaluate further conjugacy classes and their probabilities only if required. This means we have to compute $p(\mathcal{C}_i)$ only if the random number (lying in $[0,1[$) determining which conjugacy class to choose exceeds $\sum_{j=1}^{i-1} p(\mathcal{C}_j)$. The efficiency of this method heavily depends on the numbering of the conjugacy classes. So this numbering should be chosen such that $p(\mathcal{C}_i) \geq p(\mathcal{C}_{i+1})$ which leads to $\mathcal{C}_1 = \{\mathrm{id}\}$.

In the computer algebra system SYMMETRICA there are all kinds of routines implemented in order to compute orbit transversals of block codes or to generate them uniformly at random.

Acknowledgments

The author wants to express his thanks especially to Prof. A. Kerber for his support and guidance during the preparation of this article. The main results of this work were presented at the conference *Groups in Action '96,* in Thurnau.

Notes

1. The cycle index of a finite group G acting on a finite set X is the polynomial

$$Z(G,x) := \frac{1}{|G|} \sum_{g \in G} \prod_{i=1}^{|X|} x_i^{a_i(\bar{g})},$$

where $(a_1(\bar{g}), \ldots, a_{|X|}(\bar{g}))$ is the *cycle type* of the induced permutation $\bar{g}$ of g on X. I. e. the permutation $\bar{g}$ can be expressed as a product of $a_i(\bar{g})$ disjoint cycles of length i for $i = 1, \ldots, |X|$.

References

1. C.J. Colbourn and R.C. Read. Orderly algorithms for generating restricted classes of graphs. *Journal of Graph Theory*, 3:187 – 195, 1979.
2. J.D. Dixon and H.S. Wilf. The random selection of unlabeled graphs. *Journal of Algorithms*, 4:205 – 213, 1983.
3. H. Fripertinger and A. Kerber. Isometry Classes of Indecomposable Linear Codes. In G. Cohen, M. Giusti, and T. Mora, editors, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, 11th International Symposium, AAECC-11, Paris, France, July 1995*, volume 948 of *Lecture Notes in Computer Science*, pages 194–204. Springer, 1995.
4. H. Fripertinger. Enumeration of isometry classes of linear (n,k)-codes over $GF(q)$ in SYMMETRICA. *Bayreuther Mathematische Schriften*, 49:215 – 223, 1995. ISSN 0172-1062.
5. H. Fripertinger. Zyklenzeiger linearer Gruppen und Abzählung linearer Codes. *Séminaire Lotharingien de Combinatoire*, Actes 33:1 – 10, 1995. ISSN 0755-3390.
6. H. Fripertinger. Enumeration of Linear Codes by Applying Methods from Algebraic Combinatorics. *Grazer Math. Berichte*, 328:31 – 42, 1996.
7. R. Grund. Symmetrieklassen von Abbildungen und die Konstruktion von diskreten Strukturen. *Bayreuther Mathematische Schriften*, 31:19 – 54, 1990. ISSN 0172-1062.
8. M.A. Harrison and R.G. High. On the Cycle Index of a Product of Permutation Groups. *Journal of Combinatorial Theory*, 4:277 – 299, 1968.
9. M.A. Harrison. Counting Theorems and their Applications to Switching Theory. In A. Mukhopadyay, editor, *Recent Developments in Switching Functions*, chapter 4, pages 85 – 120. Academic Press, 1971.
10. A. Kerber. Der Zykelindex der Exponentialgruppe. *Mitteilungen aus dem Mathematischen Seminar Giessen*, 98:5 – 20, 1973.
11. A. Kerber. *Algebraic Combinatorics via Finite Group Actions.* B.I. Wissenschaftsverlag, Mannheim, Wien, Zürich, 1991. ISBN 3-411-14521-8.
12. A. Kerber. Algebraic Combinatorics in Bayreuth. *Séminaire Lotharingien de Combinatoire*, B34j, 1995.
`http://cartan.u-strasbg.fr/~slc//divers/../wpapers/s34bayreuth.html`.
13. A. Kerber. Anwendungsorientierte Theorie endlicher Strukturen. To be published.
14. H. Lehmann. Das Abzähltheorem der Exponentialgruppe in gewichteter Form. *Mitteilungen aus dem Mathem. Seminar Giessen*, 112:19 – 33, 1974.
15. H. Lehmann. *Ein vereinheitlichender Ansatz für die REDFIELD – PÓLYA – de BRUIJNSCHE Abzähltheorie.* PhD thesis, Universität Giessen, 1976.

16. E.M. Palmer and R.W. Robinson. Enumeration under two representations of the wreath product. *Acta Mathematica*, 131:123 – 143, 1973.
17. E.M. Palmer and Robinson R.W. The matrix group of two permutation groups. *Bull. Amer. Math. Soc.*, 73:204 – 207, 1967.
18. G. Pólya. Kombinatorische Anzahlbestimmungen für Gruppen, Graphen und chemische Verbindungen. *Acta Mathematica*, 68:145 – 254, 1937.
19. R.C. Read. Every one a winner. *Ann. Discrete Mathematics*, 2:107 – 120, 1978.
20. C.C. Sims. Computational methods in the study of permutation groups. *Computational Problems in Abstract Algebra*, pages 169 – 183, 1970.
21. D. Slepian. On the Number of Symmetry Types of Boolean Functions of n Variables. *Canad. J. Math.*, 5:185 – 193, 1953.
22. SYMMETRICA. A program system devoted to representation theory, invariant theory and combinatorics of finite symmetric groups and related classes of groups. Copyright by "Lehrstuhl II für Mathematik, Universität Bayreuth, 95440 Bayreuth". `http://www.mathe2.uni-bayreuth.de/axel/symneu_engl.html`.