**JOURNAL OF ALGEBRAIC STATISTICS**
Volume 9, No. 1, 2018, p. 75-80
https://publishoa.com
ISSN: 1309-3452
DOI: https://doi.org/10.52783/jas.v9i1.1441

# Symmetric Generatic Generations and an Algorithm to Prove Relations

**Rakesh Chandra Bhadula**
Department of Mathematics, Graphic Era Hill University, Dehradun, Uttarakhand, India 248002

**Abstract**

When researching this thesis, we found that various progenitors, including $3^{*56}$:($2^3$:(3:7), Factoring m*: N by the First Order Relations in Permutation Progenitors, and Symmetric Generation, each exhibit homomorphic pictures. To generate the Cayley graph of the ratio 3:($2\times S_5$) over $D_{12}$, we manually do Double Coset Enumeration. When dealing with finitely presented groups, coset enumeration is the primary technique for addressing the word problem. This method has been around for quite some time, and it was really one of the earliest uses of electronic computers for purely mathematical purposes. The current programs' functionality is limited by the size of the coset table that can be stored in the computer's memory. In certain situations, a significant space savings may be realized by the enumeration of double cosets. There is a description of an algorithm followed by some implementation details. We also show some of the isomorphism types and original symmetric presentations of finite groups, and we explain them as homomorphic pictures of their progenitors.

**Keywords:** Symmetric, Generation, Algorithm, Relation, Group

## Introduction

Finite images of progenitors typically need to be shown to be isomorphic to permutation groups in group theory. This procedure, however, requires solving a tremendous number of right coset connections between words. Whether or whether a certain word is the identity is very tough to discern. To apply mathematics to a real-world issue, one must typically first model the situation theoretically, often with stringent limitations, idealizations, or simplifications, before solving the mathematical problem and extracting inferences about the real-world problem from the mathematical answers. There has been a paradigm change in the last 60 years or more, with the inverse becoming the norm. The argument is that the world has succeeded before and after the advent of mathematical modeling. Adding relations of the kind 2*n:N to the progenitor allowed us to generate finite homomorphic pictures. In particular, our environment is home to an overwhelming number of intricate processes and systems that have long piqued the curiosity of scientists for their seemingly effortless perfection. The ability to model these ideas analytically and apply them to a wider set of issues has proven useful in a wide range of fields.

When dealing with finitely presented groups, coset enumeration is the primary technique for addressing the word problem. This method has been around for quite some time, and it was really one of the earliest uses of electronic computers for purely mathematical purposes. In certain situations, a significant space savings may be realized by the enumeration of double cosets. The Todd-Coxeter approach is still widely used as a benchmark for other coset enumeration algorithms. For groups with a finite presentation, it may be seen as a way to build permutation representations. To reduce the amount of space (and time) required, it is preferable to enumerate double cosets rather than single ones. Linton has created two applications to enumerate double-cosets. The procedure in the former is seen as a straightforward extension of the common single-coset enumeration since it is a rectified version of the method. Similar to the current approach, but with different computations, is the later one.

## Literature Review

**El-Sayed, Fatma & Moussa, Mahmoud & Abd elkader, Hatem. (2014)** DNA computing, which uses molecular biology to model DNA's bimolecular structure and computation, was developed in 2014. DNA cryptography is an emerging area that is currently being researched all over the world. DNA computing has been proposed as a tool for use in cryptography and steganography, raising the prospect of indestructible code for the first time. In order to circumvent the drawbacks of the traditional One Time Pad (OTP) encryption, this study suggested a novel DNA cryptographic technique that makes advantage of the fundamental properties of DNA and amino acid coding. The suggested technique has the dual functionality of encryption and concealing, which is a major advantage. The suggested approach improves

**JOURNAL OF ALGEBRAIC STATISTICS**
Volume 9, No. 1, 2018, p. 75-80
https://publishoa.com
ISSN: 1309-3452
DOI: https://doi.org/10.52783/jas.v9i1.1441
the safety of the OTP cipher as well. We put the proposed algorithm through its paces in terms of randomness testing by running it through the NIST evaluation. The results of the study demonstrated that the proposed algorithm outperformed its predecessors in terms of time, capacity, and robustness.

**Maurer, Peter. (2014)** New forms of symmetry detection are the focus of symmetry research in 2014. We provide a general-purpose technique for finding permutation symmetries, including those for which no prior algorithms have been developed. Although libraries are required for general symmetry detection, detection of parametrizable symmetries (such as complete, partial, rotational, and dihedral symmetry) does not need them. It often outpaces conventional methods in terms of speed. It's also easier to implement into preexisting software and simpler than most current methods.

**Khalil, Shuker & Alradha, Marwa. (2017)** The goal of this paper, which was published in 2017, is to provide a new subfield of pure algebra. Also, the notions of permutation topological ρ -algebra, ρ -subalgebra, ρ-ideal, ρ¯-ideal are introduced and explored. A counterexample is given to show that -algebra need not be BCK ρ -algebra or d^*- algebra. In addition, the topics presented in this article are illustrated with various instances.

**Achuthshankar, Aswin & Achuthshankar, Aswathy. (2015)** data encryption is the gold standard for safe online chats. This paper proposes a straightforward, quick, and secure encryption method, despite the abundance of existing options. Both tiny and big data may be encrypted effectively using this approach. The method utilizes two user-specified files (a symmetric key file and a message file). This method makes it impossible for hackers to determine what kind of message file was used. This effectively makes deciphering ciphertext difficult. These advantages may be seen in the suggested symmetric encryption system. To begin, the same secure, easy, and quick method is used for both the encryption and decryption processes. Second, this method can be used to encrypt virtually any data format. Finally, the user has complete freedom to choose the kind of key file to use. In this study, we compare the suggested system to both classic encryption methods and the state-of-the-art; the findings show that the new technique is both more secure and quicker.

**Symmetric Generation**

Finite images of progenitors typically need to be shown to be isomorphic to permutation groups in group theory. This is accomplished by enumerating double cosets of the picture over a transitive subgroup. This procedure, however, requires solving a tremendous number of right coset connections between words. Whether or whether a certain word is the identity is very tough to discern. The word issue for a finitely generated group G in combinatorial group theory is the computational challenge of determining if two words in the generators represent the same element. There is no known method that can produce a finite set of words and their associated relations. Yet, methods to determine connections between all words in certain classes already exist. We provide a novel algorithm that, for a large set of categories, lists every possible pair of words and their links to one another. This thesis relies on our method of manual double coset enumeration to create homomorphic pictures of the progenitors. Each element of the progenitors may be represented as nw, where n is an integer between 1 and N and w is a symmetric generating word. A finite image G may be constructed by factoring the progenitor by another relation of the kind nw(t1, t2,..., tn), where n N and w is a word in T = t1, t2,..., tn.

$$\frac{m^{*n}:N}{n_1w_1,..,n_sw_s}$$

may be recognized.

**JOURNAL OF ALGEBRAIC STATISTICS**
Volume 9, No. 1, 2018, p. 75-80
https://publishoa.com
ISSN: 1309-3452
DOI: https://doi.org/10.52783/jas.v9i1.1441

**Table 1:** Images of Progenitor $3^{*56}$:($2^3$ :(3:7)).

| a | b | c | d | e | f | g | Order of G | Shape of G |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 2 | 6 | 367416 | $3^7 \cdot N$ |
| 0 | 0 | 0 | 0 | 3 | 0 | 3 | 4032 | $2^6 \cdot N$ |
| 0 | 0 | 0 | 2 | 0 | 0 | 0 | 1512 | $3 \cdot L_2(8)$ |
| 3 | 3 | 0 | 9 | 0 | 4 | 8 | 304819200 | $A_{10}$ |

**Permutation Progenitors**

The centralizers of sample members from each distinct group are then calculated. Finally, we calculate each representative's orbit. There is a methodical procedure that must be followed if we want to factor $2^{*10}$: ($5^2$: 2) using first-order relations. Here we show you how it's done, starting with some MAGMA code.

```
S:=Sym(10);
xx:=S!(2, 4, 6, 8, 10);
yy:=S!(1, 6)(2, 7)(3, 8)(4, 9)(5, 10);
N:=sub<S|xx,yy>;
#N;
/*50*/
FPGroup(N);
 Finitely presented group on 2 generators
Relations
    Relations
    .1^5 = Id
    .2^2 = Id

.1^-1 * .2 * .1^-1 * .2 * .1 * .2 * .1 * .2 = Id
```

The display of this requires an x and y-axis conversion $2^{*10}$: ($5^2$: 2).

The next step is to determine the N types of conjugacy that make up our sample population. By applying this to any element k of N, we may determine the classes of conjugacy for the whole set of N. We perform this for every element, and when computing the conjugacy classes of very big groups, we enlist the help of MAGMA. In MAGMA, this is accomplished as follows:

```
C:=Classes(N);
#C;
20
for i in [2..20] do
i, Orbits(Centraliser(N,C[i][3]));
end for;
for j in [2..20] do
C[j][3];
for i in [1..50] do if ArrayP[i] eq C[j][3] then Sch[i];
end if; end for; end for;
```

Right multiplying a representative of class C2 by a representation of orbit C3, and so on, until class C20 is obtained (since C1 is the identity class), yields the relations of first order. We'll use an example to illustrate how to determine C2's first-order relation. Table 2 indicates that the numbers [1, 6, 10], [5, 9], [4, 8], [3, 7], and [2] correspond to the first orbit, and that the letter y stands in for the class. The correct method here is to multiply by 6, although any other method would work as well. When t ∼ t1, we get y ∗ t. Now we identify an x-y permutation that uses t1 to get t6; this y. This leads us to the first relation being $(yt)^a$.

**JOURNAL OF ALGEBRAIC STATISTICS**
Volume 9, No. 1, 2018, p. 75-80
https://publishoa.com
ISSN: 1309-3452
DOI: https://doi.org/10.52783/jas.v9i1.1441

**Table 2:** Conjugacy Classes of N = $5^2$: 2

| Class | Representative of the class | # of elements in the class | Orbits |
|---|---|---|---|
| $C_1$ | Identity | 1 | {1},{2},{3},{4},{5},{6},...,{10} |
| $C_2$ | $(y)$=(1, 6)(2, 7)(3, 8)(4, 9)(5, 10) | 5 | {1, 6, 10, 5, 9, 4, 8, 3, 7, 2 } |
| $C_3$ | $(xy)^2$=(1, 3, 5, 7, 9)(2, 4, 6, 8, 10) | 1 | {1, 6, 8, 10, 3, 2, 5, 4, 7, 9 } |
| $C_4$ | $(x^2y)^2$=(1, 5, 9, 3, 7)(2, 6, 10, 4, 8) | 1 | {1, 6, 8, 10, 3, 2, 5, 4, 7, 9 } |
| $C_5$ | $(yx^{-2})^2$ =(1, 7, 3, 9, 5)(2, 8, 4, 10, 6) | 1 | {1, 6, 8, 10, 3, 2, 5, 4, 7, 9} |
| $C_6$ | $(yx^{-1})^2$=(1, 9, 7, 5, 3)(2, 10, 8, 6, 4) | 1 | {1, 6, 8, 10, 3, 2, 5, 4, 7, 9 } |
| $C_7$ | $(yxy)$=(1, 3, 5, 7, 9) | 2 | {1, 3, 5, 7, 9 },{2, 10, 8, 6, 4 } |
| $C_8$ | $(yx^2y)$=(1, 5, 9, 3, 7) | 2 | { 1, 5, 9, 3, 7 }, {2, 10, 8, 6, 4} |
| $C_9$ | $(yx^{-2}y)$=(1, 7, 3, 9, 5) | 2 | {1, 7, 3, 9, 5}, {2, 10, 8, 6, 4} |
| $C_{10}$ | $(yx^{-1}y)$=(1, 9, 7, 5, 3) | 2 | { 1, 9, 7, 5, 3 }, {2, 10, 8, 6, 4 } |

**Double Coset Enumerations**

**Definition 1.** HG is the largest normal subgroup of G if and only if there is no normal subgroup N of G such that HG includes G; H<N<G.

**Lemma 2.** It follows that x is a member of G if and only if the groups H and K are finite |HxK| = |K| × |K|/|$K^x$ ∩ H|.

**Table 3:** Powers of t

| Inverses: | Powers: | Splits: |
|---|---|---|
| $t_1 = t_{19}^{-1}$ | $t_1^2 = t_7$, $t_1^3 = t_{13}$, $t_1^4 = t_{19}$ | $t_1 = t_7t_{19}$ |
| $t_2 = t_{20}^{-1}$ | $t_2^2 = t_8$, $t_2^3 = t_{14}$, $t_2^4 = t_{20}$ | $t_2 = t_8t_{20}$ |
| $t_3 = t_{21}^{-1}$ | $t_3^2 = t_9$, $t_3^3 = t_{15}$, $t_3^4 = t_{21}$ | $t_3 = t_9t_{21}$ |
| $t_4 = t_{22}^{-1}$ | $t_4^2 = t_{10}$, $t_4^3 = t_{16}$, $t_4^4 = t_{22}$ | $t_4 = t_{10}t_{22}$ |
| $t_5 = t_{23}^{-1}$ | $t_5^2 = t_{11}$, $t_5^3 = t_{17}$, $t_5^4 = t_{23}$ | $t_5 = t_{11}t_{23}$ |
| $t_6 = t_{24}^{-1}$ | $t_6^2 = t_{12}$, $t_6^3 = t_{18}$, $t_6^4 = t_{24}$ | $t_6 = t_{12}t_{24}$ |
| $t_7 = t_{13}^{-1}$ | $t_7^2 = t_{19}$, $t_7^3 = t_1$, $t_7^4 = t_{13}$ | $t_7 = t_{19}t_{13}$ |
| $t_8 = t_{14}^{-1}$ | $t_8^2 = t_{20}$, $t_8^3 = t_2$, $t_8^4 = t_{14}$ | $t_8 = t_{20}t_{14}$ |
| $t_9 = t_{15}^{-1}$ | $t_9^2 = t_{21}$, $t_9^3 = t_3$, $t_9^4 = t_{15}$ | $t_9 = t_{21}t_{15}$ |
| $t_{10} = t_{16}^{-1}$ | $t_{10}^2 = t_{22}$, $t_{10}^3 = t_4$, $t_{10}^4 = t_{16}$ | $t_{10} = t_{22}t_{16}$ |
| $t_{11} = t_{17}^{-1}$ | $t_{11}^2 = t_{23}$, $t_{11}^3 = t_5$, $t_{11}^4 = t_{17}$ | $t_{11} = t_{23}t_{17}$ |
| $t_{12} = t_{18}^{-1}$ | $t_{12}^2 = t_{24}$, $t_{12}^3 = t_6$, $t_{12}^4 = t_{18}$ | $t_{12} = t_{24}t_{18}$ |
| $t_{13} = t_7^{-1}$ | $t_{13}^2 = t_1$, $t_{13}^3 = t_{19}$, $t_{13}^4 = t_7$ | $t_{13} = t_1t_7$ |
| $t_{14} = t_8^{-1}$ | $t_{14}^2 = t_2$, $t_{14}^3 = t_{20}$, $t_{14}^4 = t_8$ | $t_{14} = t_2t_8$ |
| $t_{15} = t_9^{-1}$ | $t_{15}^2 = t_3$, $t_{15}^3 = t_{21}$, $t_{15}^4 = t_9$ | $t_{15} = t_3t_9$ |

**JOURNAL OF ALGEBRAIC STATISTICS**
Volume 9, No. 1, 2018, p. 75-80
https://publishoa.com
ISSN: 1309-3452
DOI: https://doi.org/10.52783/jas.v9i1.1441

| | | |
|---|---|---|
| $t_{16} = t_{10}^{-1}$ | $t_{16}^2 = t_4, t_{16}^3 = t_{22}, t_{16}^4 = t_{10}$ | $t_{16} = t_4 t_{10}$ |
| $t_{17} = t_{11}^{-1}$ | $t_{17}^2 = t_5, t_{17}^3 = t_{23}, t_{17}^4 = t_{11}$ | $t_{17} = t_5 t_{11}$ |
| $t_{18} = t_{12}^{-1}$ | $t_{18}^2 = t_6, t_{18}^3 = t_{24}, t_{18}^4 = t_{12}$ | $t_{18} = t_6 t_{12}$ |
| $t_{19} = t_1^{-1}$ | $t_{19}^2 = t_{13}, t_{19}^3 = t_7, t_{19}^4 = t_1$ | $t_{19} = t_{13} t_1$ |
| $t_{20} = t_2^{-1}$ | $t_{20}^2 = t_{14}, t_{20}^3 = t_8, t_{20}^4 = t_2$ | $t_{20} = t_{14} t_2$ |
| $t_{21} = t_3^{-1}$ | $t_{21}^2 = t_{15}, t_{21}^3 = t_9, t_{21}^4 = t_3$ | $t_{21} = t_{15} t_3$ |
| $t_{22} = t_4^{-1}$ | $t_{22}^2 = t_{16}, t_{22}^3 = t_{10}, t_{22}^4 = t_4$ | $t_{22} = t_{16} t_4$ |
| $t_{23} = t_5^{-1}$ | $t_{23}^2 = t_{17}, t_{23}^3 = t_{11}, t_{23}^4 = t_5$ | $t_{23} = t_{17} t_5$ |
| $t_{24} = t_6^{-1}$ | $t_{24}^2 = t_{18}, t_{24}^3 = t_{12}, t_{24}^4 = t_6$ | $t_{24} = t_{18} t_6$ |

We have $t_{i's}$ of order 2, if memory serves. To generate a finite homomorphic image of order 1440, we factor G by the following relations: $(y^{-1} * x^{-1} * t)^6$, $(x^{-1} * y^{-1} * x * t)^4$, $(x^2 * t)^8$, $(x * t)^5$. The enumeration of double cosets starts with an estimate of how many such sets there are likely to be. Because $\frac{|G|}{|N|}$= the number of unique cosets, this is a trivial condition to test. We expect $\frac{|G|}{|N|}$= 1440/36 = 36 unique cosets. Before we go any further, we should point out that we discovered t1 ~ t6, t2 ~ t7, t3 ~ t8, t4 ~ t9, and t5 ~ t10. Here are how the connections are checked:

```
ts[6] eq(2, 10)(3, 9)(4, 8)(5, 7)t_1,
ts[7] eq (1, 3)(4, 10)(5, 9)(6, 8)t_2,
ts[8] eq (1, 4, 5, 2)(3, 8)(6, 9, 10, 7)t_3,
ts[9] eq (1, 7)(2, 6)(3, 5)(8, 10)t_4,
ts[10] eq (1, 9)(2, 8)(3, 7)(4, 6)t_5
```

Hence, instead of dealing with 10 t's or 10 letters, we'll only focus on 5. Take the [∗] notation for the double coset, where NeN = N. If you know how many elements in N fix the coset NeN, then you know how many individual cosets there are in [∗]. Number of unique single cosets in NeN = $\frac{|N|}{|N^{(e)}|}$= $\frac{40}{40}$= 1 because every member of N fixes the coset NeN (since N is transitive). To proceed, we choose a person with a 1 in their orbit, {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}. Here, Option 1 will do the trick. All ten symmetric generators will advance since there are ten elements in the orbit {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}.

**Conclusion**

For a small subset of presentations, the proposed program provides a significant improvement over prior (single) coset enumeration method. In addition to providing additional information about the group's structure to the user, the current double-coset enumeration algorithm differs from traditional coset enumeration methods in how it deals with the group's elements. Certain mathematical operations, including inversion and multiplication, may be carried out by hand (or by machine) using simple algorithms.

**References**

1. Bahşi, Mustafa & Mező, I. & Solak, Süleyman. (2014). A symmetric algorithm for hyper-fibonacci and hyper-lucas numbers. Annales Mathematicae et Informaticae. 43. 19-27.
2. El-Sayed, Fatma & Moussa, Mahmoud & Abd elkader, Hatem. (2014). A Symmetric Encryption Algorithm based on DNA Computing. International Journal of Computer Applications. 97. 10.5120/17094-7634.

**JOURNAL OF ALGEBRAIC STATISTICS**
Volume 9, No. 1, 2018, p. 75-80
https://publishoa.com
ISSN: 1309-3452
DOI: https://doi.org/10.52783/jas.v9i1.1441
3. Maurer, Peter. (2014). A universal symmetry detection algorithm. SpringerPlus. 4. 1-4. 10.7873/DATE.2014.312.
4. Khalil, Shuker & Alradha, Marwa. (2017). Characterizations of ρ-algebra and Generation Permutation Topological ρ-algebra Using Permutation in Symmetric Group. American Journal of Mathematics and Statistics. 7. 152-159. 10.5923/j.ajms.20170704.02.
5. Achuthshankar, Aswin & Achuthshankar, Aswathy. (2015). A novel symmetric cryptography algorithm for fast and secure encryption. 1-6. 10.1109/ISCO.2015.7282273.
6. Dustin J. Grindstaff. Symmetric Presentations and Generation. CSUSB Thesis, 2015.
7. Morrison, Katherine. (2015). An Enumeration of the Equivalence Classes of Self-Dual Matrix Codes. Advances in Mathematics of Communications. 9. 10.3934/amc.2015.9.415.
8. Baleanu, Dumitru & Mousalou, Asef & Rezapour, Shahram. (2017). On the existence of solutions for some infinite coefficient-symmetric Caputo-Fabrizio fractional integro-differential equations. Boundary Value Problems. 2017. 10.1186/s13661-017-0867-9.
9. Petersen, Kyle. (2016). A Two-Sided Analogue of the Coxeter Complex. Electronic Journal of Combinatorics. 25. 10.37236/8015.
10. Billey, Sara & Konvalinka, Matjaž & Petersen, Kyle & Slofstra, William & Tenner, Bridget. (2016). Parabolic Double Cosets in Coxeter Groups. Electronic Journal of Combinatorics. 25. 10.37236/6741.