VOL. 9, NO. 12, DECEMBER 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

COMBINATORIAL GRAY CODE FOR GENERATING TREE OF PERMUTATION WITH TWO CYCLES

Sulistyo Puspitodjati, Henny Widowati, Asep Juarna and Djati Kerami
Gunadarma University, Depok, Indonesia
E-Mail: spuspitodjati@yahoo.com

ABSTRACT

Combinatorial Gray Code is a listing of all considered combinatorial objects so that every two successive objects differ in some pre-specified way. This paper concerns of a combinatorial class namely permutations with two cycles. In our research these permutations have been developed using generating tree. The generating tree is then coded, and listing the node of this tree will be shoed as a Gray code. In this paper we proposed a new combinatorial Gray code. We give the construction of the Gray code for this coded generating tree of generating permutation with two cycles. The Gray code will be measured by its Hamming distance in order to show that it is a Gray code.

Keywords: gray code, permutation with two cycles, generating tree.

INTRODUCTION

Gray codes have several applications in a diverse area, including error correction, encryption, databases, software and hardware testing, biology, and puzzles ([2], [3], [5], [10]). The classic Gray code is the Binary Reflected Gray Code (BRGC), was developed and patented by Frank Gray in 1953 for use in converting analog signals to digital signals. But there are many other possible minimal change orderings on binary tuples as well as other combinatorial objects. Other combinatorial objects, that ordered such that the order of objects have minimal change, are permutations, trees, partitions, combinations, long-run (maximal run-lengths), balanced, monotonic, and single-track.

There are many different ways to define Gray codes. The classic Binary Reflected Gray codes is a lists of binary words such that the successive words differ one digit, i.e. its Hamming distance is one. The definition becomes generalized as any method to generate combinatorial objects so that successive objects differ in some pre-specified way [10]. Ruskey in [5], define a combinatorial Gray code of combinatorial objects *S* as a listing of elements *S*, such that two successive element on the list is in *closeness relation* on *S*.

A number of author have been interested in Gray codes and generating algorithms for permutations and their restrictions (unrestricted, with given ups and downs, involutions and fixed-point free involutions, derangements, permutation with a fixed number of cycles) or their generalizations (multi-set permutations).

In [2] a general technique is presented for the generation of Gray code for a large class of combinatorial families; it is based on ECO method and produces objects by their encoding given by their generating tree (in some cases the obtained encodings can be translated into objects). Motivated by these papers, we generate a Gray code by encoding the generating tree of permutation with two cycles [11].

PERMUTATION WITH CYCLES

A permutation of a set *S* is an arrangement of the elements of *S* into a linear order using each object exactly once. We also can define permutation as a bijection π: $S \rightarrow S$.

A cycle of length *m* in a permutation is a sequence of distinct elements $a_1, a_2, ..., a_k$ such that $a_i = \pi(a_{i-1})$ for $i = 2,3,..k$ and $a_1 = \pi(a_k)$. Such cycle is written as $(a_1, a_2, ..., a_k)$. All permutation can be decomposed into the disjoint unions of their cycles. For example, a one line notation of permutation 421365 would be in a cycle notation as (1 4 3) (2) (5 6)

The number of permutation of a set with *n* elements is *n* factorial, $n! = n\,(n-1)...21$, whereas the number of permutation of a set with *n* elements with *m* cycles is the signless Stirling number of the first kind, $c(n,m)$

$$c(n,m) = \begin{cases} (n-1)! & m = 1 \\ (n-1)c(n-1,m) + c(n-1,m-1) & 1 < m < n \\ 1 & m = n \end{cases} \tag{1}$$

GENERATING TREE

One approach for generating combinatorial objects is a generating tree or is often identified as ECO (enumerating combinatorial objects) method ([1]). In ECO method, each object obtained from the smaller object is expanded with the formulation of the so-called succession rules. This succession rules can be represented in a tree and called generating trees ([4]). Precisely, a generating tree is a rooted, labelled, and typically infinite tree such that the label of a node determines the labels of its children [7]. Merlini in [8] defines generating tree as a rooted labelled tree with the property that if v_1 and v_2 are any two nodes with the same label then, for each label *l*, v_1 and v_2 have exactly the same number of children with label *l*. To define a generating tree it therefore suffices to specify: the label of the root and a set of rules explaining how to derive from the label of a parent

VOL. 9, NO. 12, DECEMBER 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

node the labels of all of its children. Sometimes the labels of the tree are taken to be natural numbers, but this is not necessary. This generating tree have been shown efficient in the context of combinatorial generation, the time to produce the *N* objects of size *n* is O (*N*).

A succession rule Ω is a system ((*a*), $\mathcal{P}$), consisting of an axiom (*a*) and a set of production or rewriting rules $\mathcal{P}$ defined on a set of labels $M \subset N^+$:

$$\Omega = \begin{cases} (a) \\ (k) \;\mapsto\; (e_1(k))(e_2(k))\ldots(e_k(k)) \quad k \in M \end{cases}$$

(2)

where $a \in$ M and e_i is a function on M → M.

The succession rule Ω can also be described with a rooted tree where the nodes are the labels of: the axiom (*a*) is the root of the tree and each node with label (*k*) generates *k* sons with labels ($e_1(k)$), ($e_2(k)$), . . . , ($e_k(k)$). The one of well known succession rule *Ω* is

$$\Omega = \begin{cases} (2) \\ (k) \;\mapsto\; (2)(3)\ldots(k)(k+1) \quad k \geq 2 \end{cases} \qquad (3)$$

defining the sequence of Catalan numbers and whose first levels of the related generating tree are shown in Figure-1

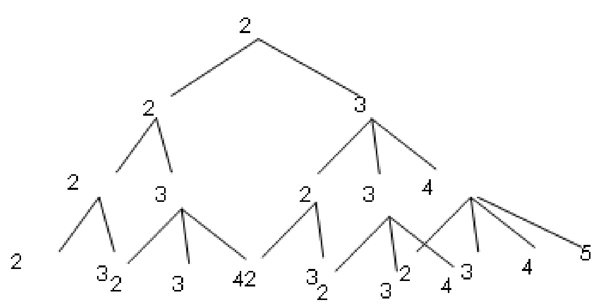

Figure-1. Generating tree for Catalan numbers of the words with length 4.

GENERATING TREE FOR PERMUTATION WITH CYCLES

Puspitodjati in [11] generate permutation *n* with *m* cycles using generating tree as in Figure-2. Generation of permutation with two cycles in [11] was developed based on the nature proposed by Baril in [9]. According [9]:

1. If $\pi \in S_{n-1,m}$, $n \geq 2$, $1 \leq m \leq n$, then it can be obtained $\pi' \in S_{n,m}$, by mapping $\pi'(i) = n$ dan $\pi'(n) = \pi(i)$, $1 \leq i < n$, e.g.: $S_{3,2}(12)(3) = 213$ → $S_{4,2}$: 4132, 2431, 2143, as much 3 of $S_{3,2}$ for $n-1 = 3$
2. if $\pi \in S_{n-1,m-1}$, $n \geq m \geq 2$, then it can be obtained $\pi'' \in S_{n,m}$, by adding *n* to position *n*, e.g. $S_{3,1}$ (123) = 231 → $S_{4,2}$ (123) (4) = 2314

The allocation of *n* to members of $S_{n-1,\,m}$ was done by putting *m* everywhere following the Johnson-Trotter's generating permutation. Here the cycle is not written in canonical form, each cycle is written to the smallest element as the first element of cycles, and the cycles are ordered by the first element of the sequence from the smallest to largest one. At each $S_{n,m}$ consists of two cycles (s_1) (s_2). The permutation *n* with two cycles here was divided into two groups, the group π_B^n = ($e_1e_2 \ldots e_{n-1}$) (*n*) that is the second cycle contains *n* only, i.e. (s_2) = (*n*) and others as π_T^n = ($e_1e_2 \ldots e_j$) ($e_{j+1} \ldots e_n$). The naming of this group follows the naming east-west by [6].

The generating trees developed, starting with the root node (1) (2), which is the smallest element of *n* permutations with 2 cycles, in this case *n* = 2 or $S_{2,2}$. The children of the root of the tree are of $S_{3,\,2}$, i.e. (1) (23), (13) (2), and (12) (3). Successively, the tree generated new children using rules: (1) π_T^n would deliver *n+1* children of π_T^{n+1}, (2) π_B^n would deliver *n*+1 children of π_T^{n+1} and *n* children of π_B^{n+1}. The succession rules of the generating tree then as follow:

$$\Omega = \begin{cases} (1)_B \\ (1)_B \;\mapsto\; (2)_T(2)_T(2)_B \\ (k)_B \;\mapsto\; [(k+1)_T]^{k-1}[(k+1)_B]^k \;, k \geq 3 \\ (k)_T \;\mapsto\; [(k+1)_T]^{k+1}, k \geq 3 \end{cases} \qquad (4)$$

The first four level of the generating tree of permutation with two cycles can be described as shown in Figure-2.

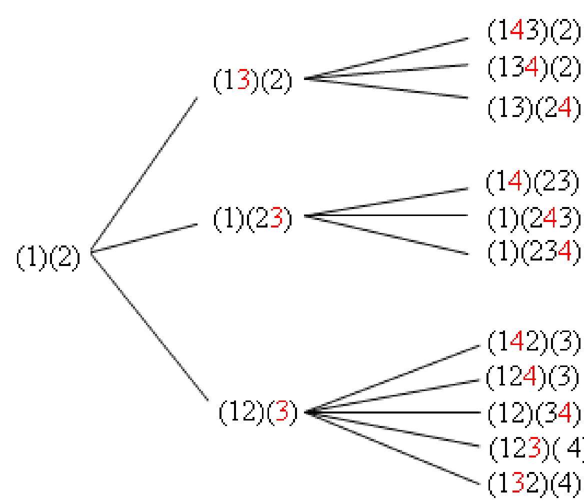

Figure-2. The generating tree for permutation with 2 cycles.

When nodes of the tree based on the number of π_B^n or π_T^n, the first three level of generating tree for permutation with two cycles then as shown in Figure-3.

ARPN Journal of Engineering and Applied Sciences

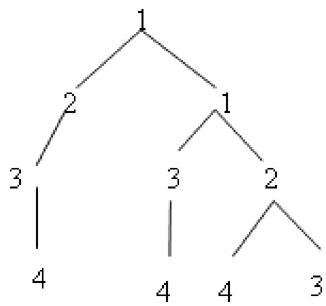

Figure-3. Generating tree of $S_{5,2}$.

CONSTRUCTION OF THE GRAY CODE FROM THE GENERATING TREE

Traversing the generating tree in such a way that two consecutive words differ only for one digit and it is a "Gray code", shown in [2]. One of codes proposed is the Gray code from the gene-rating tree of Catalan numbers (Figure-1), and by his strategy, the list of the words at level 3 generated is < 2222, 2223, 2233, 2234, 2232, 2332, 2334, 2333, 2343, 2344, 2345, 2342, 2322, 2323 >.

The idea of Bernini in [2], that is read a generating tree to generate Gray code, applied in this paper by reading from left to right the generating tree of permutation with two cycles as in Figure-3. If it is formulated such that a code generated at level l based on level l-1, is given in formula (5) below:

$$L_l = \left(\mathop{\mathrm{O}}_{i=1}^{|L_{l-1}|} L_{l-1}^{i} \circ (l+1) \right) \circ \left(1 \circ L_{l-1}^{1} \right) \qquad (5)$$

where:
$\circ$ is a concatenation.
O is multiple concatenation
L_l^i is the i-th code of code list L_l

Examples of the code for the first 4 level are:
L_0 = < 1>
L_1 = < 2,1>
L_2 = < 23, 13, 12>
L_3 = < 234, 134, 124, 123>
L_4 = < 2345, 1345, 1245, 1235, 1234>

We now proof the following:

Theorem_1: Two consecutive elements of the list L_l in formula (5) differ only for one digit.

***Proof*:** We can proceed by induction on l:
***base*:** if $l = 0$ and $l = 1$, then the theorem is trivially true since L_0 = < 1 > and L_1 = < 2,1>;
***inductive hypothesis*:** let us suppose that L_{j-1}^{i} and L_{j-1}^{i+1} in L_{j-1} differ only for one digit, with $1 \le i \le |L_{l-1}|$, and $1 \le j \le l$-1.

***inductive step*:** the list L_l is obtained by concatenating l+1 to the word L_{l-1}^{i} for $1 \le i \le |L_{l-1}|$-1. Since every consecutive words in L_{l-1} differ only for one digit, then also in L_l. Now, we must prove that last two consecutive words in L_l also differ for one digit. The last word in L_l starts with 1, and it is concatenated with the first word of L_l, so the word would be of the form 1234…l, since the word formed start with 2 in level l = 1 and every level it is concatenated with l+1. While the last two of L_l is formed from the last word of L_{l-1} that is concatenated with l+1, so the word is 123…l-1l+1. Therefore, it is obvious that they differ only for one digit, i.e. in the last one.

Note that the new Gray code just presented, comes from reading the nodes of the generating tree of permutation n with two cycles $S_{n,2}$. Therefore, when each element in the list of the Gray code considered as multiplication of integers instead of strings and all of them summed up, we then have the number of permutation n (which correspondence with the level of the tree) of two cycles, i.e. a signless of Stirling numbers of the first kind $c(n,2)$. Therefore equation (5) can be used as an alternative to count $c(n,2)$ iteratively instead of recursively by replacing concatenation operation "o" with multiplication, and the multiple concatenation O as a a summation $\sum$. As an example the Gray code L_4 becomes: 2.3.4.5 + 1.3.4.5 + 1.2.4.5 + 1.2.3.5 + 1.2.3.4.

CONCLUSION AND FURTHER DEVELOPMENT

Reading or traversing the generating tree of combinatorial objects could generate a Gray codes. One of them presented in this paper, which the new Gray code presented is generated based on Puspitodjati's generating tree of permutation with two cycles. This Gray code could be used to count signless Stirling number of the first kind for m = 2, i.e. c $(n, 2)$, iteratively.

A further development of this research could be finding algorithms to encode or translate this new Gray code generating trees based into its combinatorial object, i.e. permutation with two cycles, since its regularity from its succession rules. Besides, the implementation of this Gray code on encryption is a challenging research as Zhou in [12] has implemented the *Generalized P-Gray Code* (GPGC) to encrypt a multimedia data.

REFERENCES

[1] C. Banderier *et al*. 2002. Generating Functions for Generating Trees. Discrete Mathematics. 246. pp. 29-55. Elsevier Science B.V. http://www.sciencedirect.com/science/article/pii/S0012365X01002503.

[2] A. Bernini *et al*. 2007. A general exhaustive generation algorithm for Gray structure. Acta Informatica. 44(5): 361-376. Springer-Verlag.

[3] E. Duchi, JM. Fedou and S. Rinaldi. 2004. From Object Grammars to ECO systems. Theoretical

ARPN Journal of Engineering and Applied Sciences

Computer Science. 314: 57-95. http://www.sciencedirect.com/science/article/pii/S0304397503005826.

[4] L. Ferrari and R. Pinzani. 2005. Catalan like numbers and succession rules. PU.M.A. 16(3): 229-250.

[5] F. Ruskey. 2003. Combinatorial Generation, http://www. 1stworks.com/ref/ RuskeyCombGen-.pdf.

[6] H. S. Wilf. 1999. East Side, West side… an introduction to combinatorial families-with Maple programming. http://www.math. upenn.edu/~wilf /eastwest.pdf.

[7] V. Vatter. 2006. Finitely Labeled Generating Trees and Restricted Permutations. Journal of Symbolic Computation. 41: 559-572.

[8] D. Merlini. 2008. Proper Generating Trees and Their Internal Path Length, Discrete Applied Mathematics. 156(5): 627-646.

[9] J. L. Baril. 2006. Gray Code for Permutation with a Fixed Number of Cycle. Discrete Mathematics. 307: 1559-1571.

[10] C. Savage. 1997. A Survey of Combinatorial Gray Codes. SIAM Review. 39: 605-629.

[11] S. Puspitodjati. 2010. Exhaustive Generation Algorithm of Permutation with Fixed Cycles and the Size of Permutation as Variable. Dissertation. Doctoral Program in Information Technology. Gunadarma University - Indonesia.

[12] Y. Zhou, K. Panetta, S. Agaian. 2009. Image encryption algorithms based on generalized P-gray code bit plane decomposition, Proceedings of the 43rd Asilomar conference on Signals, systems and computers. pp. 400-404. IEEE Press Piscataway, NJ, USA.