RESEARCH ARTICLE OPEN ACCESS

Fibonacci Coding for Lossless Data Compression – A Review

Ezhilarasu P
Associate Professor
Department of Computer Science and Engineering
Hindusthan College of Engineering and Technology,
Coimbatore - 641032,
TamilNadu - India

ABSTRACT

In this paper, Fibonacci coding data compression technique reviewed. Initially, Fibonacci encoding performed for the input with fewer amounts of symbols. Then decoding for the obtained result achieved. It regenerates the original uncompressed data. Finally, input with more number of symbols taken for compression. The average bits, compression ratio, and, space savings also calculated.

Keywords:- Fibonacci Coding, Compression, Encoding, Decoding.

I. INTRODUCTION

Data compression defined as the representation of data in such a way that, the storage area needed for target data is less than that of, the size of the input data [1]. The data compression used in plenty of data processing areas. In ASCII code, we have 256 characters represented by the different numbers. Ex. 'a' represented, numerically as 97. The frequency of each ASCII code differs from each other. If text data used as an input, some characters occur most frequently, and many characters never utilized in the input. The alphabet, digits and some special characters used mostly. In the alphabet, the most used characters are vowels. The 256 symbols never used frequently.

The variations in this frequency of characters need data compression. The fixed length code, need to be replaced by variable length code. There exist many research papers on variable length code for data compression [2, 3, 4, and 5]. Fibonacci coding generates variable length codes. It uses traditional methods of replacing input characters by specific code like code words. It uses Fibonacci series to implement the compression.

II. RELATED WORK

Apostolico and Fraenkel [1987] generated variable length code for data compression [6]. They used the concept invented by Leonardo of Pisa [1202] known as Fibonacci for mathematical calculation [7]. It translated by sigler [2002] as Fibonacci's Liber Abaci [8].

III. FIBONACCI CODING

The Fibonacci series concept developed from the calculation of rabbit reproduction [7]. The rabbit pairs, ready for reproduction at one month. Then next month it produces another pair. Therefore, the rabbit pair produces another rabbit pair after the second month. Then each month it produces a pair. If all pair survives and performs reproduction as usual then at the end of the first month (i.e. starting with the second month), we have one pair. After second, third, and fourth month, we will have one, two, three, and five pairs. In the Table I, the explanation for Fibonacci series through rabbit reproduction had given.

The numbers in the red color (entire first column) represents the month. The numbers in the blue color (whole first row) represents the number given for the rabbit pair. The numbers in the green color (whole last column) represents the total number of the pair at the starting of the month. The reproduction process represented by the brown colored number. The brown colored zero shows that the rabbit pair is ready for the reproduction. The brown colored one shows that the rabbit pair reproduced a rabbit pair (Except second row and second column one). The first pair represented in the table as the value 1 in the second row and second column. The second pair, third pair represented in the table as the value 1 in the third row and second column and fourth row and second column.

TABLE I. FIBONACCI SERIES THROUGH RABBIT REPRODUCTION

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	
1	1																																		1
2	0																																		1
3	1																																		2
4	1	0																																	3
5	1	1	0																																5
6	1	1	1	0	0																														8
7	1	1	1	1	1	0	0	0																											13
8	1	1	1	1	1	1	1	1	0	0	0	0	0																						21
9	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0														34
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	55
11	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	89

The above concept used to represent the Fibonacci series through integers. The Fibonacci series are zero, one, one, two, three, five, eight, thirteen, twenty-one, thirty-four, fifty-five, eighty-nine, ... [9].

The series derived from the equation 1.

$F_n=F_{n-1}+F_{n-2}$ (1)

Where $F_0=0$, and $F_1=1$.

The list given in the Table II.

TABLE II. FIBONACCI SERIES

F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}
0	1	1	2	3	5	8	13	21	34	55	89	144

The Fibonacci series used to create Fibonacci code. The Fibonacci code involves symbol, code word, and Fibonacci representation. The two initial values 0 and 1 neglected. The symbol 1 represented as F_2. The symbol 2, 3, 4, 5, and 6 represented as F_3, F_4, $F_2 + F_4$ (1+3), F_5, and $F_2 + F_5$ (1+5). The remaining list completed using Table II. The code word ends with 11 for all symbol representation. The length of code word for F_2, F_3, and F_4 = two, three, and four. i.e. The length of code word for $F_n = n$.

TABLE III. CODE WORD FOR SYMBOL [1-34]

Position→ Value→ Symbol	1 1	2 2	3 3	4 5	5 8	6 13	7 21	8 34	Value	Code word= Value + Suffix 1
1	1								1	11
2	0	1							01	011
3	0	0	1						001	0011
4	1	0	1						101	1011
5	0	0	0	1					0001	00011
6	1	0	0	1					1001	10011
7	0	1	0	1					0101	01011
8	0	0	0	0	1				00001	000011
9	1	0	0	0	1				10001	100011
10	0	1	0	0	1				01001	010011
11	0	0	1	0	1				00101	001011
12	1	0	1	0	1				10101	101011
13	0	0	0	0	0	1			000001	0000011
14	1	0	0	0	0	1			100001	1000011
15	0	1	0	0	0	1			010001	0100011
16	0	0	1	0	0	1			001001	0010011
17	1	0	1	0	0	1			101001	1010011
18	0	0	0	1	0	1			000101	0001011
19	1	0	0	1	0	1			100101	1001011
20	0	1	0	1	0	1			010101	0101011
21	0	0	0	0	0	0	1		0000001	00000011
22	1	0	0	0	0	0	1		1000001	10000011
23	0	1	0	0	0	0	1		0100001	01000011
24	0	0	1	0	0	0	1		0010001	00100011

25	1	0	1	0	0	0	1		1010001	10100011
26	0	0	0	1	0	0	1		0001001	00010011
27	1	0	0	1	0	0	1		1001001	10010011
28	0	1	0	1	0	0	1		0101001	01010011
29	0	0	0	0	1	0	1		0000101	00001011
30	1	0	0	0	1	0	1		1000101	10001011
31	0	1	0	0	1	0	1		0100101	01001011
32	0	0	1	0	1	0	1		0010101	00101011
33	1	0	1	0	1	0	1		1010101	10101011
34	0	0	0	0	0	0	0	1	00000001	000000011

A. FEWER AMOUNT OF CHARACTERS

If input message (M) length = 100. Where [M] = [m1, m2, ..., m8] with the occurrence [32, 21, 17, 12, 10, 5, 2, 1]. The probability of each character is as given in the Table IV. The Code Word for each character also provided in the Table V by referring Table III.

TABLE IV. CODE TABLE – CHARACTER OCCURRENCE, PROBABILITY

CHARACTER	m1	m2	m3	m4	m5	m6	m7	m8
OCCURRENCE	32	21	17	12	10	5	2	1
PROBABILITY	0.32	0.21	0.17	0.12	0.1	0.05	0.02	0.01

TABLE V. CODE TABLE – CHARACTER CODE WORD(FEWER AMOUNT)

Message	**m1**	**m2**	**m3**	**m4**	**m5**	**m6**	**m7**	**m8**
Probability	0.32	0.21	0.17	0.12	0.1	0.05	0.02	0.01
Code Word	11	011	0011	1011	00011	10011	01011	000011

The total number of bits needed = 32 *2 +21 *3 + 17*4 + 12*4 +10*5 +5*5 + 2*5 +1*6

=64+63+68+48+50+25+10+6

=334 bits

The size of the input as uncompressed = 100 * 8

= 800 bits

B. MORE AMOUNT OF CHARACTERS

INPUT

Dr.Ezhilarasu Umadevi Palani obtained his Post Graduate Degree in Computer Science and Engineering from Anna University, Chennai.

TABLE VI. CODE TABLE – CHARACTER CODE WORD(MORE AMOUNT)

S.No	Symbol	Occurrence	Probability	Code Word	Code Word Length	Code Word – Total Bits
1.	Space character " "	16	0.124031	11	2	32
2.	e	13	0.100775	011	3	39

3.	n	13	0.100775	0011	4	52
4.	i	12	0.093023	1011	4	48
5.	a	11	0.085271	00011	5	55
6.	r	8	0.062016	10011	5	40
7.	t	5	0.03876	01011	5	25
8.	d	4	0.031008	000011	6	24
9.	o	4	0.031008	100011	6	24
10.	s	4	0.031008	010011	6	24
11.	h	3	0.023256	001011	6	18
12.	m	3	0.023256	101011	6	18
13.	u	3	0.023256	0000011	7	21
14.	g	3	0.023256	1000011	7	21
15.	U	2	0.015504	0100011	7	14
16.	E	2	0.015504	0010011	7	14
17.	C	2	0.015504	1010011	7	14
18.	.	2	0.015504	0001011	7	14
19.	l	2	0.015504	1001011	7	14
20.	v	2	0.015504	0101011	7	14
21.	c	2	0.015504	00000011	8	16
22.	D	2	0.015504	10000011	8	16
23.	P	2	0.015504	01000011	8	16
24.	b	1	0.007752	00100011	8	8
25.	G	1	0.007752	10100011	8	8
26.	S	1	0.007752	00010011	8	8
27.	z	1	0.007752	10010011	8	8
28.	p	1	0.007752	01010011	8	8
29.	f	1	0.007752	00001011	8	8
30.	y	1	0.007752	10001011	8	8
31.	,	1	0.007752	01001011	8	8
32.	A	1	0.007752	00101011	8	8

The total number of bits needed = 32 + 39 + 52 + 48 + 55 + 40 + 25 + 24 + 24 + 24 + 18 + 18 + 21 + 21 + 14 + 14 + 14 + 14 + 14 + 14 + 16 + 16 + 16 + 8

+ 8 + 8 + 8 + 8+ 8 + 8 + 8 + 8

=645 bits

The size of the input as uncompressed = 129 * 8

= 1032 bits

ENCODING

The given input “Dr.Ezhilarasu Umadevi Palani obtained his Post Graduate Degree in Computer Science and Engineering from Anna University, Chennai.” after encoding will be

100000111001100010110010011100100110010111011100101100011100110001101001100000111101000111010110001
100001101101010111011110100001100011100101100011001110111110001100100011010110001110110011011000011
110010111011010011110100001110001101001101011111010001110011000110000110000011000110101101111100000
110111000011100110110111110110011111010011100011101011010100110000011010110111001111000100110000001
110110110011000000110111100011001100001111001001100111000011101100110110111001110110011100001111000

0101110011100011101011110010101100110011000111101000110011101101010110111001101001110110101110001011
0100101111110100110010110110011001100011101100010 11

DECODING

Before Decoding

1000001110011000101100100111001001100101110111001011000111001100011010011000001111010001110101100011
0000110110101011101111010000110001110010110001100111011111000110010001101011000111011001101100001111
0010111011010011110100001110001101001101011111010001110011000110000110000011000110101101111110000011
0111100000111001101101111110110011111010011100011101011010100110000011010110111001111000100110000000111
0110110011000000110111100011001100001111001001100111000011101100110110111001110110011100000111100001
0111001110001110101111001010110011001100011110100011001110110101011011100110100111011010111000101101
0010111111010011001011011001100110001110110001011

The decoding process read the input, character by character until the number 11 found. Then the entire binary code decoded to corresponding unique character.

Step 1

D100110001011001001110010011001011101110010110001110011000110100110000011110100011101011000110000110
11010101110111101000011000111001011000110011101111100011001000110101100011101100110110000111100101110
11010011110100001110001101001101011111010001110011000110000110000011000110101101111110000011011100000
111001101101111110110011111010011100011101011010100110000011010110111001111000100110000000111011011001
1000000110111100011001100001111001001100111000011101100110110111001110110011100000111100001011100111
0001110101111001010110011001100011110100011001110110101011011100110100111011010111000101101001
011111010011001011011001100110001110110001011

Step 2

Dr000101100100111001001100101110111001011000111001100011010011000001111010001110101100011000011011
0101011101111010000110001110010110001100111011111000110010001101011000111011001101100001111001011101
1010011110100001110001101001101011111010001110011000110000110000011000110101101111110000011011100000
1110011011011111101100111110100111000111010110101001100000110101101110011110001001100000001110110110011
0000001101111000110011000011110010011001110000111011001101101110011101100111000001111000010111001110
0011101011110010101100110011000111101000110011101101010110111001101001110110101110001011010010111
11010011001011011001100110001110110001011

Step 3

Dr.0010011100100110010111011100101100011100110001101001100000111101000111010110001100001101101010111
0111101000011000111001011000110011101111100011001000110101100011101100110110000111100101110110100111
1101000011100011010011010111110100011100110001100001100000110001101011011111100000110111000011100110
1101111110110011111010011100011101011010100110000011010110111001111000100110000000111011011001100000
0110111100011001100001111001001100111000011101100110110111001110110011100000111100001011100111000111
0101111001010110011001100011110100011001110110101011011100110100111011010111000101101001011111101001
10010110110011001100011101100010 11

Step 4

Dr.E10010011001011101110010110001110011000110100110000011110100011101011000110000110110101011101111
0100001100011100101100011001110111110001100100011010110001110110011011000011110010111011010011110100
0001110001101001101011111010001110011000110000110000011000110101101111110000011011100001110011011011
1110110011111010011100011101011010100110000011010110111001111000100110000000111011011001100000011011
1100011001100001111001001100111000011101100110110111001110110011100000111100001011100111000111010111
1001010110011001100011110100011001110110101011011100110100111011010111000101101001011111101001100101
1011001100110001110110001011

Step 5

Dr.Ez00101110111001011000111001100011010011000001111010001110101100011000011011010101110111101000011
0001110010110001100111011111000110010001101011000111011001101100001111001011101101001111010000111000
0011010011010111110100011100110001100001100000110001101011011111100000110111000011100110110111110110
0111110100111000111010110101001100000110101101110011110001001100000001110110110011000000110111100011
0011000011110010011001110000111011001101101110011101100111000001111000010111001110001110101111001010

11001100110001111010001100111011010101101110011010011101101011100010110100101111101001100101101100110011000111011 0001011

Step 6

Dr.Ezh1011100101100011100110001101001100000111101000111010110001100001101101010111011110100001100011100101100011001110111110001100100011010110001110110011011000011110010111011010011110100001110001101001101011111010001110011000110000110000011000110101101111100000110111000011100110110111110110011111010011100011101011010100110000011010110111001111000100110000001110110110011000000110111100011001100001111001001100111000011101100110110111001110110011100001111000010111001110001110101111001010110011001100011110100011001110110101011011100110100111011010111000101101001011111010011001011011001100110001110110001011

Step 7

Dr.Ezhi10010110001110011000110100110000011110100011101011000110000110110101011101111010000110001110010110001100111011111000110010001101011000111011001101100001111001011101101001111010000111000110100110101111110100011100110001100001100000110001101011011111000001101110000111001101101111110110011111010011100011101011010100110000011010110111001111000100110000001110110110011000000110111100011001100001111100100110011100001110110011011011100111011001110000111100001011100111000111010111100101011001100110001111010001100111011010101101110011010011101101011100010110100101111101001100101101100110011000 1110110001011

Step 8-129

Dr.Ezhilarasu Umadevi Palani obtained his Post Graduate Degree in Computer Science and Engineering from Anna University, Chennai.

IV. RESULT AND DISCUSSION

The compression ratio, space savings and average bits calculated for the fewer amount of data are

Compression ratio = 800/334
= 200:87
= 2.395:1

Space savings =1-(334/800)
= 1-(87/200)
= 1-0.435
= 0.565
= 56.5%

Average bits = 334/100
= 3.34 bits per character

The compression ratio, space savings and average bits calculated for the more amounts of data are

Compression ratio = 1032/645
= 344:215
= 1.6:1

Space savings =1-(645/1032)
= 1-(215/344)
= 1-0.625
= 0.375
= 37.5%

Average bits = 645/129
= 5.0 bits per character

V. CONCLUSION

The Fibonacci coding is a data compression technique that based on Fibonacci series. It produces static variable length code for representing the data. The Shannon-Fano coding and Huffman Coding provides dynamic variable length code. Both, the fewer amount and more amount input data produces good compression ratio, space savings, and average bits per character. But for less number of input characters with more probability, Fibonacci coding gives better compression ratio, space savings, and average bits per character.

REFERENCE

[1] P. Ezhilarasu, N. Krishnaraj, and B. Dhiyanesh, "Arithmetic Coding for Lossless Data Compression – A Review", International Journal of Computer Science Trends and Technology (IJCST) – Volume 3 Issue 3, pp. 89 – 94, 2015.

[2] C. E. Shannon, “A Mathematical Theory of Communication”, Bell System Technical Journal, Volume 27, pp. 398-403, 1948.

[3] R.M. Fano, “The transmission of information”, Technical Report 65, Research Laboratory of Electronics, M.I.T, Cambridge, Mass, 1949.

[4] D.A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes", Proceedings of the I.R.E, pp. 1098–1102, 1952.

[5] P. Elias, "Universal Codeword Sets and Representations of Integers," IEEE Trans. Inform. Theory, IT-21, pp. 194-203, 1975.

[6] A. Apostolico and A. Fraenkel, "Robust Transmission of Unbounded Strings Using Fibonacci Representations", IEEE Transactions on Information Theory, 33(2), pp. 238– 245, 1987.

[7] Leonardo of Pisa (known as Fibonacci). Liber Abaci. 1202.

[8] Sigler, L.E. (2002) Fibonacci's Liber Abaci. Springer, New York.

[9] http://en.wikipedia.org/wiki/Fibonacci_number