

Compression Rate Comparison between Elias Delta and Unary Codes

¹Dedi Purwanto, ²Andysah Putera Utama Siahaan

Faculty of Science and Technology, Universitas Pembangunan Panca Budi, Medan, Indonesia

Email: ¹dedipurwanto@pancabudi.ac.id, ²andiesiahaan@gmail.com

Abstract: Compression is significant in data storage. Existing data is in desperate need of a reduction technique so that the data can be stored efficiently. Compression is useful because it helps reduce the use of hard disk storage. Also, compression is useful for reducing transmission costs for sending data on the network. There are many methods used to do compression. This study contains a comparison of two methods of compression. The Elias Delta and Unary algorithms are the two algorithms chosen to compare in this study. Both of these algorithms have different work processes. The results of both are also very competitive in data compression. In different data, the Elias Delta algorithm is superior in storing space during compression, and the Unary algorithm is weak in compression. On the other hand, the Unary algorithm is superior in storing data, while Elias Delta algorithm loses in storing data. The result of both algorithms has different performance using different data.

Key Words: Compression, Unary, Elias Delta, Storage.

1. INTRODUCTION:

In the digital era, the need for storage is getting bigger. The thing that needs to be done is to do efficiency and streamline data [1]. The solution to do storage management so that users are more effective and cost-efficient. Usually, to anticipate storage needs, users upgrade to media storage. At this time, private users no longer need to worry about storage shortages because storage with large storage has been sold cheaply. There are several storage options such as 8GB, 16GB, 32GB, 64GB, 128GB, 256GB to tens of TB. However, the problem does not occur on the local system. Large data will slow down the system's work, especially regarding sending data. Capacity upgrades will be easier than speed upgrades. Speed is very influential with the time needed. If work time is fast, work will be smoother. Compression is the main choice for streamlining. In addition to the explanations that have been explained previously, there are other reasons why users are advised to choose to compress data. This is done to save costs. Compression is possible to get capacity reductions in files that are passively used. It will be by the needs so that the costs incurred are not excessive.

In this modern era, data is essential [2]–[10]. The compression technique is essential because of the size of the data increases. Data storage has never been optimal because of the increasing movement and data growth. It can be seen from media formats requiring a large capacity. Computer users want the best quality data with a minimum size. Looking at these problems, the solution is to maximize compression [11], which is to reduce the place used by compressed data. There are many data compression methods available today. Most of these methods can be grouped into one of two large groups, namely lossy and lossless. An example of lossy compression is the image format. This compression can reduce the file size but cannot return the file size to its original shape again. Lossless compression is the compression used to compress data, but the data can be returned so that it can be reused. Two methods can be used in lossless compression, Unary, and Elias Delta [12]. Both of this compression use the calculation process of the appearance of characters [13]. By comparing these two compression methods, it is expected to use compression which is more suitable for data that has a large number of characters.

2. THEORIES:

Data compression is the process of manipulating information using other bits or units that are lower than data representations that are not encoded with a particular encoding system [14]. Data compression is the science or art of streamlining data into a more compact form. Compression is the process of converting an original raw data into smaller compressed data. Sometimes compression is an encryption process from cryptography. Data compression is a process that can convert an original data source or data stream into compressed data streams that have a smaller size [15]. Data compression is a presentation of information in a simpler form. In computer science and information theory, data compression is the process of encoding information by using fewer bits from a source that has not been encoded through the use of a specific coding scheme. The purpose of data compression is to represent digital data with as few bits as possible, but still, maintain the minimum need to reshape the original data. This digital data can be in the form of text, images, sound, and a combination of the three, such as video. A method is needed to manipulate the data to make smaller than the original data. In the data compression algorithm, no algorithm is suitable for all types of data. It is because the data to be compressed must be analyzed first, and hopes to find a certain pattern to obtain optimal results and the smallest size [16].

Data Compression Techniques can be classified into two categories:

- Lossless Compression
- Lossy Compression

Lossless Compression has a lower degree of compression, but with data, accuracy maintained between before and after the compression process. This compression is suitable for databases, documents or spreadsheets. In lossless compression, this is not allowed to be missing bits from the data in the process compression. Lossy compression causes changes in data compared to before the compression process. Instead lossy compression provides a higher degree of compression [17]. This type is suitable for digital sound file compression and digital images. Sound and image files can still be used even if they are not in the same size as earlier. Data compression consists of two processes, modeling, and coding. The primary process of data compression is to determine a series of parts of the data to convert it to code. If the compression process is sufficient, then the results of the stream of codes will be smaller regarding size than the stream of symbols. The decision to indent certain symbols with specific codes is the core of the modeling process. In general, it can be interpreted that a model is a collection of data and rules that determine the pair between symbols as input and code as the output of the compression process while coding is the process of applying the modeling to a data compression process [18].

3. METHODOLOGY:

3.1 Elias Delta Code

The Elias Delta algorithm is a compression algorithm made by Piter Elias which replaces Elias Gamma algorithm. The following stages are the steps taken to perform the Elias Delta compression process [19].

To code a number $X \geq 1$:

- Let $N = \lfloor \log_2 X \rfloor$; be the highest power of 2 in X , so $2^N \leq X < 2^{N+1}$.
- Let $L = \lfloor \log_2 (N+1) \rfloor$ be the highest power of 2 in $N+1$, so $2^L \leq N+1 < 2^{L+1}$.
- Write L zeros, followed by
- the $L+1$ -bit binary representation of $N+1$, followed by
- all but the leading bit (i.e. the last N bits) of X .

An equivalent way to express the same process:

- Separate X into the highest power of 2 it contains (2^N) and the remaining N binary digits.
- Encode $N+1$ with Elias gamma coding.
- Append the remaining N binary digits to this representation of $N+1$.

Table 1. Elias Delta Coding structure

Number	N	N+1	δ encoding	Implied probability
$1 = 2^0$	0	1	1	1/2
$2 = 2^1 + 0$	1	2	0 1 0 0	1/16
$3 = 2^1 + 1$	1	2	0 1 0 1	1/16
$4 = 2^2 + 0$	2	3	0 1 1 0 0	1/32
$5 = 2^2 + 1$	2	3	0 1 1 0 1	1/32
$6 = 2^2 + 2$	2	3	0 1 1 1 0	1/32
$7 = 2^2 + 3$	2	3	0 1 1 1 1	1/32
$8 = 2^3 + 0$	3	4	0 0 1 0 0 0 0 0	1/256
$9 = 2^3 + 1$	3	4	0 0 1 0 0 0 0 1	1/256
$10 = 2^3 + 2$	3	4	0 0 1 0 0 0 1 0	1/256

$11 = 2^3 + 3$	3	4	00 1 00 011	1/256
$12 = 2^3 + 4$	3	4	00 1 00 100	1/256
$13 = 2^3 + 5$	3	4	00 1 00 101	1/256
$14 = 2^3 + 6$	3	4	00 1 00 110	1/256
$15 = 2^3 + 7$	3	4	00 1 00 111	1/256
$16 = 2^4 + 0$	4	5	00 1 01 0000	1/512
$17 = 2^4 + 1$	4	5	00 1 01 0001	1/512

To decode Elias Delta back into the original [19]:

- Read and count zeros from the stream until you reach the first one. Call this count of zeros L.
- Considering the one that was reached to be the first digit of an integer, with a value of $2L$, read the remaining L digits of the integer. Call this integer $N+1$, and subtract one to get N.
- Put a one in the first place of our final output, representing the value $2N$.
- Read and append the following N digits.

2.2 Unary Code

This method is commonly used in data compression and is widely used with a combination of several modification techniques. Unary Coding is represented in a string of n bits 1 followed by one bit 0 ending which is defined as n-1 bit 1 followed by one bit 0. Alternatively, vice versa as an alternative can also be equivalent to starting from n bit 0 followed by bit 1 ending which is defined as n-1 bit 0 followed by one bit 1. In the Unary Coding method, there is no frequency distribution of symbols on a string [20].

Table 2. Unary Coding structure

n (non-negative)	n (strictly positive)	Unary code	Alternative
0	1	0	1
1	2	10	01
2	3	110	001
3	4	1110	0001
4	5	11110	00001
5	6	111110	000001
6	7	1111110	0000001
7	8	11111110	00000001
8	9	111111110	000000001
9	10	1111111110	0000000001

4. RESULT AND DISCUSSION:

This section is testing both algorithms. Each algorithm will process the compression of the given text. The following is an illustration of the two algorithms for data compression.

Text	=	In the digital era, the need for storage is getting bigger.
Length	=	59
Character Set	=	In the digital era, the need for storage is getting bigger.
CS Length	=	18
Bit	=	472
Padding	=	5

Table 3. Unary Code result

Compressed Text = (T...\$McUäôSÉjB'B)lϠŠFç±-\$±Z@ÉE-ḃ8¥l
 Compressed Bit = 304
 Compressed Char = 38
 Ratio = $\frac{38}{59} * 100\%$
 = 64.41%
 Saved Space = 35.59%

Previous experiments explained that Elias Delta was superior to about 10% of Unary. However, Elias Delta or Unary is not always better regarding compression. All of this depends on the text provided and the frequency of the appearance of letters. Following is the second experiment on these two algorithms.

Text = The Quick Brown Fox Jumps Over The Lazy Dog
 Length = 43
 Character Set = The QuickBrownFxBjumpsOvLazyDg
 CS Length = 28
 Bit = 344
 Padding = 3

Table 5. Unary Code result

Char	Freq	Code
	8	0
e	3	10
o	3	110
h	2	1110
u	2	11110
r	2	111110
T	2	1111110
c	1	11111110
k	1	111111110
B	1	1111111110
Q	1	11111111110
i	1	111111111110
w	1	1111111111110
n	1	11111111111110
F	1	111111111111110
x	1	1111111111111110
J	1	111111111111111100
m	1	1111111111111111000
p	1	11111111111111110000
s	1	111111111111111100000
O	1	1111111111111111000000
v	1	111111111111111111110
L	1	11111111111111111111110
a	1	111111111111111111111110
z	1	1111111111111111111111110
y	1	11111111111111111111111110
D	1	111111111111111111111111110
g	1	1111111111111111111111111110

Compressed Text = yÓy{yììÏúiyÏy·yóyyiyúyyÏ yóyyßyy_?tyyyýýýýýýýýýýÏyyúÏ yy=3
 Compressed Bit = 456
 Compressed Char = 57
 Ratio = $\frac{57}{43} * 100\%$
 = 132.56%
 Saved Space = -32.56%

Table 6. Elias Delta Code result

Char	Freq	Code
	8	1
e	3	100
o	3	101
h	2	1100
u	2	1101
r	2	1110
T	2	1111
c	1	00100000
k	1	00100001
B	1	00100010
Q	1	00100011
i	1	00100100
w	1	00100101
n	1	00100110
F	1	00100111
x	1	001010000
J	1	001010001
m	1	001010010
p	1	001010011
s	1	001010100
O	1	001010101
v	1	001010110
L	1	001010111
a	1	001011000
z	1	001011001
y	1	001011010
D	1	001011011
g	1	001011100

Compressed Text = {_FòB__ÏÏN¥ E¥"ÏRU+#_bJâÏÉj[Rç3
 Compressed Bit = 264
 Compressed Char = 33
 Ratio = $\frac{33}{43} * 100\%$
 = 76.74%
 Saved Space = 23.26%

The last experiment was inversely proportional to the previous experiment. Unary algorithm test results do not reduce data but increase up to 32.56%. This happens because the characters that appear are not repetitive. Meanwhile, the results from Elias Delta can compress up to 23.26%.

5. CONCLUSION:

The results of text compression testing with recurring characters will affect Unary's performance. The more repetitive characters, the higher the level of compression. In the text that has a bit of character repetition, Elias Delta algorithm will be superior to the Unary algorithm. The fewer recurring characters, the more use of bits in the Unary algorithm. From previous experiments, it can be concluded that the performance of the compression algorithm works depending on the contents of the message to be compressed. No one thing determines the Elias Delta algorithm be better than Unary or vice versa. Both of these algorithms have different ways of working and different results.

REFERENCES:

1. Suherman and A. P. U. Siahaan, "Huffman Text Compression Technique," *Int. J. Comput. Sci. Engineer ring*, vol. 3, no. 8, pp. 103–108, 2016.
2. Hariyanto and A. P. U. Siahaan, "Intrusion Detection System in Network Forensic Analysis and," *IOSR J. Comput. Eng.*, vol. 18, no. 6, pp. 115–121, 2016.
3. Khairul, M. Simaremare, and A. P. U. Siahaan, "Decision Support System in Selecting The Appropriate Laptop Using Simple Additive Weighting," *Int. J. Recent Trends Eng. Res.*, vol. 2, no. 12, pp. 215–222, 2016.
4. M. D. L. Siahaan and A. P. U. Siahaan, "Fingerprint Pattern Recoqnition Using LVQ," *IOSR J. Comput. Eng.*, vol. 18, no. 6, pp. 85–92, 2016.
5. L. Marlina, Muslim, and A. P. U. Siahaan, "Data Mining Classification Comparison (Naïve Bayes and C4.5 Algorithms)," *Inte rnational J. Eng. Trends Technol.*, vol. 38, no. 7, pp. 380–383, 2016.
6. W. Fitriani, R. Rahim, B. Oktaviana, and A. P. U. Siahaan, "Vernam Encrypted Text in End of File Hiding Steganography Technique," *Int. J. Recent Trends Eng. Res.*, vol. 3, no. 7, pp. 214–219, Jul. 2017.
7. S. Hartanto, M. Furqan, A. P. U. Siahaan, and W. Fitriani, "Haversine Method in Looking for the Nearest Masjid," *Int. J. Recent Trends Eng. Res.*, vol. 3, no. 8, pp. 187–195, Aug. 2017.
8. A. Lubis and A. P. U. Siahaan, "Network Forensic Application in General Cases," *IOSR J. Comput. Eng.*, vol. 18, no. 6, pp. 41–44, 2016.
9. S. Ramadhani, Y. M. Saragih, R. Rahim, and A. P. U. Siahaan, "Post-Genesis Digital Forensics Investigation," *Int. J. Sci. Res. Sci. Technol.*, vol. 3, no. 6, pp. 164–166, 2017.
10. S. Haryati, A. Ikhwan, D. Arisandi, Fadlina, and A. P. U. Siahaan, "Quality Assurance in Knowledge Data Warehouse," *Int. J. Sci. Res. Sci. Technol.*, vol. 3, no. 6, p. 239–242], 2017.
11. X. Lian, Z. Liu, W. Zhou, and Z. Duan, "Lossless Frame Memory Compression Using Pixel-Grain Prediction and Dynamic Order Entropy Coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 223–235, Jan. 2016.
12. D. A. Yansyah, "Perbandingan Metode Punctured Elias Code dan Huffman Pada Kompresi File Text," *J. Ris. Komput.*, vol. 2, no. 6, pp. 33–36, 2015.
13. L. Marlina, A. P. U. Siahaan, H. Kurniawan, and I. Sulistianingsih, "Data Compression Using Elias Delta Code," *Int. J. Recent Trends Eng. Res.*, vol. 3, no. 8, pp. 210–217, Aug. 2017.
14. M. Omari and S. Yaichi, "Image compression based on genetic algorithm optimization," in *2015 2nd World Symposium on Web Applications and Networking (WSWAN)*, 2015, pp. 1–5.
15. S. E. Creasey, M. F. Lynch, and J. Howard Petrie, "Compression of Bibliographic Data Bases Using a Variable to Fixed-length Bit-string Coding Transformation," *Program*, vol. 8, no. 4, pp. 191–195, Apr. 1974.
16. A. Fauzi, "Kompresi Data," 2012. [Online]. Available: <http://ajiefauzie.blogspot.com/2012/04/kompresi-data-kompresi-data-adalah.html>.
17. M. Johnson, J. Partan, and T. Hurst, "Low complexity lossless compression of underwater sound recordings," *J. Acoust. Soc. Am.*, vol. 133, no. 3, pp. 1387–1398, Mar. 2013.
18. H. Grailu, M. Lotfizad, and H. Sadoghi-Yazdi, "An improved pattern matching technique for lossy/lossless compression of binary printed Farsi and Arabic textual images," *Int. J. Intell. Comput. Cybern.*, vol. 2, no. 1, pp. 120–147, Mar. 2009.
19. Wikipedia, "Elias Delta Coding," 2018. [Online]. Available: https://en.wikipedia.org/wiki/Elias_delta_coding. [Accessed: 01-Oct-2018].
20. Wikipedia, "Unary Coding," 2017. [Online]. Available: https://en.wikipedia.org/wiki/Unary_coding. [Accessed: 01-Oct-2018].