Chapter 3

DNA SEQUENCE COMPRESSION USING RP/GP^2 METHOD WITH INFORMATION STORAGE AND SECURITY

Abstract

In the field of bioinformatics, the storing and transmission of Deoxyribonucleic Acid (DNA) is very important with respect to compression rate, ratio and encryption point of view. The DNA sequence requires huge space for storing & time for encryption causing considerable delay transmission of information. The most of the users demand low storage and computational cost, therefore efficient algorithm is required for compression. Proposed algorithm will be based on combinations of Reverse & Palindrome (RP) technique or Genetic Palindrome & Palindrome (GP^2) technique substring substitution. The variable substring will be replaced by the corresponding American Standard Code for Information Interchange (ASCII) code which is extracted of RP/ GP^2. The transmission of DNA/RNA/Protein sequence over wireless communication is a common issue . The size of DNA sequence is growing twice in a year proportionally. As a result, security of these data from unauthorized user is essential. The proposed selection encryption reduces the computational recourses for this data. The DNA sequence security is ensured by signature which depends on ASCII code & dynamic library file acting as a key. This approach shows that 95% of original file is modified when 44% to 45% is encrypted. The experimental result shows that the compression rate is 3.7750 bits/base.

Keywords: DNA sequence, Compression, Reverse, Palindrome, Genetic Palindrome, Security, Compression rate & ratio and encryption ratio.

1 Introduction

A DNA sequence is composed of four chemical compounds, defined by the alphabets A,T,G & C, called bases. Each base requires one byte for storage [30]. If the marketable standard compression algorithms are applied directly on DNA sequences, the file size is increased more than one byte per base, because DNA sequences are non-random. The DNA sequences produce amino acid and protein and for this, the DNA sequence has many repetitions as seen in higher eukaryotes [8]. The DNA sequences are highly compressible. In a long DNA sequence, the searching of exact RP / GP^2 is not an easy task and the process takes optimum time for compression. The process is divided into two modules, first to find out all the exact matching substring of Repeat & Reverse or Genetic palindrome & Palindrome and second to encode exact match regions & non match regions. Hence, the string is divided into a number of substring [63]. Shift rule [16] is mainly used as the basic principal of this technique. The

compression technique reduces the file size and I/O overhead considerably, also reduces the time of execution & as well as communication time from sender to receiver. In the last three decades [64-68] many researchers have studied in the field of DNA compression. The DNA sequences in highly non-random and has minimum complexity. In selective encryption process, the DNA sequence, also called plain text is encrypted by a group of nucleotide where match found, this is the highest priority in the process of compression. This process reduces the redundancy of the plaintext. Here the information is concentrated into smaller regions, therefore the selective encryption process is user friendly. The sequence is encrypted only where maximum matching is found, getting higher compression rate & ratio and higher effect on reformation of sequences through decompression. If no pattern match is found in the sequence, it has no effect in compression process as well as decompression process. If decrypt the compressed sequence without proper ASCII code corresponding nucleotide or library file the sequence will be different.

The aim of this work is to apply effectively compression as well as encryption. The basic principle of our technique is to group the nucleotide of the DNA sequence into a symbol. This helps the process of compression as well as encryption at the same time. This process reduces the time complexity of compression followed by encryption as well as decompression followed by decryption. The library files are acting as a key by which the DNA sequences are privately encrypted & these sequences are transmitted over the communication channel. These keys are known only those who encrypt the DNA sequences. This group of nucleotide act as a key or library file is not known to any authorized person, the authorize person who tries to decrypt the sequence by the process is known as cryptanalysis. The size of DNA database increases twice or thrice annually, so handling this data is a big problem, also in network environment. In the field of computer science, the main challenge is to reduce the DNA sequence storage space. Our developed algorithm has optimized the space requirement. Fast transmission without the loss of data is the demand of modern age of information & communication technology (ICT).

In day to day life the user search good compression algorithm with respect to data size & type and reduce I/O overhead including security aspect. All of the DNA sequences have some special function & structure [34] and important feature is repetitive in nature.

The marketable standard compression algorithm fails to compress genomic data. The genomic identity is hampered in dictionary-based compression scheme [31] where decompression is followed by ordinary symbol instead of genome sequences.

This Reverse & Palindrome (RP) or Genetic Palindrome & Palindrome (GP^2) technique is an efficient tool to compress DNA sequences. The DNA repetitive nature of exact Reverse & Palindrome (RP) or Genetic Palindrome & Palindrome (GP^2) are considered in this method. Our process overcomes the shortcoming of marketable standard algorithms, which increases the file size,

The first designed two lossless compression algorithms are Biocompress and Biocompress-2 [12,55] on the basis of Ziv and Lempel methods [69]. The arithmetic coding is developed by Boyer-Moore algorithm [50,48,7] on the basis of small pattern matching.The lossless GenCompressed-I algorithm [70-71] was developed on the basis of repeated sub-sequence operation. The modified algorithm GenCompressed-2 was based on edit operation.

2 Methods

This algorithm is based on exact matching sub-string of reverse & palindrome technique or Genetic palindrome & Palindrome technique. This process is first executed by scanning the DNA sequence from left to right. Next exact match sub-string of Reverse & Palindrome (RP) or Genetic Palindrome & Palindrome (GP^2) is detected. In the end match substring is encoded by a ASCII code in the match position. If no sub-string is found in Reverse & Palindrome (RP) or Genetic Palindrome & Palindrome (GP^2), then the unmatched character is sent to the output file.

2.1 Mathematical formulation

For a string S, one part is m_r & the other part is m_p has been compressed. Where m_r represents Reverse or Genetic Palindrome substring and m_p represents palindrome substring. If substrings m_r & m_p are exactly matched with S, then ASCII code has been placed in match position and the revised string will be S= $m_r m_p$. The algorithm finds an optimal postfix of $m_r m_p$ in descending order $m_r m_p$ that can be encoded economically. After outputting the encoding of $m_r m_p$, remove from S and append to string. This scanning process from left to write is continued again and again until an unless only one to two characters is left. This working process is shown in fig.-1

Fig.3.1 working process

2.2 Procedure of reverse &palindrome or Genetic palindrome & Palindrome searching process

tgctg**ccgtatcgt**at**ggc**atcagtcgt**tat**……n

tgc reverse is *cgt*, tat palindrome is *tat* and *ccg* Genetic Palindrome is *ggc*. An exact RP / GP^2 is a substring of m_r (reverse and palindrome) & m_p (Genetic palindrome and palindrome) that can be formed another string O (compressed file) by edit operation (RP / GP^2, insertion). Only exact matched RP/GP^2 substring are to be found out, this is the main aim of our technique of compression where sequence length is l.

2.2.1 Searching of exact repetitions of sub string in Reverse & Palindrome or Genetic Palindrome & Palindrome

The process of compression is described as follows

a. Start the program and find out all exact Reverse & Palindrome (m_r) / Genetic Palindrome & Palindrome (m_p) match on the string, store it by descending scores into an intermediate list s;

b. Extract all repeats of Reverse & Palindrome (m_r) / Genetic Palindrome & Palindrome (m_p) with highest scored from list s, then replace all Reverse & Palindrome (m_r) / Genetic Palindrome & Palindrome (m_p) by corresponding ASCII code into another list s and place Reverse & Palindrome (m_r) / Genetic Palindrome & Palindrome (m_p) to library file.

c. This Process is continued until all repeats store in intermediate list s so that there's no overlap with the evaluated all repeated Reverse & Palindrome (m_r) / Genetic Palindrome & Palindrome (m_p);

d. Repeat step 2 for finding next highest score;

e. other wise exit.

Example :

Consider S= **tgc**tg**ccgtatcgt**at**ggc**atcagtcgt**tat** ……..n

tgc reverse sub-string is *cgt* repeated in two places (shown by red color), tat palindrome is *tat* is repeated in two places (shown in green color) and *ccg* genetic palindrome is *ggc* is repeat

in one place (shown in purple color) continue this process. The sub string is replaced by the ASCII code on the basis of highest match score of reverse & palindrome / genetic palindrome & palindrome simultaneously. Then match position of i^{th}, where i^{th} is the reverse substring position and j^{th}, where j^{th} is the palindrome sub string match position, is replaced by an ASCII equivalent symbol.

s= &tgccgtat&atggcatcagt&tat { s is intermediate encoding step}, continue this process

O= &tgccg#&atggcatcagt&# [compressed output file is O byte]

2.3 Time & space complexity

The time complexity is $O(n^2)$, where n is the total number of nucleotide base in the file.

The space complexity is O(n), where n is the total number of nucleotide base in the file.

2.4 Process of compression

The process of compression & encryption is shown in the fig. 2

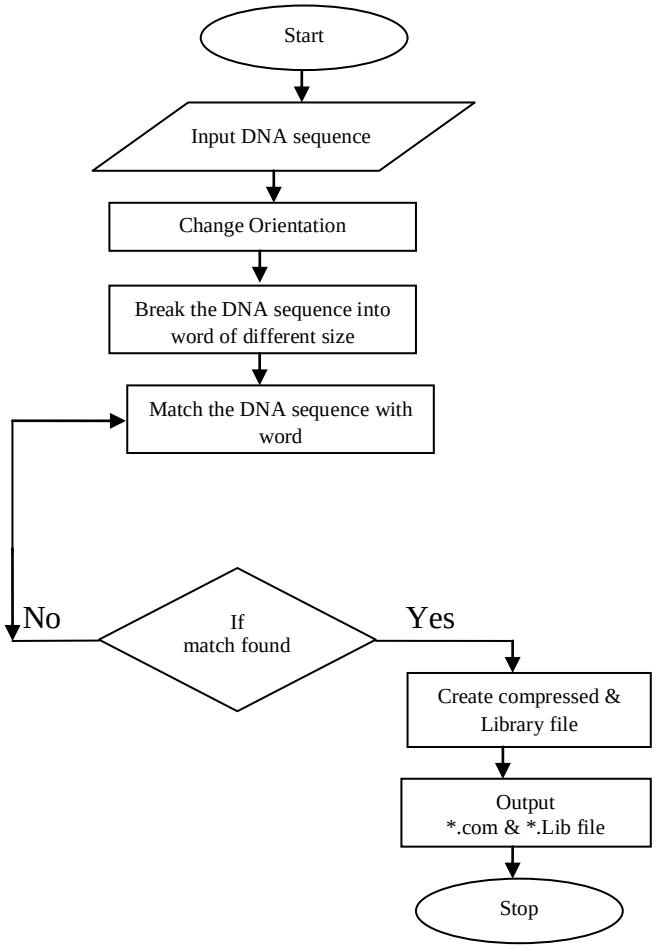

Fig.3.2 data flow diagram for compression Techniques

2.5 Encoding & Decoding Algorithm

Details algorithms of compression and encryption have been described as follows

DNA sequence encoding process using Reverse & Palindrome / Genetic Palindrome & Palindrome

INITIALIZATION OF INPUTS:
i. DNA sequence & Artificial sequence in text format of length l byte
ii. Word size of length l

ESTIMATED OUTPUT :
i. Compressed file FCOM and Library file FLIB

START
i. Define ASCII code start position
ii. Word size 1 to <10 and count
iii. Product word of Reverse & genetic palindrome string
iv. Match word with the DNA sequence
v. Request to store output in two separate file

ITERATION
1. For (break the sequence into word from 1,2,3…n)
flib[i]='l'; flib[i+1]='i';……. flib[i+n]=NULL;
fcom[i+1]='o';fcom[i+3]=fname[i];
fcom[i+n]=NULL; // end of file

2. Create an empty file TEMP.
max=0;
mw[0]=NULL;[mw=MWORD]
for(read the character a, t, g & c)
if(a[i]!='a' && a[i]!='t' && a[i]!='g' && a[i]!='c')
Check whether it exist in TEMP or not. If it exist
go to
step 3 else go to step 4.
End if

3. If it is end of file go to step 6 else go to step 2.

4. a[l]=NULL;// store sub-sequence
for(create temporary file for string compares')
for(compare Reverse & Palindrome / Genetic
Palindrome & Palindrome of the file)
if(match found)
for(store in ascending order)

5. ch++; \\ increase character position
if(ch=='a'|| ch=='t' || ch=='g' || ch=='c'||ch==32) \\ matching character

6. if(match found)
ch1++; ch=54; \\ place ASCII character economically

7. If CH1==32 append to FLIB CH and MWORD else

append to FLIB CH1 and CH and
MWORD in order.

8. Replace every word in FNAME which matches MWORD with the corresponding ASCII character. Store it in FCOM.

9. Replace the content of FNAME with FCOM and Remove FNAME and TEMP.

10. End

DNA sequence decoding process using Reverse & Palindrome / Genetic Palindrome & Palindrome

INITIALIZATION OF INPUTS:
i. Enter the text file of fcom and flib

ESTIMATED OUTPUT :
i. Exact original sequence

START
iv. Replace ASCII code by sub sequence

ITERATION
1. for(check library file size)
flib[i]=fcom[i];flib[i]='l';flib[i+1]='i';
……..flib[i+n]=NULL;
for(match library file size with sub sequence size)
fname[i]=fcom[i];fname[i]='.';
fname[i+1]='t';fname[i+2]='x';
fname[i+3]='t';fname[i+4]=NULL;

2. Read the compressed file FCOM character by character.

3. if(ASCII code match);
fput (sub sequence)
for(compare ASCII character and sub sequence);

4. Do step 2 to 3 until end of file is reached.

5. Remove FCOM and FLIB.

6. FNAME holds the original decompressed file.

7. End

3 Results & Discussion

The benchmark [55] & artificial data are used for experimental purpose. The artificial data is generated by random string generating function. The compression rate and ratio are presented in table 1 for RP & table 2 for GP^2. The encryption ratio and entropy result is presented in table 3.

Table 3.1 Cellular DNA & artificial sequence compression ratio and rate using Reverse & Palindrome Technique.

Sequence Size	Sequence Name	Base pair/ File size	Cellular DNA Sequences								Artificial Sequence	
			Original Sequences		Reverse Sequences		Complement Sequences		Reverse Complement Sequences			
			Compression ratio	Compression rate (bits /base)	Compression ratio	Compression rate (bits /base)	Compression ratio	Compression rate (bits /base)	Compression ratio	Compression rate (bits /base)	Compression ratio	Compression rate (bits /base)
Sub string Size 3	atatsgs	9647	-1.108973	4.186175	-1.106545	4.217686	-1.09967	4.17954	-1.11917	4.1953	-1.10957	4.23302
	atef1a23	6022	-1.148456	4.29691	-1.128861	4.257722	-1.13915	4.27831	-1.14148	4.28296	-1.13882	4.27765
	atrdnaf	10014	-1.054324	4.108646	-1.054723	4.109445	-1.05672	4.11344	-1.05792	4.11584	-1.08508	4.17016
	atrdnai	5287	-1.059391	4.118781	-1.05296	4.105922	-1.04274	4.08549	-1.03707	4.07414	-1.1585	4.317
	celk07e12	58949	-0.977574	3.955147	-0.976013	3.952027	-0.97276	3.94551	-0.97764	3.95528	-1.05527	4.11054
	hsg6pdgen	52173	-1.047112	4.094226	-1.050256	4.100511	-1.04382	4.08763	-1.04704	4.09407	-1.06735	4.13471
	mmzp3g	10833	-0.972491	3.944981	-0.967691	3.93538	-0.97212	3.94425	-0.96806	3.93612	-1.09286	4.18573
	xlxfg512	19338	-0.913124	3.826126	-0.922019	3.844038	-0.91292	3.82584	-0.92202	3.84404	-1.07571	4.15141
	Avg.com. rate			4.06637		4.06534		4.0575		4.06222		4.1975
	Saving perce.			54.65		54.75		54.95%		54.65		54.35
Sub string Size 4	atatsgs	9647	-0.93898	3.86276	-0.92928	3.85032	-0.93832	3.86193	-0.92928	3.85032	-1.15	4.31388
	atef1a23	6022	-1.0093	4.0186	-1.00199	4.00399	-1.00864	4.01727	-1.00199	4.00399	-1.03985	4.08074
	atrdnaf	10014	-0.95287	3.90573	-0.93928	3.87857	-0.95287	3.90573	-0.93929	3.87857	-0.97519	3.95039
	atrdnai	5287	-1.05788	4.11576	-1.01021	4.02043	-1.05712	4.11424	-1.01021	4.02043	-1.07637	4.15273
	celk07e12	58949	-0.7679	3.5358	-0.76302	3.52603	-0.77312	3.54625	-0.76169	3.52338	-0.84085	3.68171
	hsg6pdgen	52173	-0.758	3.516	-0.77801	3.55602	-0.76398	3.52796	-0.76455	3.52911	-0.80097	3.60195
	mmzp3g	10833	-0.89126	3.78252	-0.91304	3.82609	-0.89089	3.78178	-0.91341	3.82683	-0.93994	3.87989
	xlxfg512	19338	-0.79729	3.59458	-0.77888	3.55776	-0.79708	3.59417	-0.78405	3.56811	-0.89662	3.79323
	Avg. com. rate			3.79147		3.7774		3.79366		3.7750		3.9318
	Saving perce.			59.55		59.77		59.65%		59.84		59.46

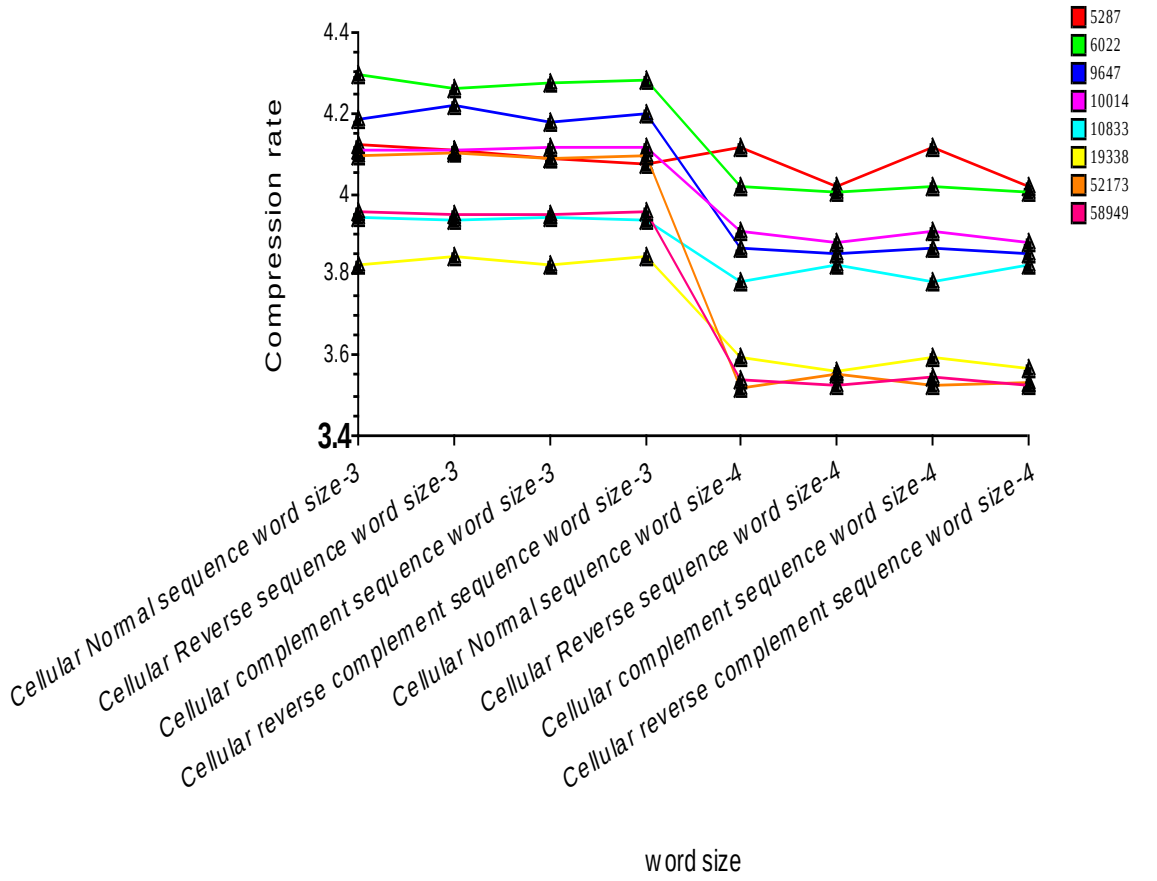

Fig. 3.3 compression rate versus different word size among original, reverse, complement, reverse complement sequence based on reverse with palindrome technique of different file size

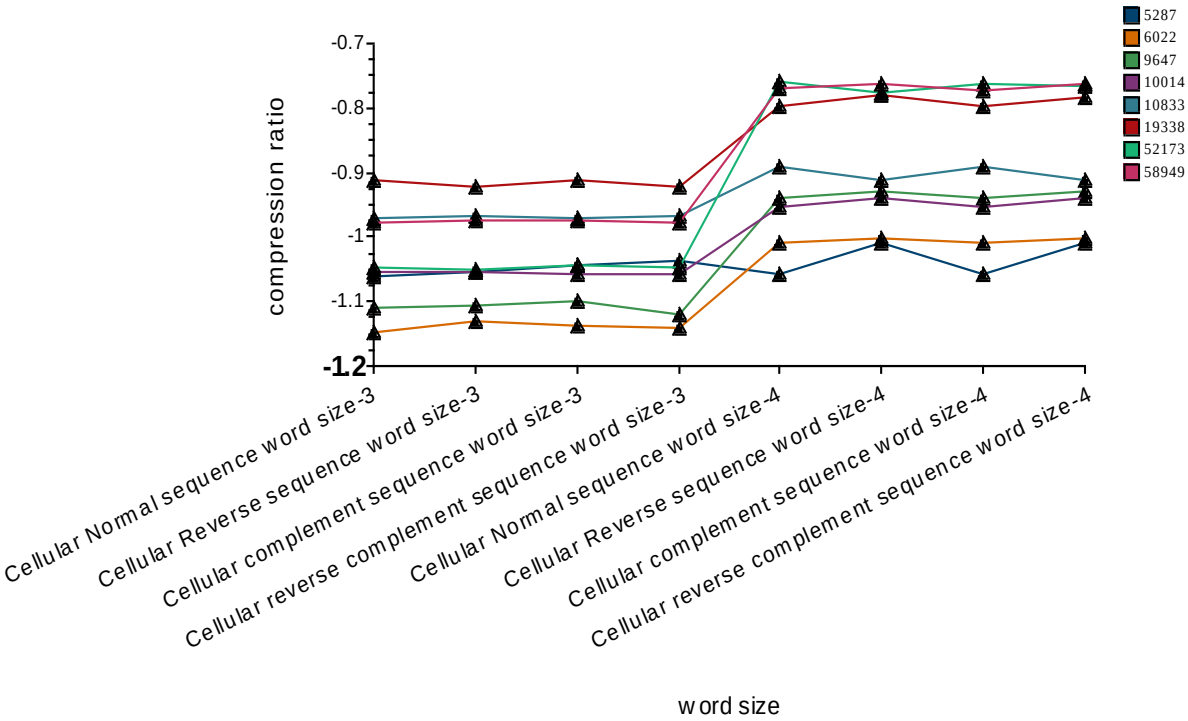

Fig. 3.4 compression ratio versus different word size among original, reverse, complement, reverse complement sequence based on reverse with palindrome technique of different file size

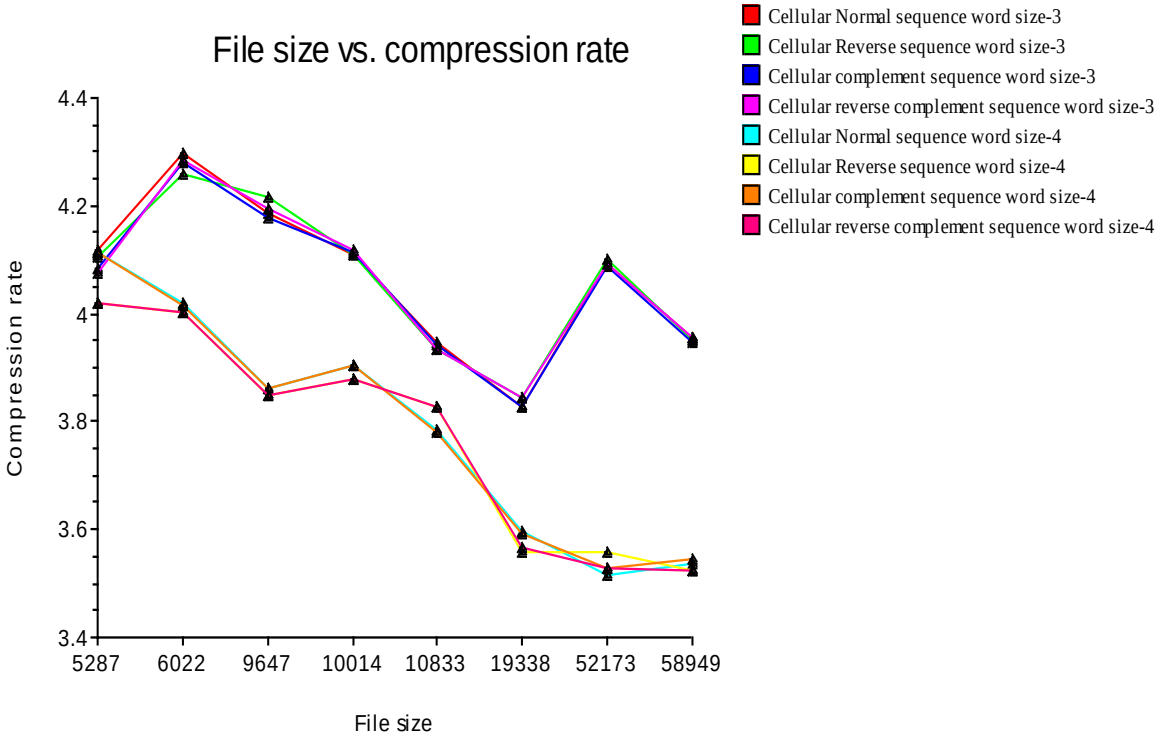

Fig. 3.5 compression rate versus file size among original, reverse, complement, reverse complement sequence based on reverse with palindrome technique of different word size.

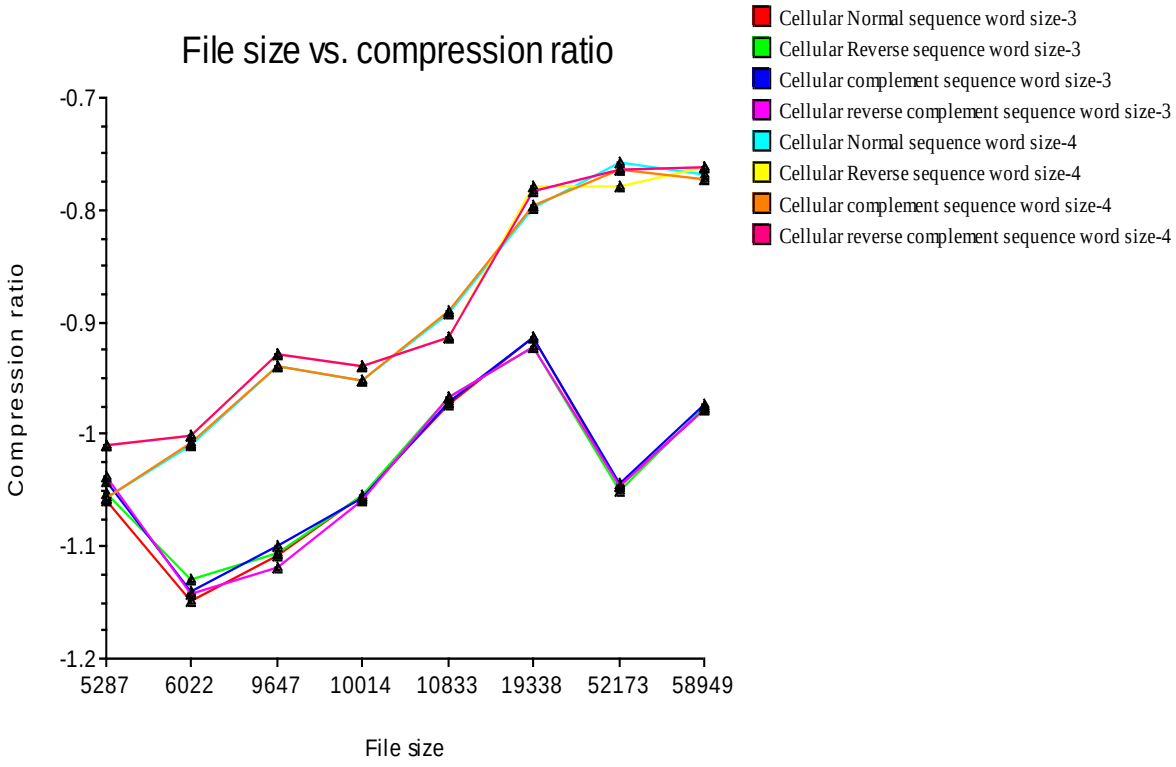

Fig. 3.6 compression ratio versus file size among original, reverse, complement, reverse complement sequence based on reverse with palindrome technique of different word size.

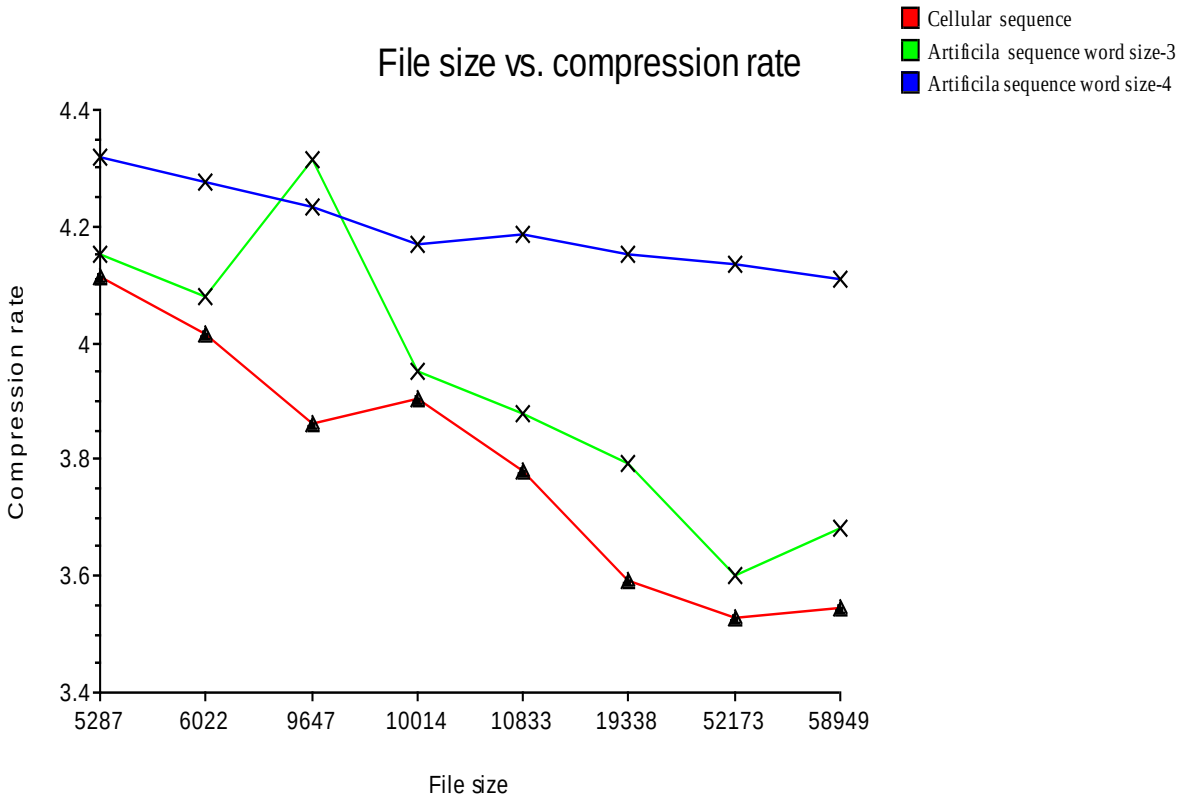

Fig. 3.7 compression rate versus file size of cellular and artificial sequence based on reverse with palindrome technique of different word size.

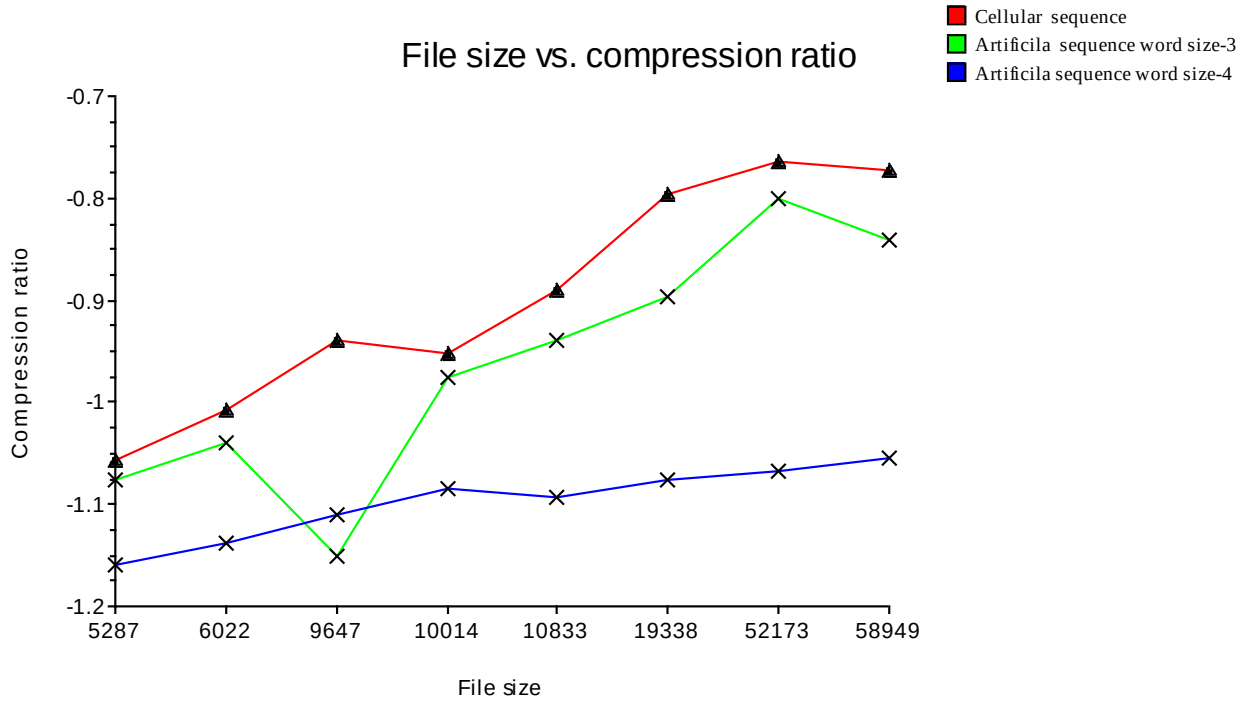

Fig. 3.8 compression ratio versus file size of cellular and artificial sequence based on reverse with palindrome technique of different word size.

Table 3.2 Cellular DNA & artificial sequence compression ratio, rate and saving percentage using Genetic Palindrome with Palindrome Technique.

Sequence Size	Sequence Name	Base pair/ File size	Cellular DNA Sequences								Artificial Sequence	
			Original Sequences		Reverse Sequences		Complement Sequences		Reverse Complement Sequences			
			Compression ratio	Compression rate (bits /base)	Compression ratio	Compression rate (bits /base)	Compression ratio	Compression rate (bits /base)	Compression ratio	Compression rate (bits /base)	Compression ratio	Compression rate (bits /base)
Sub string Size 3	atatsgs	9647	-1.09039	4.18078	-1.09309	4.18617	-1.09039	4.18078	-1.09309	4.18617	-1.09122	4.18244
	atef1a23	6022	-1.14314	4.28628	-1.14613	4.29226	-1.14314	4.28628	-1.14613	4.29226	-1.15609	4.31218
	atrdnaf	10014	-1.05113	4.10226	-1.06151	4.12303	-1.05113	4.10226	-1.06151	4.12303	-1.07949	4.15897
	atrdnai	5287	-1.08474	4.16947	-1.07831	4.15661	-1.08474	4.16947	-1.16909	4.33819	-1.18044	4.36088
	celk07e12	58949	-0.97086	3.94171	-0.9693	3.93859	-0.97086	3.94171	-0.9693	3.93859	-1.01042	4.02083
	hsg6pdgen	52173	-1.01	4.02001	-1.01192	4.02384	-1.01	4.02001	-1.01192	4.02384	-1.02825	4.05650
	mmzp3g	10833	-1.06	4.12	-1.0552	4.1104	-1.06	4.12	-1.0552	4.1104	-1.0877	4.17539
	xlxfg512	19338	-0.98263	3.96525	-0.99111	3.98221	-0.98262	3.96525	-1.01427	4.02855	-1.0424	4.08480
	Avg. com. rate		4.09822		4.10164		4.09822		4.13013		4.1690	
	Saving perce.		51.09		50.21		51.09		50.32		49.56	
Sub string Size 4	atatsgs	9647	-1.08749	4.1749	-1.08044	4.16088	-1.08749	4.17498	-1.08044	4.16088	-1.15901	4.31803
	atef1a23	6022	-1.23281	4.4656	-1.23148	4.46297	-1.23281	4.46563	-1.23148	4.46297	-1.29492	4.58984
	atrdnaf	10014	-1.11843	4.2368	-1.11344	4.22688	-1.11843	4.23687	-1.11344	4.22688	-1.14795	4.29599
	atrdnai	5287	-1.26972	4.5394	-1.25345	4.5069	-1.26972	4.53944	-1.25345	4.5069	-1.36353	4.72707
	celk07e12	58949	-0.81449	3.6289	-0.82049	3.64098	-0.81119	3.62239	-0.80824	3.61648	-0.85537	3.71073
	hsg6pdgen	52173	-0.82152	3.6430	-0.82527	3.65055	-0.82508	3.65016	-0.82527	3.65055	-0.85636	3.71273
	mmzp3g	10833	-1.05003	4.1000	-1.06683	4.13367	-1.05003	4.10007	-1.06683	4.13367	-1.14068	4.28136
	xlxfg512	19338	-0.88541	3.7708	-0.88448	3.76895	-0.88541	3.77081	-0.88179	3.76357	-0.99473	3.98945
	Avg. com.rate		4.06997		4.06897		4.07004		4.06524		4.2031	
	Saving perce.		54.73		54.82		53.10		54.98		52.20	

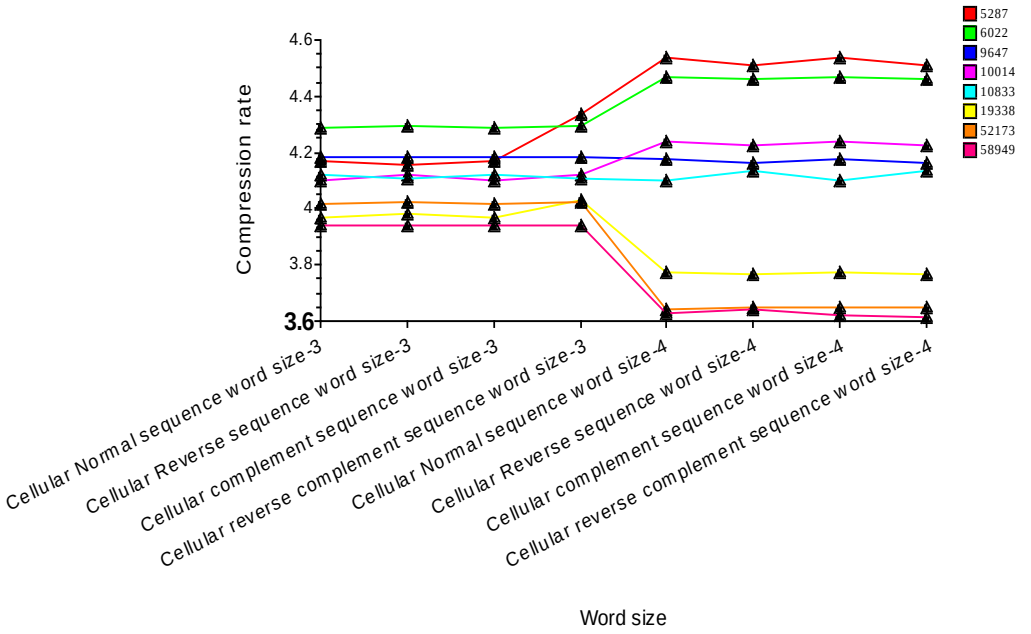

Fig. 3.9 compression rate versus different word size among original, reverse, complement, reverse complement sequence based on Genetic palindrome and palindrome technique of different file size

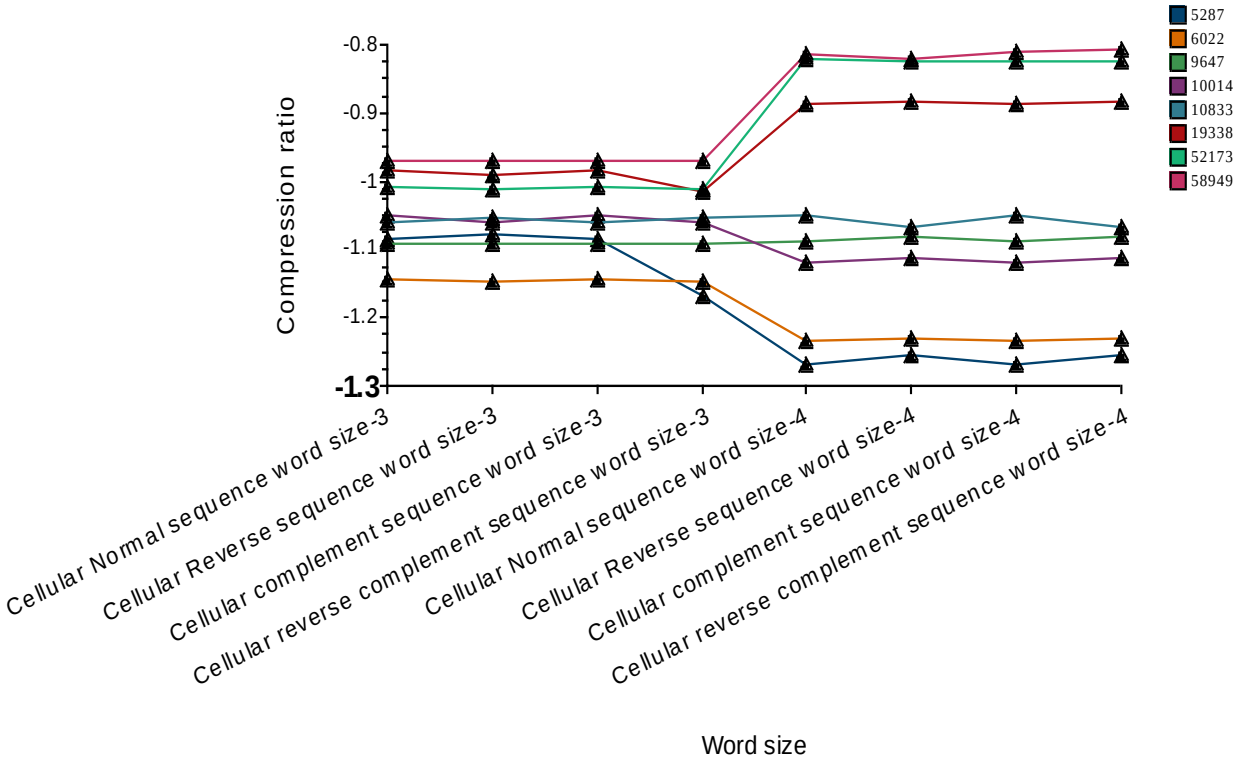

Fig. 3.10 compression ratio versus different word size among original, reverse, complement, reverse complement sequence based on Genetic palindrome and palindrome technique of different file size

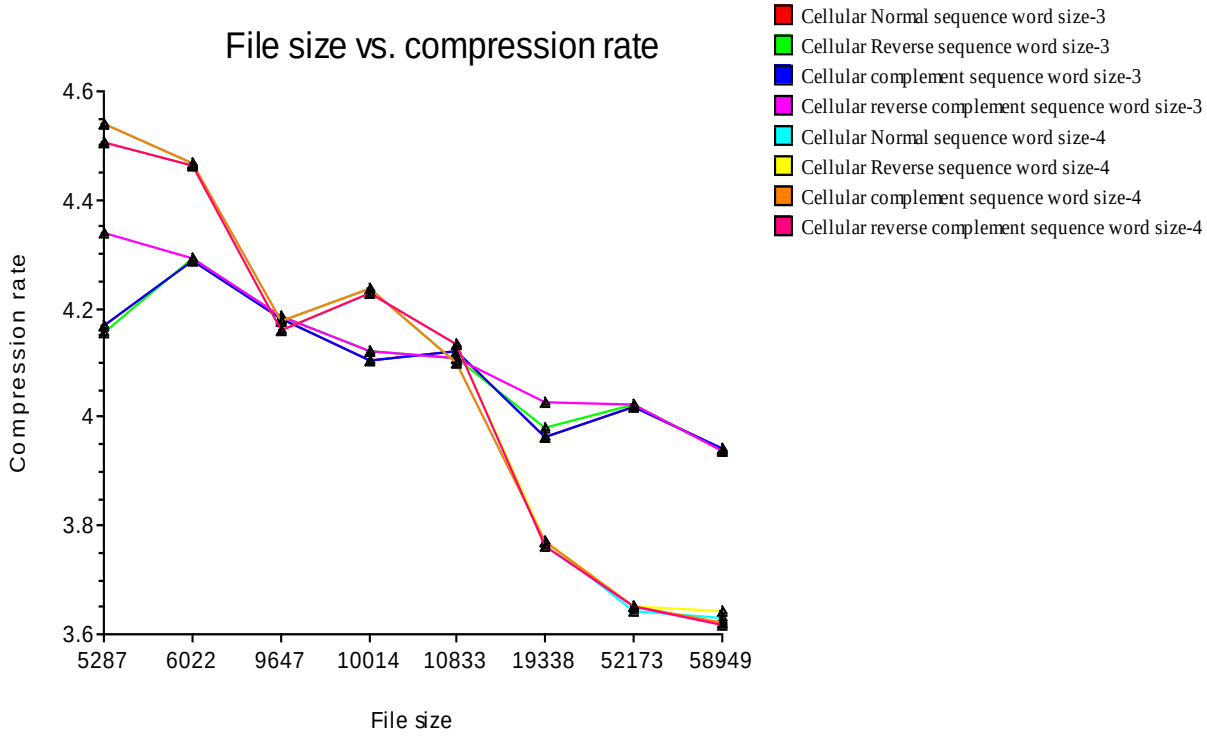

Fig. 3.11 compression rate versus file size among original, reverse, complement, reverse complement sequence based on genetic palindrome with palindrome technique of different word size.

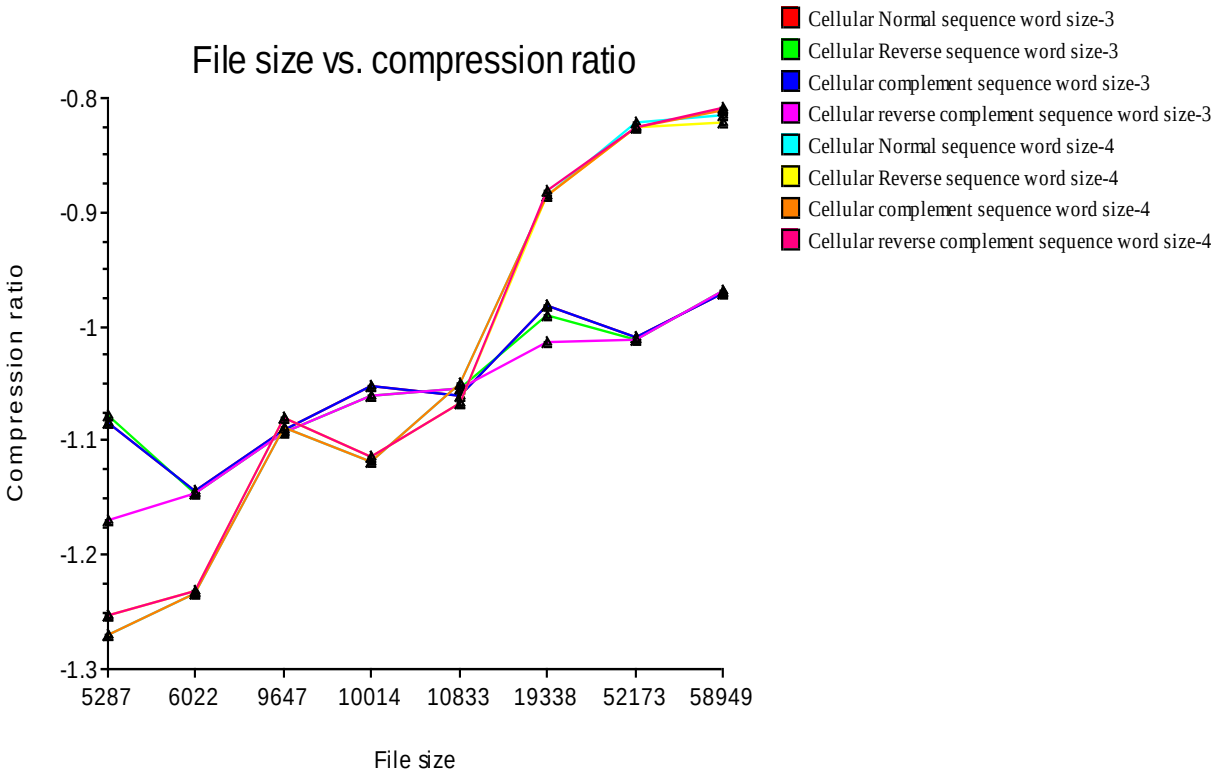

Fig. 3.12 compression ratio versus file size among original, reverse, complement, reverse complement sequence based on genetic palindrome with palindrome technique of different word size.

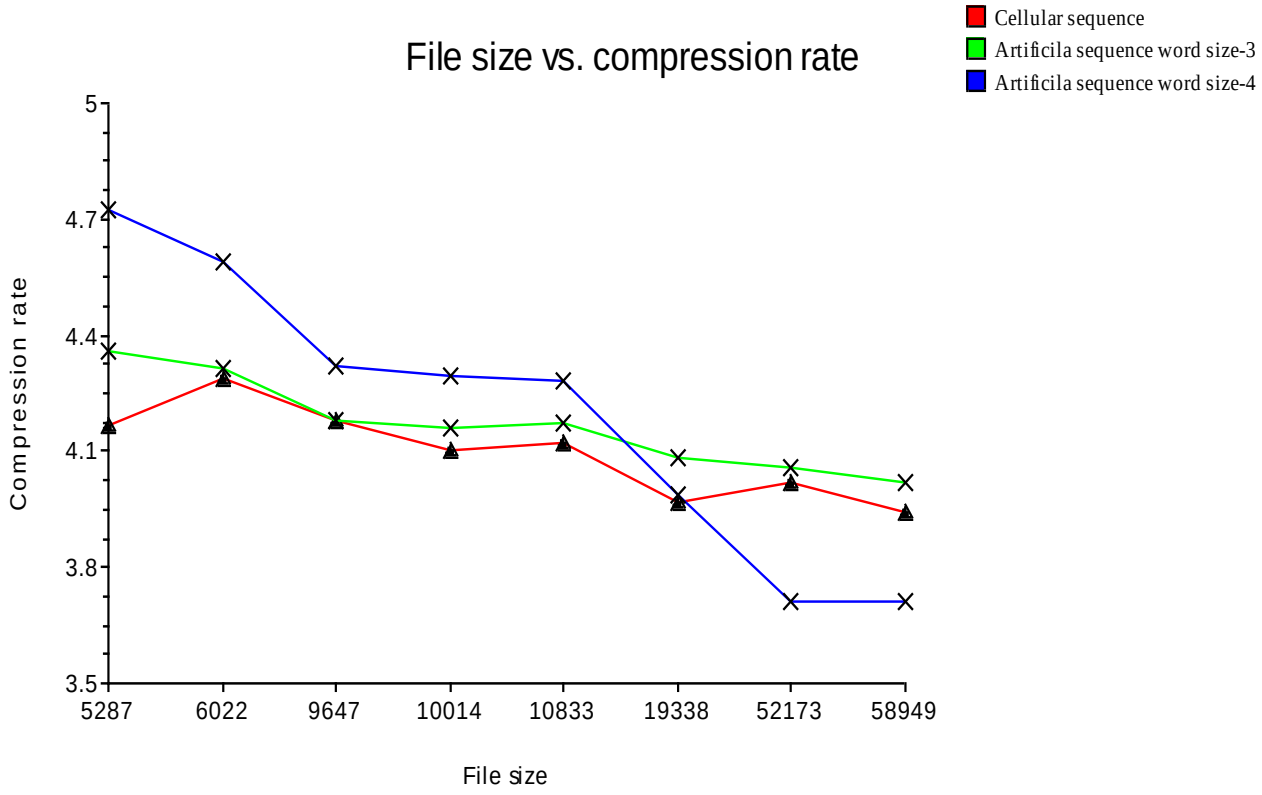

Fig. 3.13 compression rate versus file size of cellular and artificial sequence based on genetic palindrome with palindrome technique of different word size.

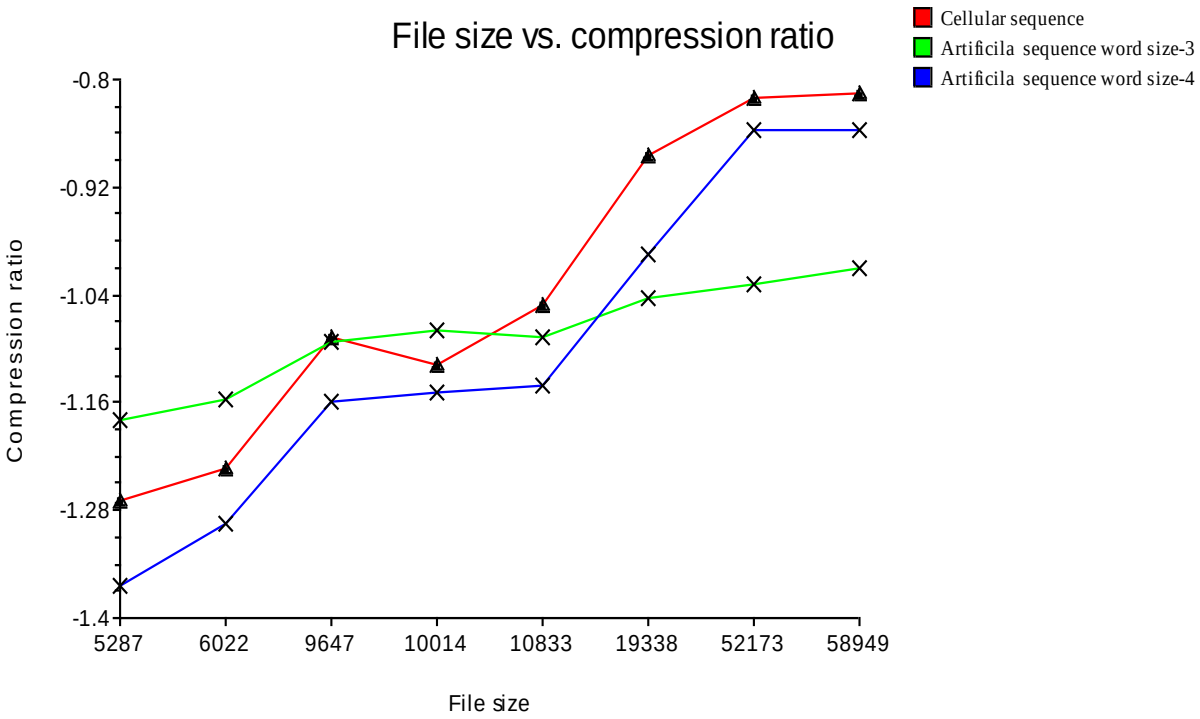

Fig.3.14 compression rate versus file size of cellular and artificial sequence based on genetic palindrome with palindrome technique of different word size.

Table 3.3 % encryption and % modification on actual text

Sub string size	Sequence Name	Base pair/ File size	Compress file size	Library file size	Lavenstein Distance	% of Encryption ratio	% effect on actual text	Entropy before compression	Entropy after compression	
									Based on compressed file	Based on library file
Sub string 3	atatsgs	9647	4377	224	9178	45.371618	95.138385	1.93269	4.65388	3.01161
	atef1a23	6022	2716	210	5743	45.101295	95.366987	1.97207	4.64976	3.00146
	atrdnaf	10014	4470	224	9578	44.637507	95.646095	1.99287	4.80697	3.01566
	atrdnai	5287	2387	224	5044	45.148477	95.403820	1.99159	4.63982	3.01782
	celk07e12	58949	26399	224	56413	44.782778	95.697976	1.96998	4.71711	3.01890
	hsg6pdgen	52173	23613	224	49780	45.259042	95.413336	1.99773	4.70485	3.00917
	mmzp3g	10833	4907	224	10300	45.296778	95.079848	1.99465	4.67461	3.01161
	xlxfg512	19338	8658	224	18519	44.771951	95.764815	1.99220	4.68326	3.01890
	Average	21532.88	9690.87	222.25		45.04618	95.43891	1.98047	4.69128	3.01314
Sub string 4	atatsgs	9647	3929	672	9104	40.73	94.37	1.93269	4.97931	3.19335
	atef1a23	6022	2407	592	5729	39.97	95.13	1.97207	5.05737	3.17953
	atrdnaf	10014	4080	720	9501	40.74	94.88	1.99287	5.10467	3.21104
	atrdnai	5287	2182	496	5022	41.27	94.99	1.99159	4.83867	3.14415
	celk07e12	58949	23483	1016	55926	39.84	94.87	1.96998	4.68740	3.20510
	hsg6pdgen	52173	20820	1024	49593	39.91	95.05	1.99773	4.73630	3.13172
	mmzp3g	10833	4404	720	10239	40.65	94.52	1.99465	5.07769	3.21303
	xlxfg512	19338	7662	856	18356	39.62	94.92	1.99220	4.86589	3.18869
	Average	21532.88	8620.87	762		40.34125	94.84125	1.98047	4.91841	3.18332

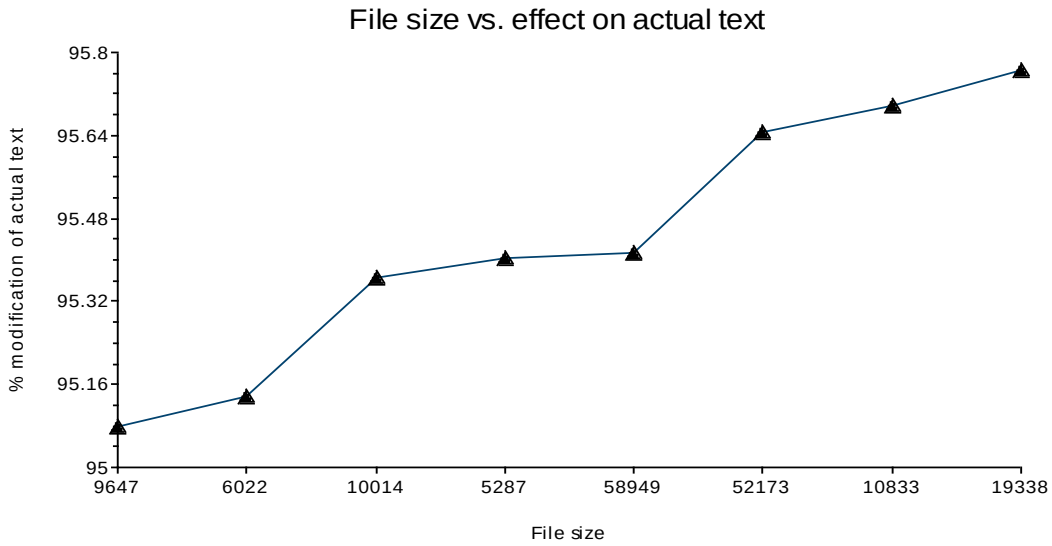

Fig.3.15 % modification on actual text versus file size

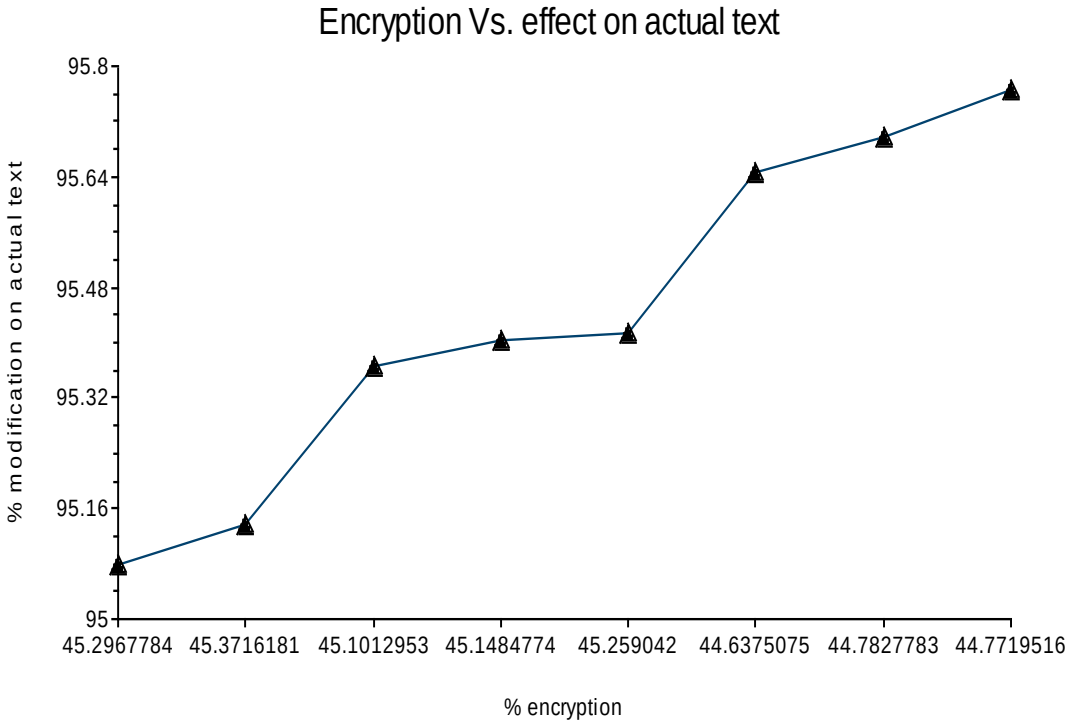

Fig. 3.16 % modification on actual text versus % encryption

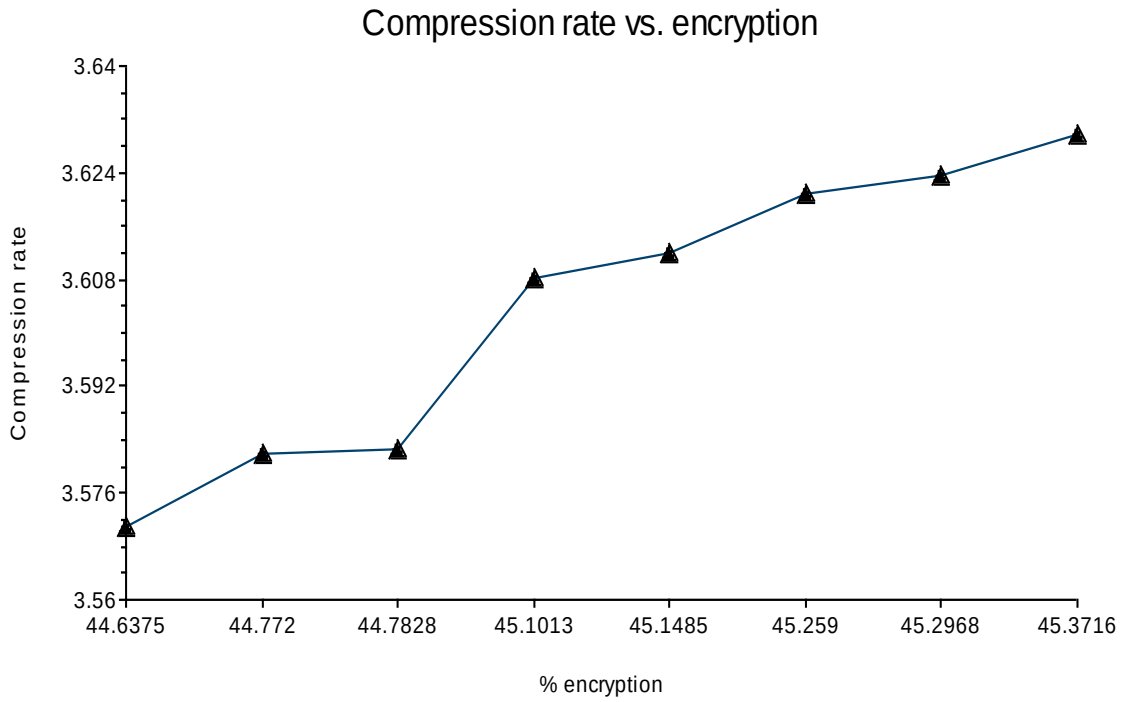

Fig. 3.17 % encryption versus compression rate

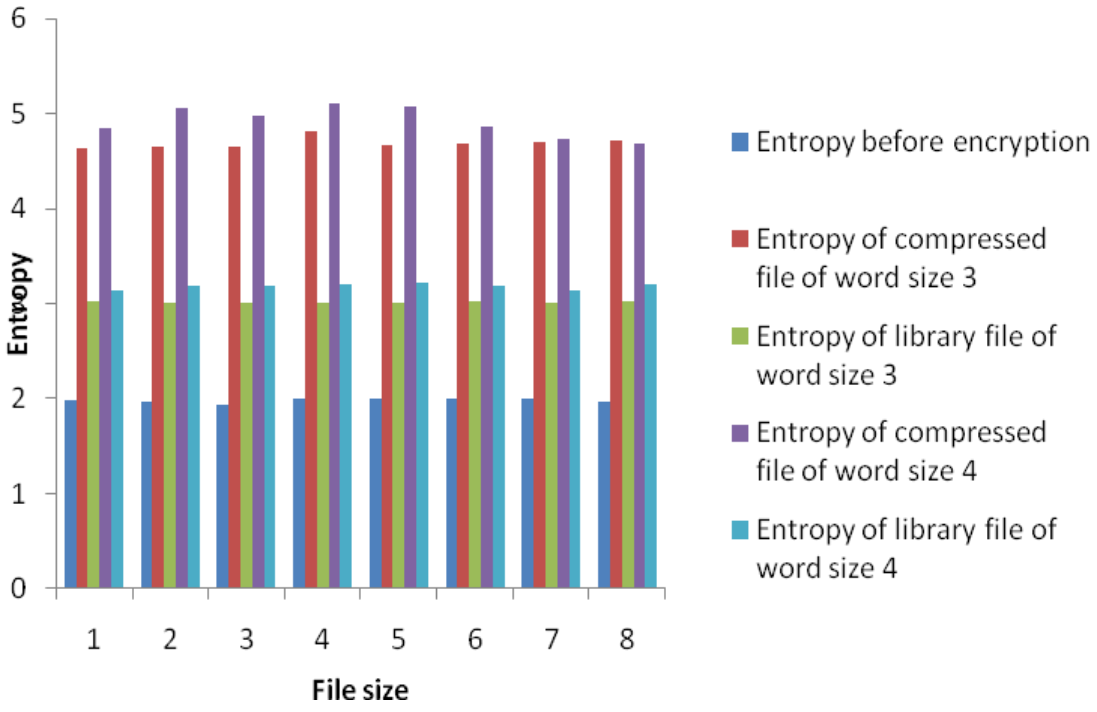

Fig. 3.18 entropy coding in genomic sequences versus file size

The results for Reverse & palindrome are presented in table 3.1, while the results for genetic palindrome with palindrome technique are represented in table 3.2. These values are graphically presented in fig. 3.3 to 3.8 for Reverse & Palindrome and 3.9 to 3.14 for Genetic Palindrome & Palindrome. The fig. 3.3, 3.9 & 3.4,3.10 shows that the compression rate & ratio varies with word size and independent of file size. The fig. 3.5,3.11 & 3.6,3.12 shows the more or less compression rate & ratio same of the different orientation, if the word size is 4 , the minimum compression rate is 3.7750 bits/base for reverse with palindrome & 4.0654 bits/base for genetic palindrome with palindrome. If the word size increases, the compression rate & ratio decrease. The fig. 3.7,3.13 & 3.8,3.14 shows that the artificial compression rate and cellular sequence compression rate are completely different. It is also observed that the compression rate is similar in a particular word size because the sequence comes from different species structure and matching pattern are same but in case of artificial data the compression rate is dissimilar because this data is random, shown in fig. 3.7,3.13 & 3.8,3.14. The table 1 and 2 shows the space saving percentage for both cases of RP & GP^2, lower the compression rate higher the space saving percentage.

If the sequence is encrypted by group of four characters, the percentage of encryption & percentage of modification of actual sequence is reflected in the table 3.3. This data is graphically presented in fig. 3.15 to 3.18. This data shows that every time the encryption is increasing in nature, providing high security of the data. Our result shows that the entropy increases two or three times after encryption. It is experimentally found that whenever the entropy increases simultaneously the information randomness also increases. Therefore, it is very difficult to decrypt the sequences by an attacker. The sequence internal matching pattern is similar as shown in fig. 3.14 by this technique and dissimilar in artificial sequence. The output is the mixture of ASCII code and nucleotide bases which provides the data security over transmission point of view. The proposed algorithm is very helpful in database storage.

4 Conclusions

The nature of DNA sequences compression rate is homogeneous in any DNA sequence orientation whereas heterogeneous in artificial sequences. The substring of DNA sequence repeat of reverse, palindrome & genetic palindromes techniques is described in this paper. It is observed that the DNA sequence is not random and also not chaotic. So, living beings are in a systematic way. This algorithm is a part of a perfect model of compression and reveals the true characters of DNA sequence. It also finds, the DNA regularities including mutation,

crossover. In order to get a better compression rate & ratio , the word size has been increased. It is observed that the rate of compression is decreased, at the same time the library file size is tremendously increased. Also Execution time is increased. Therefore, word size 3 base compression is better than other word size 4 or more. The higher the substring size the lower the matching probability, the substring size ranging from 2 to 6. The higher matching probabilities are found out when the substring size is 3. The technique Reverse & Palindrome (RP) / Genetic Palindrome & Palindrome (GP^2) is used for gene comparison.

Dynamic key base encryption provides better security. The algorithm has reduced the storage space & transmission time over the internet. Our selective encryption process has reduced the computational cost. This process does not compromise with speed & the security level of encryption. This technique performs the compression as well as encryption at the same time. To apply this technique on the DNA sequence, it is observed that by only 44% to 45% encryption can damage 95% on actual sequence and compression rate is maintained between 3.7750 bits/base to 4.1975 bits/base. After encryption, the entropy increases from 1.98 to 4.91 per byte. The degree of randomness of an information is measured by entropy. Randomness is a very important and highly desirable characteristic of compression encryption process. The output information of compressed file and library file yields high degree of randomness, making the output less susceptible to attacks. As a future scope of this work the modified Huffman's technique may be introduced to reduce the time complexity, to improve the encryption level and compression rate & ratio.