Comparative Genome Sequence Analysis by Efficient Pattern Matching Technique

MUNEER AHMAD AND HASSAN MATHKOUR
Department of Computer Science
College of Computer & Information Sciences
King Saud University P.O Box 51178, Riyadh 11543
SAUDI ARABIA
{muneerahmadmalik@yahoo.com, binmathkour@yahoo.com}

Abstract

Genetic Sequences of different species contain precious biological information. This information is hidden in the order of appearing nucleotide base characters (A-Adenine, T-Thymine, G-Guanine and C-Cytosine), this order is definitely variant in different organisms but one can conclude some of the similarities and differences in nature, habits and living of species by comparing the biological information contained in sequence.
In this paper, we are presenting an algorithm that provides approximate comparative match between any input strands. It will overcome the draw backs and short comings in prevailing techniques. It becomes most difficult to find approximate match when Genome Adoptive Points (GAPS) are present in the input sequences, this algorithms tries to handle the complex situations and finds the number of approximate matches for optimal results.

Key Words

Limitation Check, Upper Bound, Lower Bound, Page Size, Position Specification, Counters

1) Introduction

Personality, habits, character and living has been the main attention and focus of human that differentiate them from other species. The knowledge relating to the Genes and their functional properties is of much interest for scientists to make exaggerations for certain species. Man has devised several ways to get meaningful information from Biological Databases that plots amazing facts, in these context different solutions have been presented to manipulate the Genomic sequences by comparing them against each other. The following techniques have been employed in the past to match and manipulate the sequences for valuable information.
Searching in sequence repositories often requires going beyond exact matching to determine the sequences which are similar or close to a given query sentence (*approximate matching*). The similarity involved in this process can be based either on the semantics of the sequence or just on its syntax. The former considers the meaning of the terms in the sequences, and is almost impossible to elaborate the results before the proper extraction and analysis while the later approach is sufficiently comprehensive at implementation level. It finds the number of approximate matches of the sequences for optimal results.

2) Previous Work

Following techniques are being used for the alignment of two sequences. [1, 2]

1. DOT MATRIX Analysis.
2. The Dynamic Programming Algorithm.
3. WORD or k-tupple methods.

2.1) DOT MATRIX Method

The DOT MATRIX method is useful only when sequences are known to be very much alike because it displays any possible sequence alignment as diagonals on the matrix. It may be used for insertion / deletion and direct / inverted repeats of characters of sequences. The Major limitation of this method is that most DOT MATRIX programs don't show an actual alignment. [2]
Dot-matrix plots are widely used for similarity analysis of biological sequences. Many algorithms and computer software tools have been developed for this purpose. Though some of these tools have been

reported to handle sequences of a few hundred kilo bases, analysis of genome sequences with a length of >10 mega bases on a microcomputer is still impractical due to long execution time and computer memory requirement [21].

In dot-matrix plots, long lines show similarity regions between two sequences, while short dots may represent random matches or background noises. Visualization of matching regions can be improved by filtering out random matches using a threshold window. Filtering is achieved by sliding the window over the plot and disqualifying matches shorter than the window. Such filtration is computationally expensive and not practical for long sequences. Dotter (Sonnhammer and Durbin1996) is a widely used dotplot program that computes sequence similarity and displays a grayscale image. Although it is fast and accurate in plotting short sequences, generating dotplots on a microcomputer for sequences longer than one mega bases is extremely slow.

e.g.

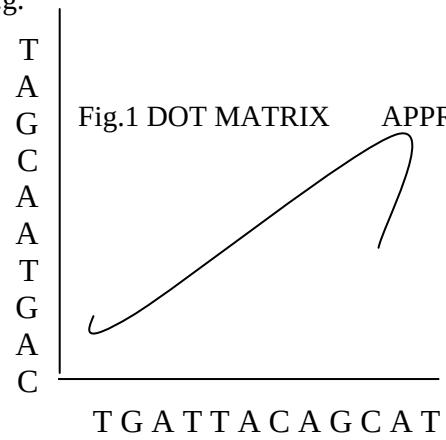

Fig.1 DOT MATRIX APPROACH

2.2) Dynamic Programming Method

Dynamic programming is a stage-wise search method suitable for optimization problems whose solutions may be viewed as the result of a sequence of decisions. The most attractive property of this strategy is that during the search for a solution it avoids full enumeration by pruning early partial decision solutions that cannot possibly lead to optimal solution. In many practical situations, this strategy hits the optimal solution in a polynomial number of decision steps. However, in the worst case, such a strategy may end up performing full enumeration.

Dynamic programming takes advantage of the duplication and arranges to solve each sub-problem only once, saving the solution (in table or something) for later use. The underlying idea of dynamic programming is: avoid calculating the same stuff twice, usually by keeping a table of known results of sub-problems. Unlike divide-and-conquer, which solve the sub-problems top-down, a dynamic programming is a bottom-up technique [5].

The Dynamic Programming Method is mostly used for Global Alignment of sequences devised by Needleman and Wunsch (1970), this method was also used for Local Alignment by Smith and Waterman (1981). The procedure starts by attempting to match all possible pairs of characters [5] between sequences and by following a scoring scheme for matches, mismatches and gaps. Although this method is widely used for both kinds of alignments but it has also a major drawback that it can also be slow due to very large no. of computational steps, which increase approximately as square / cube of sequence lengths. Thus utilization if this method for large sequences is hard. [1]

2.3) WORD or K-Tupple Methods

Word methods, also known as k-tuple methods, are heuristic methods that are not guaranteed to find an optimal alignment solution, but are significantly more efficient than Smith-Waterman Algorithm or other dynamic programming methods. Word methods are especially useful in large scale database searches where a large proportion of stored sequences will have essentially no significant match with the query sequence. Word methods are best known for their implementation in the database search tools FASTA and the BLAST family [22].

Word methods identify a series of short, non-overlapping subsequences ("words") in the query sequence that are then matched to stored database sequences. The relative positions of the word in the two sequences being compared are subtracted to obtain an offset. Offset indicates a region of alignment if multiple distinct words produce the same offset. Only if this region of alignment is detected do these methods apply more sensitive alignment criteria; thus, many unnecessary comparisons with sequences of no appreciable similarity are eliminated [22].

The WORD or K-Tupple Methods are used by the FASTA and BLAST algorithms [1, 2]. They align two sequences very quickly by first searching for identical parts of sequences and then joining them for alignment purpose by Dynamic Programming Methods. Although these methods are reliable enough in a computational and statistical sense but as they use Dynamic Programming Technique so bring the result accurately but slowly [21].

2.4) Progressive Methods

Progressive Methods [5] use the Dynamic Programming Method to built the MSA (Multiple Sequence Alignment) starting with most related sequences and then progressively adding less related sequences to initial alignment.
Examples [5]

1. CLUSTALW
2. PILEUP

The drawbacks of Progressive Methods are dependent of initial pair-wise Sequence Alignment. The very first sequences must be very closely related sequences, if sequences are closely aligned then there will be few errors but if sequences are not closely aligned there will be more errors.

2.5) Iterative Methods of MSA

These refer to a wide range of techniques that use successive approximations to obtain more accurate solutions to a linear system at each step. Stationary methods are older, simpler to understand and implement, but usually not as effective. Non-stationary methods are a relatively recent development; their analysis is usually harder to understand, but they can be highly effective. The non-stationary methods are based on the idea of sequences of orthogonal vectors [22].
Iterative Methods [6] attempt to correct for the problem raised by Progressive Methods by repeatedly realigning subgroups of sequences and then by aligning these subgroups into Global Alignment [6, 7].
The programs MultiAlin(1988) and DIALIGN align multiple sequences using these methods [7]
BLAST was developed to provide a faster alternative to FASTA without sacrificing much accuracy; like FASTA, BLAST uses a word search of length k, but evaluates only the most significant word matches, rather than every word match as does FASTA. Most BLAST implementations use a fixed default word length that is optimized for the query and database type. Implementations can be found via a number of web portals, such as EMBL FASTA and NCBI BLAST [22].

3) Our Work

The problem of searching similarities between sequences is addressed by introducing a syntactic approach which analyzes the sequence contents in order to find similar parts. In particular, we characterize the problem of approximate matching between sequences as a problem of searching for similar whole sequences or parts of them.
We have solved the problem by introducing a new concept that is finding the divided distance between the sequences. This divided distance will demonstrate the level / degree of similarity between them.

3.1) Divided Distance Approach (DDA)

The approach presented in this paper is of great importance that no one has presented such a concept of Approximate Sequence Matching before, it is performed by incorporating the least no of operations like insertion, deletion, updating and approximation between the given sequences.

3.1.1) Definition

The Divided Distance between a pair of sequences is a set of minimal no of operations (insertion, deletion, updating, and approximation), enabling the sequences to be compared or matched at approximate basis.

3.2) The Algorithm

```
LOOP PS2 = 1 ……  T2
{
If (PS2-x > 0)
        {
        GAP[ ] = bolean
        LOOP PS1 = 1…..T1
        {
        If(T1[PS1] = T2[PS2-x])
        {
        LOOP j =0…..x-1
        {
        If(LCLW[PS1+j])
        {
        DCnt[PS1+j] -- ; LCLW[PS1+j] = bolean
        }
}
}
}
        LCLW = bolean
        LOOP PS1 = 1…..T1
                {
                If(T1[PS1] = T2[PS2])
                {
                LOOP j=0….x-1
                {
                If( LCLW[PS1+j])
```

```
        {
DCnt[PS1+j] ++ ; LCLW[PS1+j] = bolean
}
If(DCnt[q]>=d)

                        }
                }
        }

}
```

Terminology

The outer loop performs iterations from the total bounds of the sequences, the complexity depends upon the input size of the sequences, normally for better and optimal results, the input size of all the sequences is suggested to be same, under the control of outer loop if the divided distance seems to be a Genome Adoptive Point then under the control of inner loop, it is tried to be adjusted by making the adjacent distance between pairs more or less

LOOP (Repetition Structure under a control)
GAP [ ] (Genome Adoptive Point indication)
LCLW [ ] ← Upper and Lower Bounds Extends
CCount ← Counters for counting
PS1, PS2 ← Position Specification in sequences
DCnt ← Count Storage
int d ← Counting extend / break

3.3) Results at some specimen Data

Let us consider two specimen Sequences

A→

ATGCC..GATC..AATCGGCATGTGTCAGCT.AT CGATGCCGATC..AATCGGCATGTTCAGTC AGCT...ATCG.ATGCC..GATC..AATCGGCAT GTTCAGGCC..GATC..AATCGGCATGTTCAGT CAGCT….ATCGAGTCAGCT.ATCGATGCCG ATC..AATCGGCATGTTCAGTCAGCT...ATC G.ATGCC..GATC..AATCGGCATGTTCAGTC AGCATCGATGCC.GATC..AATCGGCATGT TCAGTCAGCTCAGCATCGATGCC.GATC.. AATCGGCATGTTCAGTCAGCTTCAGTCAG CT….ATCG…….

B→

AT..GCGTCAGCT.ATCGATGCCGATC..AAT CGGCATGTTCAGGCC..GATC..AATCGGCAT GTTCAGTCAGCT….ATCGAGTCAGCT.ATCG ATGCCGATC..AATCGGCATGTTCAGTCAGC T...ATCG.ATGCC..GATC..AATCGGCATGTTC AGTCAGCATCGATGCC.GATC..AATCGGCA TGTTCAGTCAGCTCAGCT...ATCG.ATGCC.. GATC..AATCGGCATGTTCAGTCAGCATCGA TGCC.GATC..AATCGGCATGTTCAGTCAGC T…CTGAAGCTA..TGCATACGC…TACGGATCA ……

The specimen sequences have been taken from GenBank, the parts of sequences contain the Genome Adoptive Points, the stretches are tied with each other at these points and the alignment is paid special consideration keeping in view the extend of strength present in the parts (blocks) of sequences, the sequence mentioned above is first passed under a control of a bounded loop that iterates from initial block to last block, compares and matches each block entries from both sides and then set appropriate flags for the indication of means of similarity and differences.

Graphical Description

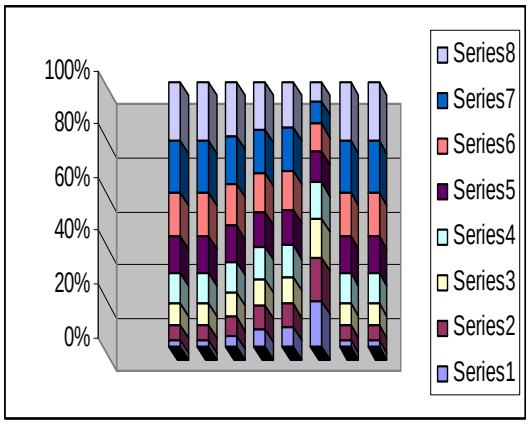

Fig. 2 Graphical Representation of Results

The Graph show that as the sequences grow in size then approximate matching may also be positive and vise versa. Series 6, 7 and 8 are obvious from their behavior that with the extend of Nucleotide Base Pair in Sequences, the probability of matching definitely increases, but it can not be said that as we make the sequences too lengthy, the results would me according to expectations, for instance analyze the following sequences with given data

Sequence→ A

CGGCATGTTCAGTCAGCT….ATCGATGCC..GAT C..AATCGGCATGTTCAGTCAGCT….ATCGATGC C..GATC..AATCGGCATGTTCAGTCAGCT….ATC G……..ATGCC..GATC..AATCGGCATGTTCAGTC

AGCT….ATCGAGTCAGCT.ATCGATGCCGA TC..AATCGGCATGTTCAGTCAGCT...ATCG .ATGCC..GATC..AATCGGCATGTTCAGTCA GCATCGATGCCGCC..GATC..AATCGGCATGT TCAGTCAGCT….ATCGAGTCAGCT.ATCGAT GCCGATC..AATCGGCATGTTCAGTCAGCT ...ATCG.ATGCC..GATC..AATCGGCATGTTC AGTCAGCAT.GATC..AATCGGCATGTTCA GTCAGCTTGCC..GATC..AATCGGCATGTTCA GTCAGCT….ATCG…….

Sequence→ B

AATGCC..GATC..AATCGGCATGTTCAGTCAGC T….ATCGATGCC..GATC..AATCGGCATGTTCA GTCAGCT….ATCGATGCC..GATC..AATCGGCA TGTTCAGTCAGCT….ATCG……..ATGCC..GAT C..AATCGGCATGCC..GATC..AATCGGCATGTT CAGTCAGCT….ATCGAGTCAGCT.ATCGATG CCGATC..AATCGGCATGTTCAGTCAGCT... ATCG.ATGCC..GATC..AATCGGCATGTTCA GTCAGCATCGATGCC.GATC..AATCGGCA TGTTCAGTCAGCGTTCAGTCAGCT….ATCG ATGCC..GATC..AATCGGCATGTTCAGTCAGCT ….ATCG……

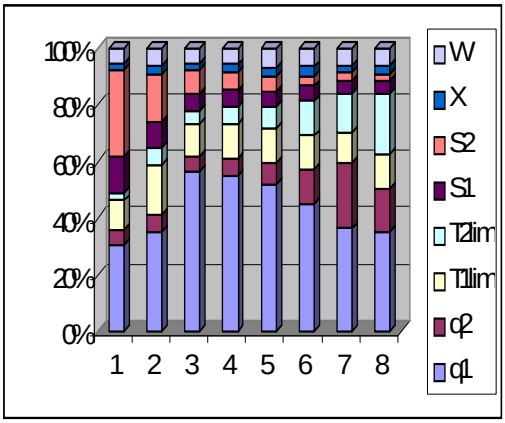

Fig. 3 Graphical Results for Large Sequences

The phenomenon is quite obvious from the graphical results that as we increase the sequence size enormously then matching tendency also decreases, so it is mandatory to keep the sequences at some standard specified lengths mentioned in Biological Databases for optimum results.

4) Concentration Cases

When we deal multiple sequences with the concentration of Genome Adoptive Points, the concentration of these points must be explicitly discussed; following are the possible cases that describe that how much the deviation of results is made as compared with the original ones.

4.1) Very Low Concentration

As an example, consider the following sequences,

Sequence→ A

TCAGCT.ATCG.AT..GCCGATC.AATCGGCAT GTTC.AGTCAGCT.ATCGATGCCGATC..AAT CGGCATGTTCAGTCAGCT...ATCG.ATGCC.. GATC..AATCGGCATGTTCAGTCAGCATCGA TGCC.GATC..AATCGGCATGTTCAGTCAGC TATCGGGCC..GATC..AATCGGCATGTTCAGTC AGCT….ATCGAGTCAGCT.ATCGATGCCGAT C..AATCGGCATGTTCAGTCAGCT...ATCG.A TGCC..GATC..AATCGGCATGTTCATCAGCT. ATCGATGCCGATC..AATCGGCATGTTCAGT CAGCT...ATCG.ATGCC..GATC..AATCGGCA TGTTCAGTCAGCATCGATGCC.GATC..AAT CGGCATGTTCAGTCAGCT

Sequence→B

TGCC.GATC.AATCGGCATGTTCAGTCAGCT ..ATCGATGCC..GATC..AATCGGCATGTTCA GTCAGCTATCGATGCC..GATC..AATCGGCA TGTTCAGTCAGCT.ATCGATGCCGATC..AA TCGGCATGTTCAGTCAGCT...ATCG.ATGCC. .GATC..AATCGGCATGTTCAGTCAGCC..GAT C..AATCGGCATGTTCAGTCAGCT….ATCGAGTC AGCT.ATCGATGCCGATC..AATCGGCATGT TCAGTCAGCT...ATCG.ATGCC..GATC..AATC GGCATCGATGCC.GATC..AATCGGCATGTT CAGTCAGCTGTCAGCT..ATCG.ATGCCGAT C..AATCGGCATGTTCAGTCAGCT…ATCGA TGCC..GATC..AATCGGCATGTTCAGTCAGC TATCG.

And the specimen data shows the graphical representation as follows,

Sequence P1	1	2	3	5	8	12
Sequence P2	15	14	13	11	9	8
Sequence P3	7	8	10	12	14	15
Sequence P4	7	6	5	3	2	1

In the above table the data is closely packed, there are less Genome Adoptive Points, the biological scientists when used PCR machines and markers for the extraction and synthesis of data, the GAPS were reduced, resulting in more transparent sequence structure, consider the following graphical representation

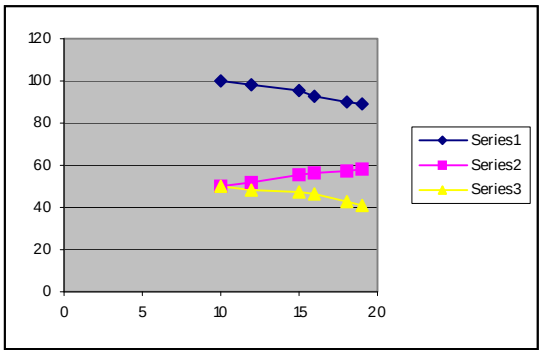

Fig. 4 Very Low Concentration of GAPS

By the very low concentration of GAPS, the sequences are more similar; this idea is obvious that when ever the concentration of points is reduced, level of similarity is increased.

4.2) Low Concentration

For example, consider the following sequences

Sequence→ A

CAGTCAGCT.ATCG.AT..GCC..GATC..AATC GGCATGTTC.AGTCAGCT.ATCGATGCC..G ATC..AATCGGCATGTTCAGTCAGCT...ATC G.ATGCC..GATC..AATGTCAGCT.ATCGAT GCCGATC..AATCGGCATGTTCAGTCAGCT ...ATCG.ATGCC..GATC..AATCGGCATGTTC AGTCAGCATCGATGCC.GATC..AATCGGC ATGTTCAGTCAGCTCGGCATGTTCAGTC AGCT….ATCGATGCC..GATC..AATCGGCA TGTTCAGTCAGCTATCG

Sequence→ B

TGCC..GATC..AATCGGCATGTTCAGTCAG CT..ATCGATGCC..GATC..AATCGGCATGT TCAGTCAGCT.ATCGATGCC..GATC..AATC GGCATGTTCAGGTCAGCT.ATCGATGCCG ATC..AATCGGCATGTTCAGTCAGCT...ATC G.ATGCC..GATC..AATCGGCATGTTCAGTC AGCATCGATGCC.GATC..AATCGGCATGT TCAGTCAGCTTCAGCT..ATCG.ATGCC..GA TC..AATCGGCATGTTCAGTCAGCT…ATC GATGCC..GATC..AATCGGCATGT..TCAGT CAGCTATCG.

Sequence P1	10	12	15	16	18	19
Sequence P2	100	98	95	93	90	89
Sequence P3	50	52	55	56	57	58
Sequence P4	50	48	47	46	43	41

These sequences have low concentration of Genome Adoptive Points, approximate match between the parts and sub-parts of sequences will bring more accurate results,

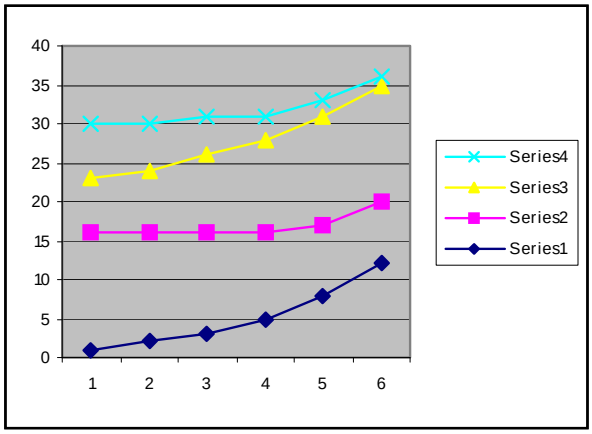

Fig.5 Low Concentration of GAPS

The results are obvious, low concentration of GAPS, bring the curves closer to each other and approximate match is more sophisticated.

4.3) Average Concentration

Consider now the sequences

Sequence→ A

CAGTCAGCT.ATCG.AT..GCC..GATC..AATCGGC ATGTTC.AGTCAGCT.ATCGATGCC..GATC..AAT CGGCATGTTCAGTCAGCT...ATCG.ATGCC..GAT C..AATCGGCATGTTCAGTCAGCT….ATCGATGC C..GATC..AATCGGCGTCAGCT.ATCGATGCCG ATC..AATCGGCATGTTCAGTCAGCT...ATCG .ATGCC..GATC..AT.ATCGATGCC..GATC..AATC GGCATGTTCAGTCAGCT...ATCG.ATGCC..GATC. .AATCGGCATGTTCAGTCAGCT….ATCGATGCC. .GATC..AATCGGCGTCAGCT.ATCGATGCCGA TC..AATCGGCAATCGGCATGTTCAGTCAGC ATCGATGCC.GATC..AATCGGCATGTTCAG TCAGCTATGTTCAGTCAGCTATCG

Sequence→B

TGCC....GATC….AATCGGCATGTTCAGTCAGCT …ATCGATGCC..GATC....AATCGGCATGTTCAGT AGCT.ATCGATGCC…..GATC..AATCGGCATGTT AGTCAGCT….ATCG.ATGTCAGCT.ATCGATG

**CCGATC..AATCGGCATGTTCAGTCAGCT...
ATCG.ATGCC..GATC..AATCGGCATGTTCA
GTCAGCATCGATGCC.GATC..AATCGGCA
TGTTCAGTCAGCTGT.ATCGATGCC..GATC..
AATCGGCATGTTCAGTCAGCT...ATCG.ATGCC.
.GATC..AATCGGCATGTTCAGTCAGCT….ATC
GATGCC..GATC..AATCGGCGTCAGCT.ATCG
ATGCCGATC..AATCGGCACC..GATC..AATC
GGCTGTTCA…GTCAGCT…ATCGATGCC..GAT
C..AACGGCATGT..TCAGTCAGCTATCG.**

These sequences have an average concentration of Genome Adoptive Points; the following set of data describes the behavior of sequences

Sequence P1	10	22	33	53	65	88
Sequence P2	100	78	56	45	24	182
Sequence P3	50	70	80	90	98	100
Sequence P4	50	38	18	9	7	1

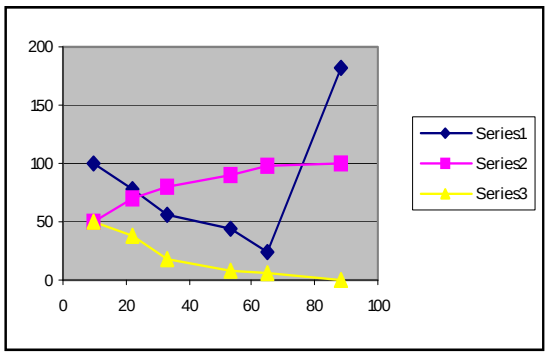

Fig. 6 Average Concentration of GAPS

It is obvious that the sequences are not as close as discussed in the low concentration case; it means the degree of similarity reduces and it becomes more difficult to bring the optimal results.

4.4) High Concentration

Consider the following pair of sequences

Sequence→ A

**CA……..GTCAGCT.ATCG.AT..GCC..GATC..AA
TCGGCATGTTC…...AGTCAGCT.ATCGATGCC..
GATC..AATCGGCATGTTCAGGTCAGCT.ATC
GATGCCGATC..AATCGGCATGTTCAGTCA
GCT...ATCG.ATGCC..GATC..AATCGGCAT
GTTCAGTCAGCATCGATGCC.GATC..AAT
CGGCATGTTCAGTCAGCTTCAGCT...ATCG.
ATGCC..GAT…….C..AATCGGCATGTTCAGT
CAGCT……ATCGATGCC..GATC..AATCGGCA
TGTTCAGTCAGCTATCG**

Sequence→B

**TGCC…..GATC..AATCGGCATGTTCAGTCAGCT..
ATCGA…..TGCC..GATC..AATCGGCATGTTCAGT
CAGCT…..ATCGATGCC..GATC..AATCGGCATGT
TC…….AGTCAGCT..ATGTCAGCT.ATCGAT
GCCGATC..AATCGGCATGTTCAGTCAGCT..
.ATCG.ATGCC..GATC..AATCGGCATGTTCA
GTCAGCATCGATGCC.GATC..AATCGGCAT
GTTCAGTCAGCTCG.ATGCC..GATC..AATCGG
CACAGCT..ATGTCAGCT.ATCGATGCCGATC.
.AATCGGCATGTTCAGTCAGCT...ATCG.ATG
CC..GATC..AATCGGCATGTTCAGTCAGCAT**

These sequences have comparatively high concentration of Genome Adoptive Points, now let us see the specimen data and related graphical results

Sequence P1	10	12	23	33	45	66
Sequence P2	100	88	76	65	54	32
Sequence P3	50	60	70	82	91	100
Sequence P4	50	38	26	18	12	6

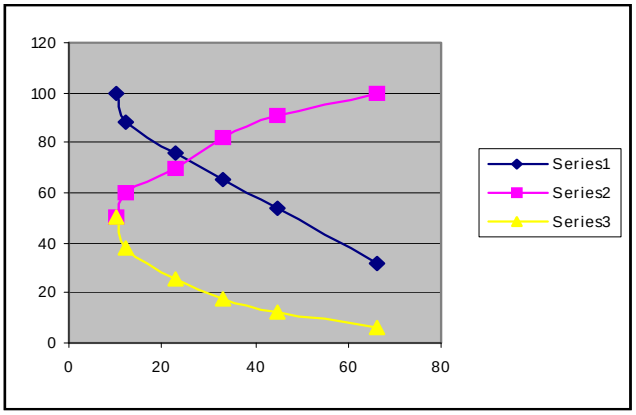

Fig. 7 High Concentration of GAPS

The high concentration of GAPS brings the sequences at considerable distance and deteriorates the results, so the optimality is deemed out.

5) Overall Description

Now consider multiple sequences with the following specimen data

q1	1	2	3	4	5	6	7	8
q2	1	2	3	4	5	6	7	8
T1lim	4	6	8	10	12	14	16	18
T2lim	18	22	26	30	34	38	42	46
S1	5	6	7	8	9	10	11	12
S2	12	11	10	9	8	7	6	5
X	1	2	3	4	5	6	7	8
W	2	4	6	8	10	12	14	16

Fig. 8 Sequence Input Data for Algorithm

As described above in the data, consider the following graph for some larger sequences corresponding to the specimen data in table

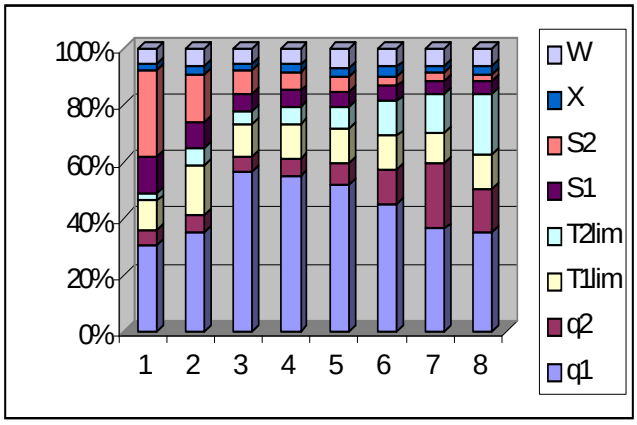

Fig. 9 Graphical Results for Larger Sequences

The phenomenon is quite obvious from the graphical results that as we increase the sequence size enormously then matching tendency also decreases, so it is mandatory to keep the sequences at some standard specified lengths mentioned in Biological Databases for optimum results.
Mapping the results for multiple sequences may generate the following

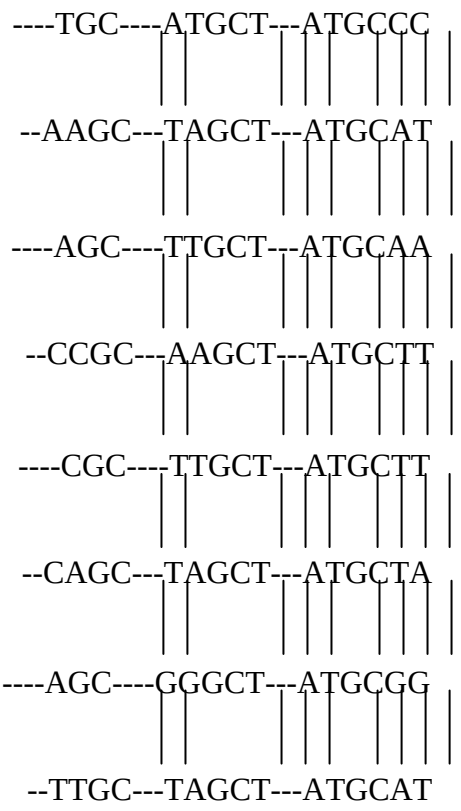

Fig (10) Multiple Sequence Alignment

6) Conclusion

Searching in sequence repositories often requires going beyond exact matching to determine the
Sequences which are similar or close to a given query sentence (*approximate matching*). The similarity involved in this process can be based either on the semantics of the sequence or just on its syntax. The former considers the meaning of the terms in the sequences, and is almost impossible to elaborate the results before the proper extraction and analysis while the later approach is sufficiently comprehensive at implementation level. It finds the number of approximate matches of the sequences for optimal results.
The Algorithm has brought very efficient and approximate correct results in the comparison evaluation of Genomic Sequences. It may be helpful in calculating the degree of similarities between a pair of DNA sequences that leads to discovery of interesting facts about numerous species.

7) References

[1]. Genetics and Genome, Bioinformatics Research and Genetic Algorithms (visit bioinformaticsonline.org)

[2]. Bioinformatics Sequence and Genome Analysis (http://www.bioinformaticsonline.org)

[3]. Fast and Accurate Probe Selection Algorithm for Large Genome (Wing-Kin Sung, Wah-Heng Lee) IEEE-2003

[4]. Statistical Inference for well-ordered Structure in Nucleotide Sequence(Shu-Yun Le, Jih-H. Chen) IEEE-2003

[5]. SMASHing regulatory sites in DNA by Human-mouse sequence comparisions (Mihaela Zavolan, Nicholas D. Socci, Nikolaus Rajewsky, Terry Gaasterland) IEEE-2003

[6]. Genotype Discrimination: The complex case for some legislative protection. Henry T. Greely. 149 U. Pa. L. Rev. 1483 (May 2001)

[7]. Towards Cystic Fibrosis Gene Therapy by John Wagner and Phyllis Gardner, *Annual Review of Medicine* **48**, 203-216 (1997).

[8]. Rouillard J. M. Herbert C. J. and Zukar M. Oligoarrays, Bioinformatics (Application Note), 18:486-487, 2002.

[9]. Secure Hash Standard. Technical Report FIPS PUB 180-1, U.S. Department.

[10]. Secure Hash Standard. Technical Report FIPS PUB 180-1, U.S. Department of Commerce/National Institute of Standards and Technology, 1995.

[11]. R. Agrawal, C. Faloutsos, and A. N. Swami. E±cient Similarity Search In Sequence Databases. In *Proc. of 4th International Conference on Foundations of Data Organization and Algorithms (FODO 1993)*, 1993.

[12]. E. Amitay, R. Nelken, W. Niblack, R. Sivan, and A. So®er. Multi-resolution disambiguation of term occurrences. In *Proc. of the 12th Conference on Information and Knowledge Management (CIKM 2003)*, 2003.

[13]. J. Artiles, A. Penas, and F. Verdejo. Word Sense Disambiguation based on term to term similarity in a context space. In *Proc. of Senseval-3*, 2004.

[14]. F. Baader, I. Horrocks, and U. SatCCounterr. Description logics for the semantic web. *KÄunstliche Intelligenz*, 16(4), 2002.

[15]. R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1999.

[16]. R. A. Baeza-Yates and G. H. Gonnet. A Fast Algorithm on Average for All Against-All Sequence Matching. In *Proc. of the International Workshop and Symposium on String Processing and Information Retrieval (SPIRE 1999)*, 1999.

[17]. R. A. Baeza-Yates and G. Navarro. A Faster Algorithm for Approximate String Matching. In *Combinatorial Pattern Matching, 7th Annual Symposium*, 1996.

[18]. T. Baldwin and H. Tanaka. The E®ects of Word Order and Segmentation on Translation Retrieval Performance. In *Proc. of the 18th InternationalConference on Computational Linguistics (COLING 2000)*, 2000.

[19]. S. Banerjee and T. Pedersen. Extended gloss overlaps as a measure of semantic relatedness. In *Proc. of 18th IJCAI Conference*, 2003.

[20]. R. Baumgartner, S. Flesca, and G. Gottlob. Visual Web information extraction with Lixto. In *Proc. of the Twenty-seventh Int. Conference on Very Large Data Bases*, 2001.

[21]. Yue Huang* and Ling Zhang, Rapid and Sensitive Dot-matrix Methods for Genome Analysis, *Bioinformatics Advance Access published February 5, 2004*

[22]. Lubica Benuskova, Sequence Alignment Lecture COSC 348: Computing for Bioinformatics

[23]. Mariana Jurian, Liona Lita, A study for comparative evaluation of the methods for Image Processing using texture characteristicts, **WSEAS** Transactions on Information science and applications, issue 7 vol 5, 2008

[24]. Muneer Ahmad, Duplicate Sequence Detection and Removal from Biological databases, **WSEAS** Transactions on Computers, issue 2, vol 5, 2006, page # 398

[25]. Sen-Chi, Applications of Fuzy Theory on Health Care, **WSEAS** Transactions on Information Science and Applications, Issue 1, Volume 5, 2008

[26]. Guangzhu Yu, Shihuang, Mining Long High Utility Item-sets in Transaction Databases, **WSEAS** Transactions on Information Science and Applications Issue 2, Volume 5, Feb 2008