International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 5– No.1, January 2013 – www.ijais.org

Bio-Software Engine Analysis for High Concentration Regions (HCR) Detection of a Fixed Base String on an Affected Sequence

Syed Mahamud Hossein
District Officer, Regional Office
D.V.E.T., West Bengal

A. K. Bandyopadhyay
CSE Dept., Govt. College of Engineering
& Ceramic Technology

ABSTRACT

A program to enable optimal alignments searching between two sequences, the host sequence (normal plant) and query sequence (Virus). Looking for homologues has become a routine operation of biological sequences in 4X4 combination with different subsequence (word size). The program takes advantage of the high degree of homology between such sequences to construct an alignment of the matching regions. The main aim is to detect the overlapping reading frames. This program also enable to find out the highly infected colones selection highest matching region with minimum gap or mismatch zones and unique virus colones matches. This is a small, portable, interactive, front-end program intended to be used to find out the regions of matching between host sequence and query subsequences. All the operations are carried out in fraction of seconds, depending on the required task and on the sequence length.

Keyword: Homology searching, HCRD, FBS

1. INTRODUCTION

It is known that VIROIDS are the smallest replicating pathogenic agents[1], which is entirely composed of RNA with genome sizes in the range of 330−380 nucleotides[2] that is 10 times smaller than the smallest bacteriophase of Escherichia coli[3]. It is also known that they infect a wide variety of plants and produce severe disease symptoms in many plant [4], but there is no evidence for the existence of a protective protein coat for viroids. The molecular mechanisms by which viroids replicate and interact with their hosts are not yet understood. In its most severe form, the disease[5] caused by potato spindle tuber viroid (PSTV) causes general stunting of potato plant growth, deformity of the upper foliage, and production of disfigured potatoes[6]. Mild strains of PSTV which produce barely detectable symptoms have also been isolated [7]. Furthermore, plants infected by mild strains are somehow protected from developing symptoms following subsequent inoculation with severe strains[6,8]. This biological protection phenomenon has been called cross-protection[6]. As the only known component of viroids is RNA, mild and severe strains must differ in nucleotide sequence. The study described here is aimed to determine the extent of this difference[1]. The sequence of the 247 nucleotide residues of the single strand circular RNA of avocado sunblotch viroid (ASBV) was determined using partial enzymes cleavage methods on overlapping viroid fragments obtained by partial ribonucleic digestion followed by 32p-labelling in vitro at their 5'-ends. ASBV is much smaller than potato spindle tuber viroid (PSTV; 359 residues) and chrysanthemum stunt viroid (CSV; 356 residues). A secondary structure model for ASBV is proposed and contains 67% of its residues base paired. In contrast to the extensive (69%) sequence homology of CSV with PSTV, only 18% of the ASBV sequence is homologous to PSTV and CSV. There are eight potential polypeptide translation products with chain lengths from 4 to 63 amino acid residues coded for by the plus (infectious) strand and four potential translation products (2 to 60 residues) coded for by the minus strand. An improved method is described for the synthesis of -32p- ATP of high specific activity (Ref. Gross, 1978).

The nucleotide sequence of 5.8S rRNA from the Chinese silkworm, Philosamia cynthia ricini has been determined by gel sequencing and mobility shift methods. The complete primary structure is pAAACCAUUACCCUGGACGGUGGAUCACUUGGCUC GCGGG UCGAUGAAGAA

CGCAGUUAACUGCGCGUCAUAGUGUGAACUGmCAG GACACAUUUGAACAUCGAC

AUUUCGAACGCACAUUGCGGUCCGUGGAGACACAU CCAGGACCACUCCUGUCU

GAGGGCCGAUUAAOH. This is one of the largest known 5.8S rRNAs.

As compared to Bombyx 5.8S rRNA, it is two nucleotides longer; two nucleotides near the 5'end and two nucleotides near the 3'end are different, and 61 of the Bombyx RNA sequence is an unmodified Uin Philosamia RNA. The secondary structure of Philosamia 5.8S rRNA may differ from the Bombyx RNA structure by three additional base pairs at the 5'/3' ends (Ref. Gross, 1982). Fingerprint analyses of two potato spindle tuber viroid (PSTV) isolates causing severe and mild symptoms~ respectively, in tomato exhibited defined differences in the RNase T1 and RNase a fingerprints.

The complete sequencing of the mild isolate and the comparison of its primary structure with the previously established one of the pathogenic type strain revealed that oligonucleotides CAAAAAAG, CUUUUUCUCUAUCUUACUUG, and AAAAAAGGAC in the 'severe' strain are replaced by CAAUAAG, CUUUUUCUCUAUCUUUCUUUG, AAU, and AAGGAC in the 'mild' strain. Thus, three nucleotide exchanges at different sites of the molecule may change a pathogenic viroid to a practically non-pathogenic isolate. The possible correlation between the secondary structures in a defined region of the PSTV molecule[4].

The causative agent of chrysanthemum stunt disease is a member of the unique group of plant pathogens known as viroids of which only eight have been described so far. The

sequence of the 356-nt residues of chrysanthemum stunt viroid (CSV) has been determined. Overlapping linear viroid fragments were obtained by partial ribonuclease digestion, radiolabelled *in vitro* at their 5'-ends, and sequenced using partial enzymic cleavage methods. Of the CSV sequence, 69% is contained in the published sequence of potato spindle tuber viroid (PSTV). Differences in the primary sequence of CSV and PSTV suggest that neither the positive nor putative negative str**and**s of these two viroids code for functional polypeptide products. However, the two viroids can form similar secondary structures, implying a role for viroid structure in replication[5].

Avocado sunblotch viroid (ASBV) is an important disease affecting avocado trees. Infections result in lower yields and poorer quality fruit. ASBV is the smallest known viroid that infects plants and is transmitted by pollen and infected seeds or budwood. The sequence of the 247- nt residues of the single strand circular RNA of avocado sunblotch viroid (ASBV) was determined using partial enzymic cleavage methods on overlapping viroid fragments obtained by partial ribonuclease digestion followed by 32p-labelling in vitro at their 5'-ends.

A secondary structure model for ASBV is proposed and contains 67% of its residues base paired. In contrast to the extensive (69%) sequence homology of CSV with PSTV, only 18% of the ASBV sequence is homologous to PSTV and CSV. There are eight potential polypeptide translation products with chain lengths from 4 to 63 amino acid residues coded for by the plus (infectious) strand and four potential translation products (2 to 60 residues) coded for by the minus strand. An improved method is described for the synthesis of -32p- ATP [7].

For most cases, such for RT-PCR, 3' or 5' RACE, they just used single stranded cDNA (the first strand cDNA), while in case of preparing the cDNA for library construction, or RDA/SSH analysis which might need additional second strand synthesis step to generate double strand cDNA. Restriction fragments composed of only hop stunt viroid (HSV) cDNA were prepared from recombinant clone pHS-P4P, which carries four tandemly repeated HSV cDNAs, and inoculated into cucumber cotyledons.

The results showed that double-stranded cDNAs consisting of 1 to 3 units of HSV sequences were infectious. In cucumber plants inoculated with these double-stranded cDNAs, infectious RNA molecules indistinguishable from authentic HSV were propagated (Ref. van Wezenbeek, 1982). Contiguous restriction fragments from two cloned partial-length potato spindle tuber viroid (PSTV) cDNAs were used to construct recombinant DNAs containing full-length monomeric and dimeric PSTV cDNA. When five different PSTV cDNA plasmids and RNA isolated from *E. coli* cells harboring these plasmids DNAs containing PSTV cDNA dimers were infectious when inoculated onto tomato seedlings. RNA transcripts containing the sequence of PSTV from these plasmids were also infectious. The sequences of the viroid progeny and the cloned DNA were identical. *In vitro* mutagenesis of infectious PSTV cDNAs will allow systematic investigation of the role of specific sequences in viroid replication and pathogenesis [9].

A complex of considerable stability is possible between the 5′ end of U1 RNA and a specific nucleotide sequence of the potato spindle tuber viroid complement. Small nuclear RNAs (snRNAs) that are associated with ribonucleoprotein particles are believed by some to be

involved in the processing of the primary transcription products of split genes. The 5' end of one such RNA, U1, has been shown to exhibit complementarity with the ends of introns, and it is believed that this affords a mechanism ensuring correct excision of the intron sequences and accurate joining of the coding sequences [10].

The invention provides a novel retroviral packaging system, in which retroviral packaging constructs and packagable vector transcripts are produced from high expression plasmids by replicating in a human's cell via the enzyme reverse transcriptase to produce DNA from its RNA genome. Retroviruses are enveloped viruses that belong to the viral family *Retroviridae*. High titers of recombinant retrovirus are produced in infected cells.

The methods of the invention include the use of the novel retroviral constructs to transducer primary human cells, including T cells and human hematopoietic stem cells, with foreign genes by cocultivation at high efficiencies. The invention is useful for the rapid production of high viral supernatants, and to transduce with high efficiency cells that are refractory to transduction by conventional means[11].

For the present paper, some points need to be indicated:

Similarity: To define similarity, perhaps it is useful to first introduce the notion of "distance" between two strings. The distance between two strings is zero if they are exactly the same. The distance between two strings increases if they get more dissimilar. One way of defining distance between two strings is to look at the amount of change they needed to do to one to obtain the other. They could go on to introduce other changes, insert and delete. Insert 'happens' when they inserted some letter into the sequence (at some position), and delete happens when they deleted some letter at some position.

Edit distance: This is defined as the minimum number of changes to be performed on one

sequence to make it exactly the same as another.

Alignment of sequences: For every two sequences, there are huge permutations of possible

alignments (cubic in the length of sequences). Alignment procedure itself can be visualized as

a series of insert, delete operations.

Scoring function: A scoring function determines this notion of goodness of alignment. They could compute the distance between alignments in such a way that the cost of a match is 0 (when the sequence on top and below have the same ith character). Cost of a mismatch is that they could choose different scoring schemes. Another sample scoring scheme could give lesser weights for replacement of A by T, and G by C (and vice versa) as against replacement of A by G or the others. Domain knowledge is used while determining scoring schemes.

2. METHODOLOGY:

2 **Match occurs in the following way**:

2.1 We look for matches in between Host sequence and Query sequence in the below figure

Q[i]=H[j] to H[m-L+1]

As for example Q[1]=H[1] first match found.

Next Q[2] match with H[1] to H[m-L+1]

This process will continue at the end of query sequence. This process is repeated at the end of

query sequence, until all possible match are found.

Match found then Q[i]=H[j],

Analysis of Method for each matching:

a) **Consider a DNA sequence and their related changes**

1 2 3 4 5 6 7 8 9 10 11 12……..…n

DNA CG G A A C T A A A C T C ….……..nn

RNA CG G A A C U A A A C U C ….……..nn

cDNA G C C T T G A T T T G A G ….……..nn

cRNA GC C U U G A U U U G A G ….……..nn

Where, n is the number of bases in the nucleotide sequence.

nn is the nth (i.e. last) base (A/T/G/C) in host and query genome sequences, which consist of

bases A,T,G and C. (note that T is replaced with U in the case of the RNA).

This example is applicable both in host and query sequences, and n is the length of the

sequence in both cases but they are same or not depends on user.

b) They broke the host and query sequence into user requirement subsequences length for

easy implementation of this algorithm [41]

Generating the query subsequence from input sequence:

2.2 :**Generating the substring from input sequence**

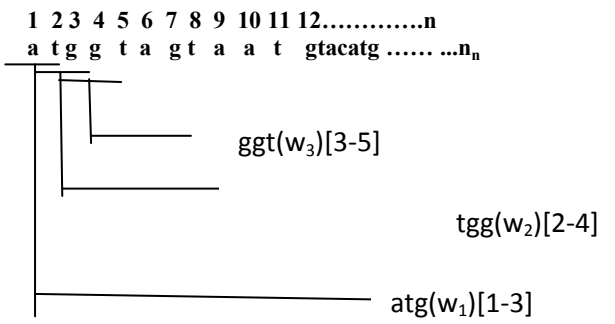


Fig.-I : Substring creation

From the above pictorial representation, it is clear that for ith subsequence Wi (called colons):

GAAC(W3)[3-6]

GGAA(W3)[2-5]

GGAA(W3)[1-4]

i, is the starting position of the subsequence and.

j= (i-1) + L, is end position of the subsequence; where L is the subsequence length (word

size).

For example, if word size is 4 then:

For: W1 starting position (i)=1 and (end position)j= (1-1) + 4=4,

W2 starting position (i)=2 and (end position)j= (2-1) + 4=5 and

W3 starting position (i)=3 and (end position)j= (3-1) + 4=6 and so on.

The clones with word size less than 3 (three) has no importance in matching context and hence we considered the clones with word size in the range: $3 \leq L \leq n$

Therefore, range for i and j are as $3 \leq i \leq n-L+1$ and $L+1 \leq j \leq n$ respectively.

The subsequence generation time, both in host and query sequences cases, at the end (subsequence length – 1) number of nucleotide base pair(a,t,g and c) remain as it is. This is the reason why probability of infection decreases. To solve this problem, we have to find the result in reverse order.

The host sequence is defined by H and query sequence is defined by Q, each of the sequences must have the same or different lengths.

So, we could write H=ATGCTAGCAGTAGACGATAGC……..n n>0

and T=TGCAGTAGCAGATGAC……m m>0

Where n and m is the length of host and query sequences.

After subsequence division they could get the result as below.

So, they could rewrite H[i]=H[1]H[2]……..H[n-L+1] $1 \leq i \leq n-L+1$

and Q[j]=Q[1]Q[2]……Q[m-L+1] $1 \leq j \leq m-L+1$

If the subsequence length or word size is L ($3<L \leq n-L+1$).

If the number of subsequence is S, total number of subsequence are generated in case of host

sequence is $1 \leq S \leq n-L+1$ and case of query sequences is $1 \leq S \leq m-L+1$.

This subsequence method is required to reduce the complexity of the programme execution.

c) Let us look for matches in between Host sequence and Query sequence in the following cases:

Source Sequence Target sequence

DNA DNA

RNA RNA

cDNA cDNA

cRNA cRNA

Here, host sequence is the virus sequence and Query sequence is the Tomato chloroplast,…etc , complete genome sequence of the Tomato plant and Root sequence.

16 possible matches may occur, and matches found are shown below:

DNA vs DNA

DNA vs RNA

DNA vs cDNA

DNA vs cRNA

RNA vs DNA

RNA vs RNA

RNA vs cDNA

RNA vs cRNA

cDNA vs DNA

cDNA vs RNA

cDNA vs cDNA

cDNA vs cRNA

cRNA vs DNA

cRNA vs RNA

cRNA vs cDNA

cRNA vs cRNA

In these cases, the value of i is incremented by i=no. of unmatched character+ no. of substring match X 3, similarly j is incremented by this same procedure.

Otherwise Q[i]≠H[j] i.e. unmatch occurs, the value of i and j is incremented by one.

At the end, we could get the result as follows:

H[1] H[5]H[6] ……H[n-L+1]

Source sequence S[i] : CGG C U AAAC…….n

Target Sequence T[i] : CG G A A C U A A A C U C …… ...m

T[1] T[4] T[5]……T[m-L+1]

Total word match = 3

d) Host and Query sequence infections are calculated by |NBM|/||TL| where NBM is the total no of base pair match that is equivalent to total number word match multiplied by word size, is divided by length of host sequence in case of virus infection, length of query sequence in case of plant infection.

e) Proving this hypothesis we have considered a threshold value, on this threshold value we can take the decision as described below:

Infectivity 'HIGH' means that the virus is highly infectious on target sequence, i.e. chloroplast of the tomato plant is infected by PSTVd virus from head to tail. In this situation, the infection between the source (PSTVd) and the target sequence (tomato chloroplast) is very high.

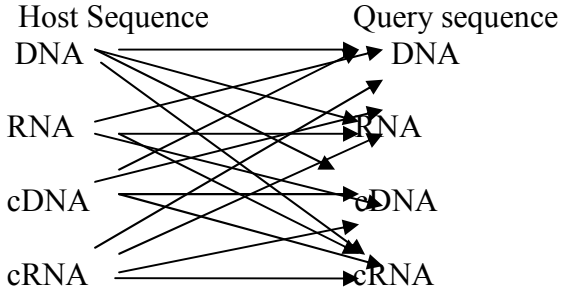
Infectivity 'NEGLIGIBLE' means that the virus is infected on target sequence, i.e. chloroplast of the tomato plant is infected by PSTVd virus from head to tail are not infected.

In this situation, the infection between the source (PSTVd) and the target sequence (tomato chloroplast) is infected, but it is not harmful.

Infectivity 'LOW' means the virus infection is found, but not so called infectious on target sequence, i.e. chloroplast of the tomato plant is infected by PSTVd virus from head to tail are not infected. In this situation, the infection between the source (PSTVd) and the target sequence (tomato chloroplast) is non infectious.

2.2 A Look for Matches between Host Sequence and Query Sequence:

This aspect is given in Fig. I. Next, we deal with the alignment demo.



Fugure-I

2.3 Alignment Demo:

This matter of alignment is shown in Fig. II. Next, let us deal with the pictorial representation for the 'match region'

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 ……n

a	t	g	g	t	a	g	t	a	a	t	g	t	a	c	a	t	g	c	a	t	g	…… ...n_n → Normal sequence
a	t	g	g	t	a	a	t	a	a	a	g	t	a	a	g	t	g	c	a	t	g	…… ...m_m → Virus sequence
+	+	+	+	+	+	-	+	+	+	-	+	+	+	-	-	+	+	+	+	+	+	→ match score
A[1]			A[3]			A[5]				A[1]						A[8]			A[4]			→ Word size

Figure-II

2.4 Pictorial Representation shows that Match Region:

This is shown in Table-I (word size 3). Next, let us look at higher matching words.

position	Match position	Total base pair match	Gap	Highest match position without gap	Highest match position with gap
1st position	1-6(1-3 & 4-6)	6	0		
2nd position	8-10	3	1		
3rd position	12-14	3	1		
4th position	17-22	6	2		
5th position	25-36	12	2	25-33	
6th position	38-39	3	1		25-39

Figure-III(word size 3)

2.5 Highest Matching Word:

This is given in Table-II. Next, let us look at the project spectrum.

Words/ colones	Repeat numbers
ATT	3
TTT	5
TAT	1
TGC	1

2.6 Project Spectrum:

• A base program to detect the HCRs in a target sequence for a given viral sequence

• A method to locate the start and end positions of infection and isolate the infected regions.

• A method to identify the longest infected region or the largest HCR.

• An extension to allow all 4 possible transforms of the viral sequence (i.e. DNA, RNA,

cDNA, cRNA).

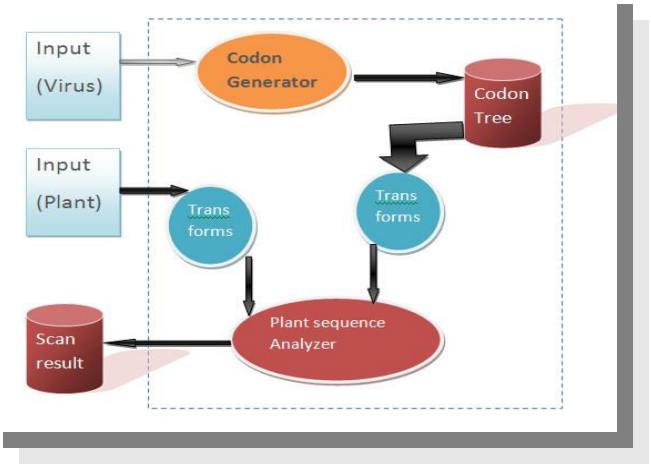
• An extension to allow scanning of all possible transforms of the normal plant (target) sequence. i.e. DNA, RNA, cDNA, cRNA. A total of 4x4 scan orientations.

• An extension to identify successive regions of *Edit Distance = 1*.

• An extension to detect and report all such extrapolated infection regions and locate the largest of them.

2.7 Process Diagram:

This process diagram is shown in Fig. III. Next, let us look at Inputs



2.8 Inputs:

• The Inputs Taken are:

• Normal Plant Sequence:

• A Steam of DNA bases in fasta format, i.e. a text file containing an

DNA sequence.

• Limitations: none.

• Viral Sequence:

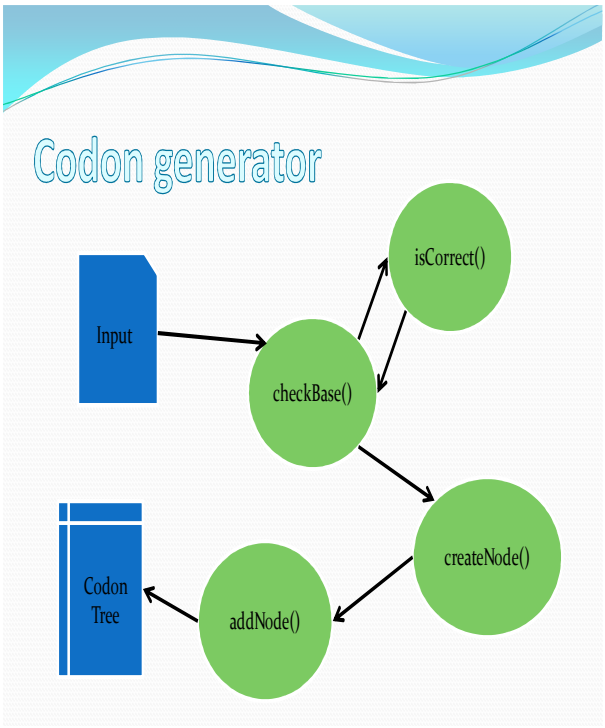
• A Steam of RNA bases in fasta format, i.e. a text file containing an RNA sequence.

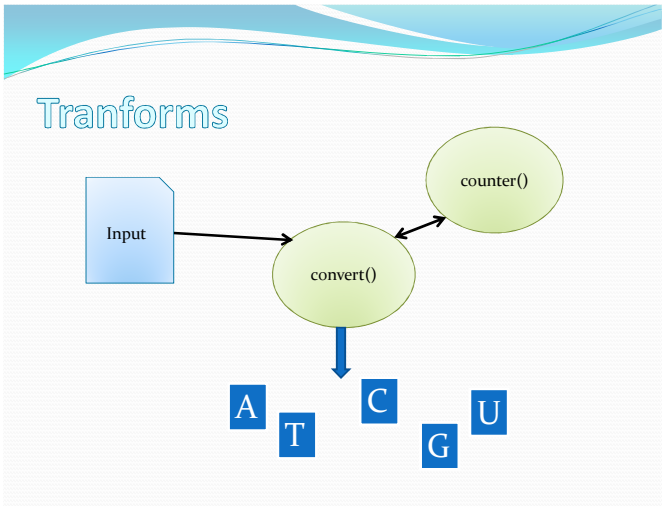
• Limitations: size of file should be less than 400 Kbytes.

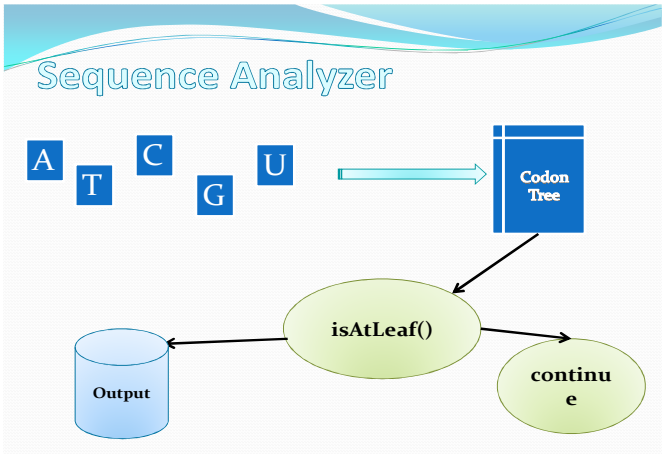
2.9 Codon Generator:

Code Generator is shown in Fig. IV. Next, let us look at the Schema of Node.

2.8 Codon Generator

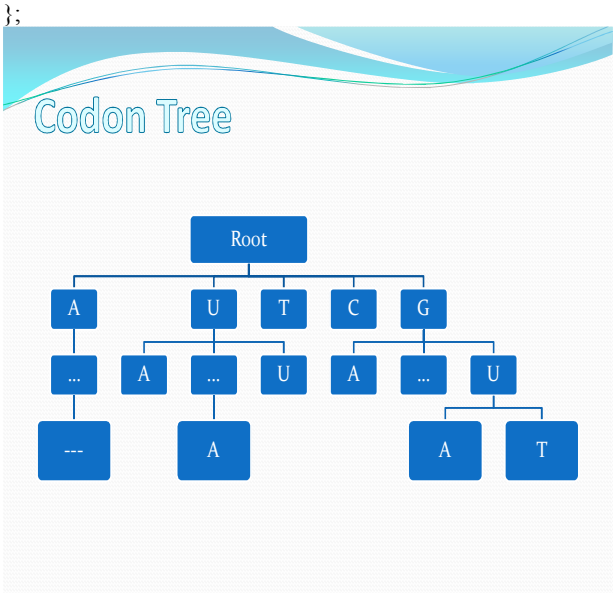






2.10 Schema of 'Node':

```
struct C_NODE
{
int c_no;
C_NODE* A;
C_NODE* C;
C_NODE* G;
C_NODE* U;
int mat;
int ovr;
};
```



Condon Tree, Transforms and sequence Analyzer are given in Fig. V, VI and VII

respectively. Next, let us look at the Outputs.

2.11 Outputs:

- Codon lists of all viral transforms i.e. DNA, RNA etc.
- Start and End Positions of Alignments
- Sequences of the HCRs
- Maximum Length of Alignments
- Successive Alignments with Edit Distance = 1
- Highest Interpolated Alignments
- The most active codon of each type.
- Infectivity of the virus (no. of infecting codons /Total)

3. SPECIFICATIONS FOR SOFTWARE REQUIREMENT:

Hardware Specification:

Processor: Pentium 2.4 Ghz or higher

Memory : 256MB RAM (minimum for desktop engine)

Hard Disk : 1 GB (minimum installation)

2.5 GB (full installation)

Software Specification:

Operating System:

Windows XP, Linux (2.6 kernel & above)

Programming Languages:

C, JAVA, JSP

Databases:
JDBC

4. The ALGORITHM:

VIIRUS CODON DETECTIION ALGORIITHM :

Requirements:

• Any computer running on windows platform.

• A virus codon base file in simple text format.

• A plant / target codon base file in text format.

Special Data Structures needed:

• A Structure **C_NODE** to create a tree of viral codon bases. Every node of the

C_NODE structure contains only one base of the viral codon sequence. This helps to ensure that only the unique codons are scanned in the target genome sequence.

• A class **result** to store the scanning details for the completed scan.

• A structure **time** to compute the total time taken to scan the files.

5. COMPLEXITY:

The algorithm uses an M-ary tree to structure the input sequence and then allows the target to

'pour through' the root and fit in place. Thus the target sequence looks at a match, rather than the other way round. Here M=5 so the time complexity of the program is:

O (n1 log M O (n1 log 5 n2) n2)
O (n1 log 5 n2)
n1 :- size of viral sequence
n2 :- size of plant sequence

6. ANALYSIS:

A comparison of a variant of the same program, using the strcmp() library function yielded
the following timings. This is tabulated in Table-III.

7. PERFORMANCE:

The program was tested with real inputs and the time spent is tabulated in Table-IV.

VIRUS (in Kb)	PLANT (in Kb)	Time taken
< 400 (bytes)	< 5	~ 0.5 milliseconds
500 -1024 (bytes)	< 5	~ 0.5 milliseconds
1-5	< 100	~ 90 milliseconds
1-5	200 - 1024	~ 400 milliseconds
10 - 100	1024 – 5 Mb	~ 1 – 4 Seconds
10 - 100	5 – 7 Mb	~ 5 – 10 Seconds
100 -300	~10	~ 15 – 20 Seconds

Target input with fixed base seq. 349 bytes	Time with strcmp()	Time with this algorithm
200 Kb	200 seconds	25 milliseconds
1 Mb	7 minutes	456 milliseconds
1.5 Mb	15 minutes	1 – 2 second(s)
> 2 Mb	The computer hanged	~15 seconds

8. CONCLUSION:

This algorithm show that virus and normal plant interaction found only in between virus

RNA with normal plant cDNA and RNA stand only. The virus and plant interaction found only in normal in nature, no such other orientation is applicable. The colone size varies from 3 to 9. The lower the subsequence size, the higher the interaction rate. This algorithm also apply on any type of virus and any type of normal plant genome sequences. In future, an attempt will be made to apply this software in real life example such as Potato Spindle Tuber Viroid is infected only chloroplast of the Tomato plant not in their root.

9. ACKNOWLEDGEMENTS:

Above all, author are grateful to all our colleagues for their valuable suggestion, moral support, interest and constructive criticism of this study. The author offer special thanks to Ph.D guides Dr. P.K.Das Mohapatra, Asst. Professor in Vidyasagar University, Midnapur and Dr. S.Basu, Sr.Lecturer in West Bengal University of Technology, Kolkata .also like to thank our PCs.

10. REFERENCES:

[1.] Diener, T.O. (1979) Viroids and Viroid Diseases. Wiley, New York.

[2.] Gross, H.J.. Domdey, H., Lossow, C., Jank, P., Raba, M., Alberty, H., and Sanger, H.L. (1978) Nature 273, 203-208.

[3.] Gross, H.J., Krupp, G., Domdey, H., Raba, M., Jank, P., Lossow, C.,Alberty, H., Ranm, K. and Sanger, H.L. (1982) Eur. J. Biochem. 121,249-257.

[4.] Gross, H.J., Liebl, U., Alberty, H., Krupp, G., Domdey, H., Ramn, K., and Sanger, H.L. (1981) Bioscience Reports 1, 235-241.

[5.] Haseloff, J. and Symons, R.H. (1981) Nucleic Acids Res. 9, 2741-2752.

[6.] Haseloff, J., Mohamed, N.A., and Symons, R.H. (1982) Nature 299, 316-321.

[7.] Symons, R.H. (1981) Nucleic Acids Res. 9, 6527-6537.

[8.] van Wezenbeek, P., Vos, P., van Boom, J. and van Kammen, A. (1982) Nucleic Acids Res. 10, 794-7957.

[9.] Gross, H.J. and Riesner, D. (1980) Angewandte Chemie (English edition) 19, 231-243.

[10.] Diener, T.O. (1981) Proc. Natl. Acad. Sci. USA 78, 5014-5015.

[11.] Dickson, E. (1981) Virology 115, 215-221.

11. AUTHORS PROFILE

Syed Mahamud Hossein received his post graduate degree in Computer Applications in 1998 from Swami Ramanand Teerth Marathawada University[M.Sc.-C.A.], Nanded and Master of Engineering in Information Technology[M.E.-I.T.] in 2005 from West Bengal University of Technology, Kolkata. He has worked as the Senior Lecturer in Haldia Institute of Technology, Haldia and Contractual Lecturer in Panskura Banamali College, Panskura since 1999 to 2009.Now he is working as a Lecturer in Iswar Chandra Vidyasagar Polytechnic, Govt. of West Bengal, Jgargram, presently posted at District Officer, Regional Office, Kolaghat, Directorate of Vocational Educational & Training, West Bengal on deputation basis since 2009 to till date. His research interests includes Bioinformatics, Compression Techniques & cryptography, Design and Analysis of Algorithms, Information Security, & Development of Software Tools. He is a member of professional societies like FOSET and a life members Computer Society of India & Indian Science Congress Association