


Spring 2013

Genome wide search for pseudo knotted non-coding RNAs

Meghana S. Vasavada

New Jersey Institute of Technology

Follow this and additional works at: <https://digitalcommons.njit.edu/theses>

 Part of the [Bioinformatics Commons](#), and the [Computer Sciences Commons](#)

Recommended Citation

Vasavada, Meghana S., "Genome wide search for pseudo knotted non-coding RNAs" (2013). *Theses*. 159.
<https://digitalcommons.njit.edu/theses/159>

This Thesis is brought to you for free and open access by the Theses and Dissertations at Digital Commons @ NJIT. It has been accepted for inclusion in Theses by an authorized administrator of Digital Commons @ NJIT. For more information, please contact digitalcommons@njit.edu.

Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be “used for any purpose other than private study, scholarship, or research.” If a user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of “fair use” that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select “Pages from: first page # to: last page #” on the print dialog screen

The Van Houten library has removed some of the personal information and all signatures from the approval page and biographical sketches of theses and dissertations in order to protect the identity of NJIT graduates and faculty.

ABSTRACT

GENOME WIDE SEARCH FOR PSEUDO KNOTTED NON-CODING RNAs

by
Meghana S. Vasavada

Non-coding RNAs (ncRNAs) are the functional RNA molecules that are involved in many biological processes including gene regulation, chromosome replication and RNA modification. Searching genomes using computational methods has become an important asset for prediction and annotation of ncRNAs. To annotate an individual genome for a specific family of ncRNAs, a computational tool is interpreted to scan through the genome and align its sequence segments to some structure model for the ncRNA family. With the recent advances in detecting an ncRNA in the genome, heuristic techniques are designed to perform an accurate search and sequence–structure alignment. This study uses a novel approach for such genome wide search of ncRNAs using the RNATOPS and Infernal software tools, which incorporates heuristic dynamic programming algorithms to carry out the sequence analysis using the profiles of RNA consensus secondary structures.

Genome wide search for ncRNAs from thirteen genomes is performed using RNATOPS and Infernal. The training set of ncRNA multiple sequence alignments is prepared from RFAM and homologous Genomes are retrieved from RNASRAND database. Through the experiments, performance of each tool is analyzed and compared with respect to their ncRNA search accuracies. It is further interfered that Infernal, compared to RNATOPS, is more accurate in detecting an ncRNA in all the thirteen genomes tested.

GENOME WIDE SEARCH FOR PSEUDO KNOTTED NON-CODING RNAs

By

Meghana S. Vasavada

**A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Bioinformatics**

Department of Computer Science

May 2013

Copyright © 2013 by Meghana S. Vasavada

ALL RIGHTS RESERVED

APPROVAL PAGE

GENOME WIDE SEARCH FOR PSEUDO KNOTTED NON-CODING RNAs

Meghana S. Vasavada

Dr. Jason T.L. Wang, Thesis Advisor Date
Professor of Bioinformatics and Computer Science, NJIT

Dr. Zhi Wei, Committee Member Date
Assistant Professor, Department of Computer Science, NJIT

Dr. Mei Liu, Committee Member Date
Assistant Professor, Department of Computer Science, NJIT

Dr. Usman Roshan, Committee Member Date
Associate Professor, Department of Computer Science, NJIT

BIOGRAPHICAL SKETCH

Author: Meghana S. Vasavada

Degree: Masters of Science

Date: May 2013

Undergraduate and Graduate Education:

- Master of Science in Bioinformatics
New Jersey Institute of Technology, Newark, NJ, 2013
- Master of Technology in Bioinformatics,
Dr. D.Y. Patil University, Mumbai, India, 2010
- Bachelor of Science in Life Sciences and Biochemistry,
University of Mumbai, Mumbai, India, 2008

Major: Bioinformatics

Presentations and Publications:

Kamath KS, Vasavada MS, Srivastava S. Review: Proteomic databases and tools to decipher post-translational modifications. *J Proteomics* 2011;75(1):127-44

I dedicate this thesis to my parents who have always stood by me and presented me an opportunity to pursue education from one of the best institutions

ACKNOWLEDGMENT

It is with immense gratitude that I acknowledge the help and valuable guidance of my thesis advisor, Dr. Jason Wang, Professor of Bioinformatics and Computer Science, NJIT, and the committee members, who not only guided all through the work but also kept my spirit high with his valuable suggestions and constant encouragement.

I am indebted to my colleagues especially Yang Song and Kevin Byron for their suggestions and kind support throughout. Also I am thankful to the authors of RNATOPS and Infernal who made their software packages publicly available to use and helped me thoroughly understand their algorithms.

I hereby conclude my acknowledgement by regarding my deep sense of affection to my beloved family members for their encouragement through this endeavor. This thesis would have remained a dream had it not been for the love and encouragement of my parents along with guidance of my thesis advisor.

TABLE OF CONTENTS

Chapter	Page
1 INTRODUCTION	1
1.1 Motivation	1
1.2 Research Objectives and Overview	2
2 LITERATURE REVIEW	3
2.1 Non-coding RNAs and the Secondary Structure	3
2.2 ncRNA search and Alignment tools	6
2.2.1 RNATOPS Algorithm	9
2.2.2 Algorithm used by Infernal	16
2.2.3 Accelerated ncRNA search	38
3 METHODS	45
3.1 Preparation of Training Set	45
3.2 Genome Selection for test	48
3.3 Test with RNATOPS	51
3.4 ncRNA search using Infernal	55
3.4.1 Building a model using cmbuild	55
3.4.2 Searching the CM against a sequence database using cmsearch	58
4 EXPERIMENTAL RESULTS AND INTERPRETATION	60
4.1 ncRNA search by RNATOPS	60
4.2 ncRNA search by Infernal	66

TABLE OF CONTENTS
(continued)

Chapter	Page
5 DISCUSSION	73
6 CONCLUSIONS	86
APPENDIX A STOCKHOLM FORMAT OF THE TRAINING SET	77
APPENDIX B OUTPUTS GIVEN BY RNATOPS TOOL	82
APPENDIX C COVARIANCE MODEL OF IRES CRIPAVIRUS FAMILY	93
APPENDIX D INFERNAL ORIGINAL OUTPUTS WITH ALIGNMENTS	98
APPENDIX E PERL SCRIPT TO CALCULATE THE MISALIGNMENTS	117
REFERENCES	119

LIST OF TABLES

Table		Page
2.1	Seven State Types of a Covariance Model	17
3.1	ncRNA Families of Rfam	46
3.2	Genome belonging to ncRNA Families	50
4.1a	Result summary of ncRNA search with RNATOPS and Infernal	71
4.1b	Result summary of ncRNA search with RNATOPS and Infernal	72

LIST OF FIGURES

Figure		Page
2.1	ncRNA Pseudoknotted Secondary Structure	5
2.2	A typical Pseudo-knotted Structured Graph	10
2.3	An Example of a Profile HMM derived from any Multiple Alignment	10
2.4	Structure graph for the Second Pseudoknot of the Consensus structure of Bacterial tmRNAs	12
2.5	Case 1- Tree Decomposition for Subgraph H_t^s	13
2.6	Case 2- Tree Decomposition for Subgraph H_t^s	14
2.7	Case 3- Tree Decomposition for Subgraph H_t^s	14
2.8	Tree Decomposition for the Sample Structural Graph	15
2.9	An Example of RNA Sequence Family	16
2.10	Guide Tree from the Structural Alignment	19
2.11	CM of the RNA Alignment from a Guide Tree	21
2.12	Parse Trees shown for Two Sequences-Structures from the Multiple Alignment and the CM	22
2.13a	Representation of the Model used while carrying out the Inside Algorithm ...	23
2.13b	CM of the Alignment with Transition Probabilities	24
2.14	Matrices filled for all states from End to State 1 at the Initialization Step	25
2.15	Matrices from End state to first state of Inside Algorithm Recursion	27
2.16	Matrices in CYK Algorithm	33
2.17	Recursion Matrix of End State	34

LIST OF FIGURES
(Continued)

Figure	Page
2.18 Representation of Recursion matrices of four states	35
2.19 An Example showing a Hit in a CYK	36
3.1 A Web Server of the Rfam Database	45
3.2 Training Sequences in Stockholm Format	47
3.3 RNASTRAND v2.0 Database Home Page	48
3.4 FASTA Formatted Sequence of IRES of the Organism <i>S.cerevisiae</i>	49
3.5 BLAST Results	49
3.6 Optimum Alignment of the Highly Identical Genome Sequence with the Query RNA Sequence	49
3.7 University of Georgia RNATOPS-W Home Page	51
3.8 RNATOPS-W Open Access Web Server	52
3.9a Pasta Format of the Multiple Alignment in the RNApasta tool	53
3.9b The Pairing Structure and RNA Multiple Alignment in Pasta format	54
3.10 cmbuild code for IRES Cripavirus Family used in Infernal	56
3.11 CM of IRES Cripavirus ncRNA built by the cmbuild Program	57
4.1 RNATOPS Result for Search of IRES Cripavirus against Cricket Paralysis Virus Genome	60
4.2 RNATOPS Output for the ncRNA Search of RNaseP arch ncRNA against the Genome <i>Methanocaldococcus jannaschii</i> DSM 2661 of Organism <i>Methanocaldococcus jannaschii</i>	64
4.3a Infernal Output given for the Genome Cricket Paralysis virus of Organism <i>S. cerevisiae</i> tested for ncRNA from family IRES Cripavirus	66

LIST OF FIGURES
(Continued)

Figure		Page
4.3b	Consensus Sequence and Structure of Infernal aligned to that of the Genome Cricket Paralysis Virus Nonstructural Polyprotein	69
5.1	ncRNA Multiple Alignment of IRES Cripavirus Family in Stockholm Format	74

LIST OF ACRONYMS

ncRNA	Non-coding RNA
CM	Covariance Model
CYK	Cocke–Younger–Kasami
QDB	Query Dependent Banding
PDB	Protein Database
HMM	Hidden Markov model
BLAST	Basic Local alignment Search Tool
SD	Standard deviation
NP	non-deterministic polynomial-time

CHAPTER 1

INTRODUCTION

1.1 Motivation

Since the non-coding RNAs (ncRNAs) are known to be involved in many biological processes, searching genomes for these ncRNAs by their secondary structure has become an important goal for Bioinformatics. When the secondary structure is pseudoknot-free, the ncRNA search becomes more effective when it applies to covariance model and CYK-type dynamic programming, which use heuristic techniques of probabilistic modeling to describe the secondary sequence of an RNA, and to find the similar matches to a target RNA within the databases.

However, the difficulty arises while aligning an RNA sequence to a pseudoknot. The structural configuration of a pseudoknot consists of the base pairs that overlap one another in sequence position, which makes its presence in RNA sequences more difficult to predict by the standard dynamic programming methods. It has been interpreted that, the overlapping nature of pseudoknots prohibits the fast and accurate search. According to the study in the previous researches over the past few years RNATOPS and Infernal are the two among the significant tools used for ncRNA search which scan the genomic sequence to detect a pseudo-knotted secondary RNA structure from its multiple alignment profile, thereby producing a structural alignment with a genome with different speed and an RNA detecting capacity possessed by both of these tools.

It has therefore become essential to compare the significant ncRNA search tools, in order to analyze the speed and accuracy of an RNA alignment to a genomic sequence.

1.2 Research Objective and Overview

The main objectives of this thesis are to provide initially, a review describing an ncRNA and its pseudo-knotted secondary structure, survey of different ncRNA search tools and finally, an emphasis on comparing pseudo-knotted ncRNA search tools, namely RNATOPS and Infernal.

The first part of the objective will deal with the background information of an ncRNA, its pseudo-knotted secondary structural configuration and review regarding the survey of different software tools for a genomic ncRNA search. The second and the final part of the objective will be accomplished by describing the heuristics techniques of the dynamic programming algorithms used by RNATOPS and Infernal software tools, performing a genome wide search for a sample of six different ncRNAs from Rfam, against the sample homologous genomes of thirteen different organisms each, using RNATOPS and Infernal, and finally emphasizing on the comparative analysis of the their ncRNA detecting accuracies.

CHAPTER 2

BACKGROUND AND LITERATURE REVIEW

2.1 Non-coding RNAs and the Secondary Structure

A non-coding RNA is considered to be a functional RNA which is not translated into proteins. The DNA sequence from which a non-coding RNA is transcribed is called an RNA gene. The ncRNA gene includes highly abundant and functionally important RNAs namely transfer RNA (tRNA), ribosomal RNA (rRNA), messenger RNA (mRNA) transfer-messenger RNA (tm-RNA), small nuclear RNA (snRNA), Ribonuclease P (RNaseP), telomerase RNA, MicroRNA etc.

tRNA, rRNA, snRNA and RNaseP are few of the ncRNAs involved in translation mechanism. Ribosomal RNAs, present in ribosomal part of the cell are known to catalyse the translation of nucleotide sequences to protein. tRNAs, form an 'adaptor molecule' between mRNA and protein. RNase MRP is another set of ncRNAs which is restricted to eukaryotes and are involved in the maturation of rRNA. RNaseP is a ubiquitous RNA, which is involved in maturation of tRNA sequences by generating mature 5'-ends of tRNAs through cleaving the 5'-leader elements of precursor-tRNAs. RNaseP is also known to influence gene expression by efficient transcription of various ncRNAs transcribed by RNA polymerase III. U1, U2, U4, U5, and U6 are the ncRNA components of the major spliceosome. The ncRNA components of the minor spliceosome namely U11, U12, U5, U4atac and U6atac, along with the major ncRNA components are involved in RNA splicing in Eukarotes forming the mature RNAs. Group I catalytic intron and group II catalytic intron are self-splicing ncRNAs which catalyze their own

excision from mRNA, tRNA and rRNA precursors in a wide range of organisms. The SnoRNAs found in mammals are another set of ncRNA which regulate alternative splicing of mRNA. SmY ncRNA found in nematodes are apparently involved in mRNA trans-slicing. Internal ribosome entry sites (IRES) are among the cis-acting ncRNAs which has RNA structure that allow for translation initiation in the middle of a mRNA sequence as part of the process of protein synthesis. Few of the ncRNAs are actively responsible for producing disease. miRNAs, long mRNA-like ncRNAs, telomerase RNA and Y RNAs are the set of ncRNAs show abnormal expression patterns in cancerous tissues. Mutations with ncRNAs namely snoRNA, snRNA, RNase MRP and the antisense RNA, BACE1-AS are responsible for Prader–Willi syndrome, Autism, Cartilage-hair hypoplasia and Alzheimer's disease respectively. ncRNAs is also known to regulate p53 expression.

Pseudoknot is considered to be an important structural motif in secondary structures of many types of ncRNAs that perform their functions through both their sequences and secondary structures, defined by the interacting base pairs namely Guanine (G)-Cytosine (C) and Adenine (A)-Uracil(U) forming three covalently hydrogen bonded and two hydrogen bonded base pairs respectively. Base pairs in an RNA structure are approximately coplanar and are stacked into other base pairs and such contiguous stacked base pairs are called *stems*. Single stranded subsequences bounded by base pairs are called *loops* and the loop which is formed at the end of a stem is called a stem loop or *hairpin loop*. Single stranded bases occurring within a stem are called a bulge or bulge loop if the single stranded bases are on one side of the stem and an interior loop if there

are single stranded bases interrupting both sides of the stem. Finally, there are multi-branched loops from which three or more stems radiate.

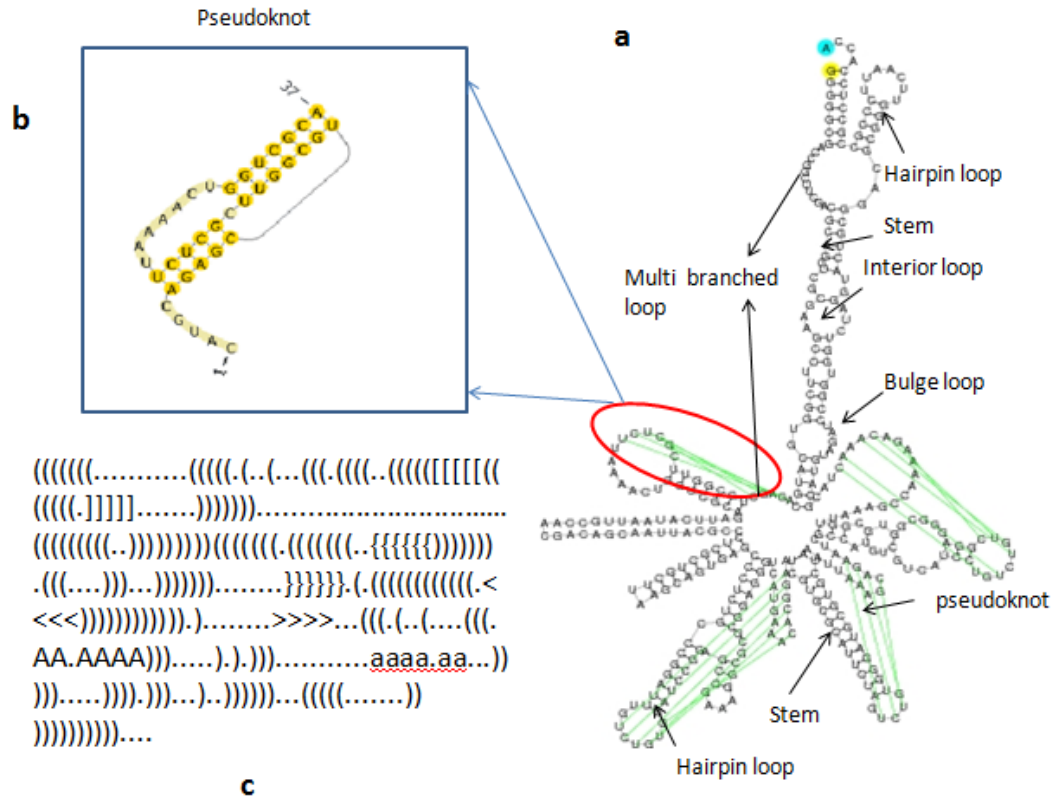


Figure 2.1 ncRNA Pseudoknotted Secondary Structure. a. Structure of a Transfer messenger RNA (TMR_00008) from RNASTRAND database with all the secondary structure features. b. the magnified structure of one of the pseudo knots from the secondary structure (a) obtained from Pseudoviewer. c. The dot-parenthesis format of a secondary structure (a)

Base pairs occur in the nested fashion in the RNA secondary structure. Precisely some RNA molecules appear to fold into pseudoknot structures pairing bases in loop regions with bases outside the stem loop as shown in Figure 2.1 b. ncRNA secondary structure can be represented in two ways, arc-based and an extended dot-bracket notation respectively. In the arc-based representation nucleotides and hydrogen bonds are represented by vertices and arcs, respectively, for pseudoknot-free secondary structures,

all arcs are either nested or in parallel whereas for pseudo knotted structures Crossover arcs indicate pseudoknots. In the dot-bracket notation '.' represents unpaired bases and matching parenthesis '(' and ')' or '<' and '>' indicate base-pairing nucleotides. The base-pairing nucleotides forming pseudoknots are represented by upper-lower case character pairs, such as A..a or B..b as shown in Figure 2.1 c.

2.2 ncRNA Search and Alignment Tools

According to the study in previous work, searching genomes using computational methods has become essential for annotation of ncRNAs. The ncRNA search was performed using the programs namely RNATOPS, Infernal and FastR [4]. FastR is an efficient database search tool for ncRNA which when given an RNA sequence with known secondary structure, efficiently compute all structural homologs (computed as a function of both sequence and structural similarity) in a genomic database. RNATOPS is known to be a profile based RNA structure search program that can detect RNA pseudoknots in genomes. (www.uga.edu/RNA-Informatics/?f=software&p=RNATOPS-w). Infernal (INFERENCE of RNA Alignment), written and maintained by the Sean Eddy laboratory at Janelia Farm (infernal.janelia.org) is a software package for searching DNA sequence databases for RNA structure and sequence similarities. Infernal appears to be primarily concerned with the functional ncRNAs. 43 bacterial genomes were searched for the ncRNA from Bacterial tmRNA, 7 genomes for RNaseP bacterial type B, Yeast telomerase RNAs and bacterial 16S rRNA by Zhibin Huang and the coauthors in their research. RNATOPS uses non-conventional tree decomposition-based dynamic programming to detect the components of pseudo-knotted ncRNA [4].

The known ncRNA search is based on context-free grammar (CFG), by which the pseudoknots cannot be effectively modeled. Commonly used known ncRNA search tools such as Infernal, RSEARCH and tRNAScan-SE are all based on Stochastic Context free grammars (SCFG). To build a model of consensus RNA secondary structure INFERNAL using a formalism called a covariance model (CM), which is a type of profile stochastic context-free grammar (profile SCFG) [1,5]. The context-free grammar is a transformational grammars, also known as theory of modeling strings of symbols according to Chomsky. The grammar create nested, long distance pair wise correlation between terminals. Context-free grammars possess the production of the form $W \rightarrow \beta$, where W in the left hand side is a nonterminal and the β in the right hand side of the production rule can be any combinations of terminal and nonterminal. Each grammar possesses a corresponding abstract computational device called automaton which are apparently the parsers that accept or reject a given sequence. In a stochastic grammar (regular or context free) the sum of the probabilities of all the possible production from any given nonterminal is supposed to be 1. The resulting stochastic grammar defines a probability distribution over sequences s represented as $\sum_s P(s|\theta)=1$. For example if $A \rightarrow pX_1|qX_1$, a stochastic grammar is likely to assign the probabilities of 0.5 to the productions $A \rightarrow pX_1$ and $A \rightarrow qX_1$. The stochastic regular grammars emit a terminal on transition to a new non terminal, productions being of the form $P_1 \rightarrow aP_2$. It has been interpreted that Hidden Markov models (HMMs) are equivalent to stochastic regular grammars in which a terminal is emitted on transition to a new non terminal and the production rule is of the form $W_1 \rightarrow aW_2$. HMMs emit symbols on a state independent of transitions. For instance, any HMM state which makes N transitions to a new state that

each emit one of M symbols can also be modeled by a set of NM stochastic regular grammar production [5]. Stochastic regular grammars possess a Finite state automaton which reads one symbol at a time from an input string. When symbol is accepted the automaton moves to the new state or else leaves the string if rejected. At the final state the string is successfully recognized and parsed. Stochastic Context free grammars (SCFG) uses the Chomsky normal form for which the production rules are of the form $W_v \rightarrow W_y W_z$ or $W_v \rightarrow a$. SCFGs possess the parsing automaton for called push-down automaton which parses the sequence from left to right keeping the limited memory of symbols in the form of a push-down stack, which is an array or list accessed last-in-first-out, where the elements(symbols) are pushed onto and popped off of the top of the stack. The automaton's stack is initialized by pushing the start non terminal onto it, the following steps are iterated until no input symbols remain and the stack gets empty then the sequence is successfully parsed [5]. The Covariance model (CM) implements the rules of SCFGs in inside-outside algorithm as Chomsky normal form in which the inside algorithm calculate the probability score of a sequence on a given SCFG and the best path variant of the inside algorithm called Cocke-Younger-Kasami (CYK) algorithm finds the maximum probability alignment of the SCFG to the sequence. Inside-outside is a recursive dynamic algorithm and the computational complexity of inside-outside algorithm is substantially greater compared to the one used by HMM [5].

2.2.1 RNATOPS Algorithm

RNATOPS-W, a web server to search sequences for RNA secondary structures containing pseudoknots and was built upon RNATOPS (RNA via Tree decomposition), which is known to be a profile based RNA structure search program. The RNATOPS algorithm is designed based on the following four approaches

- a. Preparation of a graph model for structure search
 - b. Model training
 - c. Stem candidates identification
 - d. Tree decomposition
- a. Preparation of a graph model for structure search

The model was constructed base on structure graphs which specified the consensus structure of an RNA family consisting all the structural units such as stems and loops. The vertex in the graph defined either base pairing regions of a stem with two vertices representing two complementary regions which formed a stem connected with a non directed edge from 5' to 3' direction. Individual helices and loops were modeled with a restricted Covariance Model and profile HMM, respectively. The complete graph was capable of modeling RNA structures consisting of multiple interactions among nucleotides, which also contain pseudoknots [4].

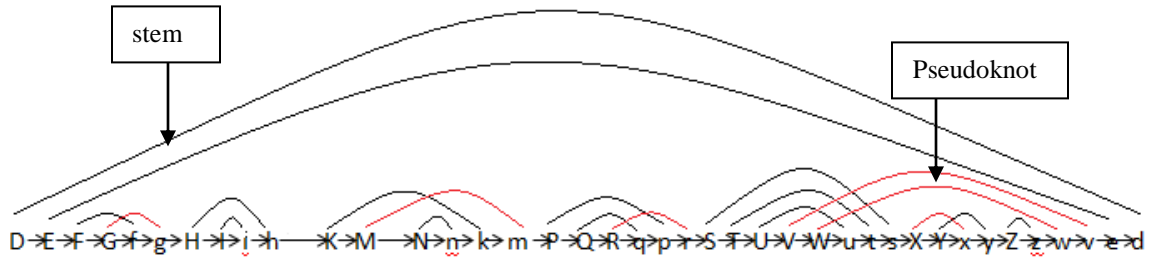


Figure 2.2 A Typical Pseudo-knotted Structured Graph. Arc linking the upper and lower cases of the same letter denotes the stem and the red arcs indicate the stems causing pseudoknot.

Source: [4]

b. Model training

The Model built from graph containing fold topology, helices, and loops were trained with an input pasta file that contained a multiple structural alignment for a set of training RNA sequences (<http://rna-informatics.uga.edu/?f=software&p=RNATOPS>). One profile HMM is generated from every two neighboring base regions which further allow possible match, insertion and deletion states in every column of the multiple alignment [4].

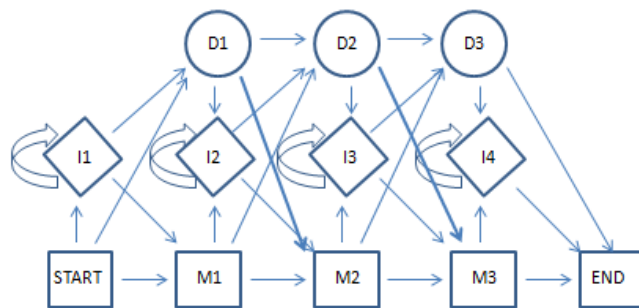


Figure 2.3 An Example of a Profile HMM derived from any Multiple Alignment. Emission probabilities are associated with the residues in any state. Transition probabilities are indicated by thickness of the arrows. The diamonds, circles and squares indicate the insert, delete and match states, respectively, including the Start and End state. The insert states are self-transitional.

Source: [5]

The maximum likelihood method was then used to compute the parameters from the multiple structural alignment. Pseudocounts, which adds constants to all counts preventing from zero probabilities, were considered for nucleotides in the match, insertion and deletion states of the profile HMM. A probability matrix $P(\text{bp})$ for base pairs were constructed and a weighting parameter was introduced for a CM to define the probability of a base pair. When P_m is the probability matrix and w is the weighted parameter, the probability of base pairs $P(x,y)$ was calculated as the weighted sum $wP_{\text{bpt}}(x,y)+(1-w)P_m(x,y)$, where P_{bpt} is the base pair probability matrix obtained from the training set [4].

b. Identifying stem candidates.

The sliding window approximately of the size of target genome was considered to search for stem candidates in a target genome. For every given CM consensus stem, the score of every possible structural motif aligned to the model was computed within the window. A dynamic programming algorithm was used to find the candidates. The heuristic techniques were developed to ensure the correct motif structure for the CM if existing in the sequence was likely to be among the top k candidates for small values of k . According to the statistical distribution of the consensus stem in the training sequences regions from the selected candidates were considered to be constrained. Through the probabilistic Gaussian distribution function given by the equation,

$$f_g(x) = 1/\sqrt{(2\pi\sigma^2)} * e^{-(x-m)^2/2\sigma^2}$$

the position of the consensus stem in the RNA structure was determined and the constrained region for the correct motif of this consensus stem was observed to lie within the Standard Deviation (σ) of the mean position(m). The candidates so identified were

then ranked again according to statistical distributions of various length parameters associated with a consensus stem which includes the length of the stem, the distance between the two stem arms. The scores of every possible motif candidate ‘c’ of the model were recalculated. The structural motifs which are heavily overlapping in their positions were likely to have higher alignment score with respect to the stem model [4].

c. Tree decomposition for structure graphs

It was found that optimum tree decomposition method with the smallest tree width in almost all ncRNA structural graphs was NP-hard (non-deterministic polynomial-time hard). Zhibin Huang and the coauthors had developed a linear greedy algorithm that yielded tree decomposition of tree width almost always bounded by four. The algorithm involved removal of arc crossing the most other arcs (the pseudoknots), and repeating the step on the remaining graph until there is no crossing arc in the graph [4].

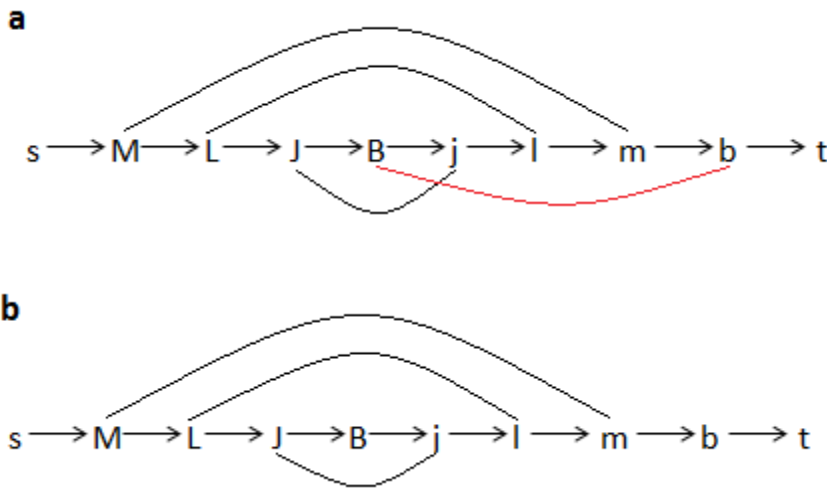


Figure 2.4 a. Structure Graph for the Second Pseudoknot of the Consensus Structure of Bacterial tmRNAs. The vertices s and t are added for technical purposes. **b.** The structure graph in (a) after removing arc (B, b) that crosses with the most other arcs
Source: [4]

Initialization: The vertices are arranged from direction 5' to 3' with the leftmost vertex being source s and rightmost t . The decomposition of subgraph given by notation H_t^s follows the recursive process and terminates when the considered subgraph is empty.

Recursion: (s,X) is a directed edge and (X,x) is an arch forming a stem. $\{s,t\}$ is a root node which has a child node $\{s,x,t\}$, which in turn has child node $\{s,X,x\}$, which will further be the root for the subtree generated from the subgraph H_x^X . and the node $\{s,x,t\}$ will be the root for the subtree generated from the subgraph H_t^x [4].

Case1: Recursive case for H_t^s when (x,t) is not a directed edge and (X,x) is an arc

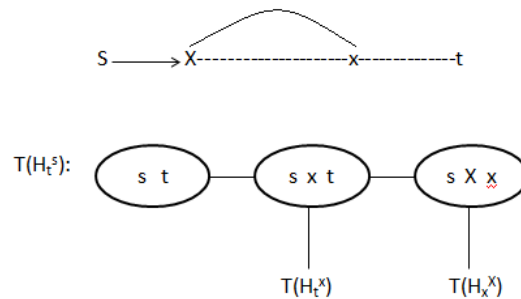


Figure 2.5 Case 1- Tree Decomposition for Subgraph H_t^s .
Source: [4]

When (s,X) and (x,t) both are directed edge and (X,x) is an arc then the root $\{s,t\}$ has child node $\{s,X,t\}$, which in turn has child node $\{X,x,t\}$. Node $\{X,x,t\}$ will be the root for the subtree generated from the subgraph H_x^X .

Case 2: Recursive case for H_t^s when $\{x,t\}$ is a directed edge and $\{X,x\}$ is an arc:-

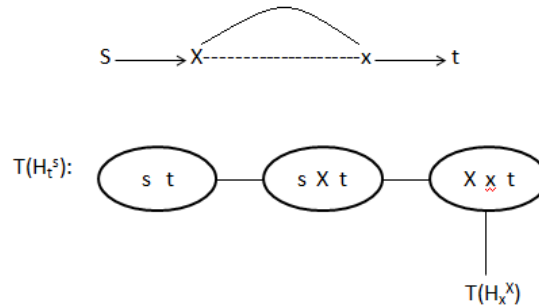


Figure 2.6 Case 2- Tree Decomposition for Subgraph H_t^s
Source: [4]

When (s,X) is a directed edge but (X,x) is an arc then the root $\{s,t\}$ has child node $\{s,X,t\}$, which in turn will be the root for the subtree generated from the subgraph H_t^X .

Case 3:- Recursive case for H_t^s when $\{x,t\}$ is not a directed edge and $\{X,x\}$ is not an arc.

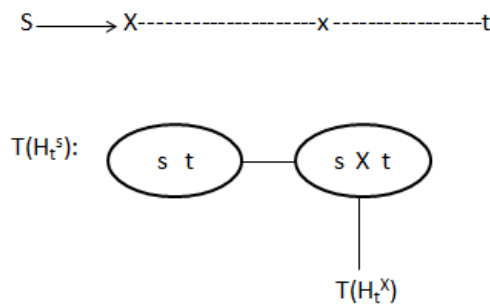


Figure 2.7 Case 3- Tree Decomposition for Subgraph H_t^s
Source: [4]

Eventually the tree decomposition is modified by the algorithm in a following way. For every removed arc the algorithm identifies two nodes, one containing vertex V and another containing its counterpart v , which is further added to latter node from the former node on the path for every tree node on the path.

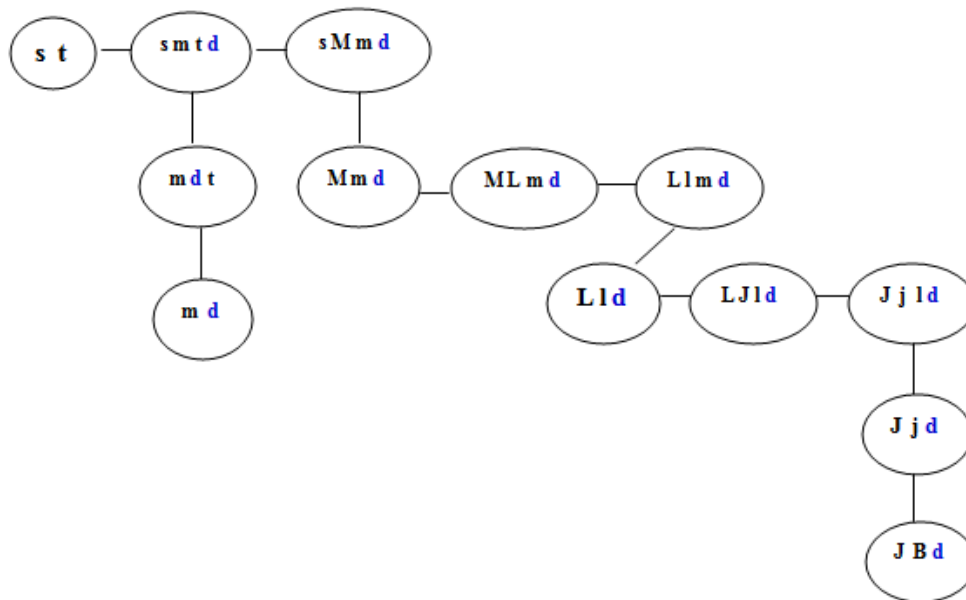


Figure 2.8 Tree Decomposition for the Sample Structural Graph of Figure 2.3(a). First the specialized tree decomposition algorithm is applied on the graph in Figure 2.3b and the vertex 'd' is added to every tree node on the path from the node containing d to the node containing B. Node {s, t} is the root and the tree width is 3.

Source: [4]

It was observed that RNATOPS also executes the whole structure search on filtering result or filtered sequences (i.e. substructure profiles or subsequences) RNATOPS-W server possessed a built-in function for automatic hidden Markov model (HMM) filter selection where the selected filter was used to speed up the search program. It was known that the filter selection chooses a conserved region as an HMM filter from the given RNA structural profile (a set of structurally aligned RNA sequences).

2.2.2 Algorithm used by Infernal

It has been proved that the CMs specify a repetitive tree-like SCFG architecture consisting of groups of model states that are associated with base pairs and single-stranded positions in an RNA secondary structure consensus. Infernal tool implements a heuristic technique which involves building of co-variance models.

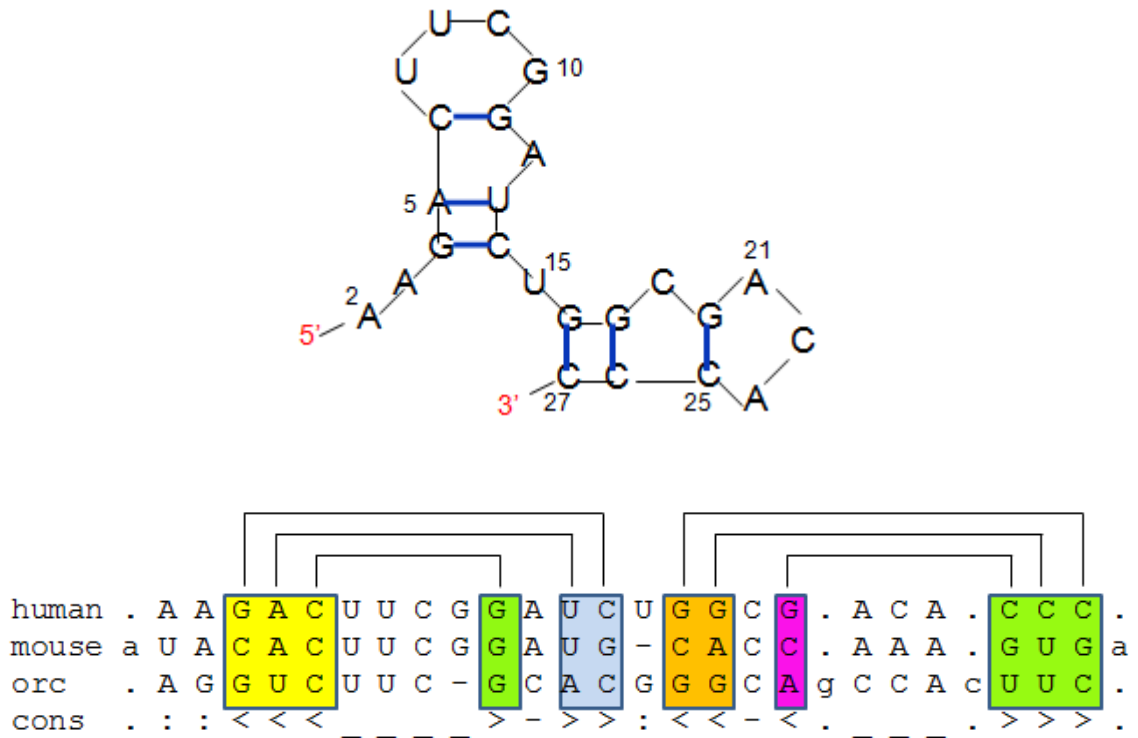


Figure 2.9 An Example of RNA Sequence Family. Above: structure of a human ncRNA. Below: sample multiple alignment of three sequences, with 28 total states (columns), 24 of which will be modeled as consensus positions. The cons line annotates the consensus secondary structure in WUSS notation. Below: the secondary structure of the “human” sequence. Each column of seed alignment corresponds to column of matrix. Source: [5,6]

A CM has seven types of states and production rules. For instance base-paired columns modeled by ‘pairwise’ emitting non terminals, single stranded columns modeled by ‘leftwise’(5’) emitting nonterminal, ‘rightwise’ (3’) nonterminal for the bulges and interior loops on the 3’ end of the stem, ‘bifurcation nonterminal to split into multiple stems and multi-branch loops. a ‘start’ as initial non terminal and also can be as the immediate children produced from a bifurcation and a special ‘end’ non terminal which terminates a derivation. Pairwise productions would have a total of 16 productions and production probabilities for 16 possible pairs, whereas leftwise and rightwise would have four. The model of a consensus structure has a total of 24 nonterminals, modeling a 24 nucleotide RNA alignment with 28 different states consisting of insertions and deletions.

Table 2.1 Seven State Types of a Covariance Model.

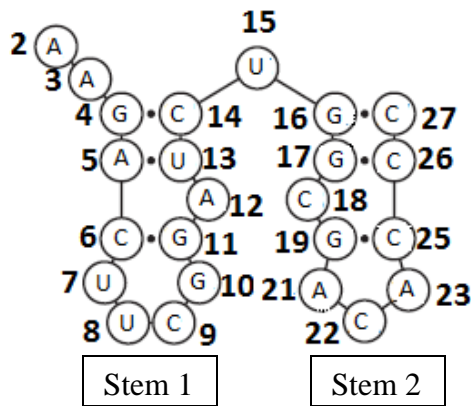
State (S _v)	Description	Production	Δ_v^L	Δ_v^R	Emission	Transition
P	Pairwise	$W_v \rightarrow x_i W_y x_j$	1	1	$e_v = (x_i, x_j)$	$t_v = (y)$
L	Leftwise	$W_v \rightarrow x_i W_y$	1	0	$e_v = (x_i)$	$t_v = (y)$
R	Rightwise	$W_v \rightarrow W_y x_j$	0	1	$e_v = (x_j)$	$t_v = (y)$
D	Delete	$W_v \rightarrow W_y$	0	0	1	$t_v = (y)$
S	Start	$W_v \rightarrow W_y$	0	0	1	$t_v = (y)$
B	Bifurcation	$W_v \rightarrow W_y W_z$	0	0	1	1
E	End	$W_v \rightarrow \epsilon$	0	0	1	1

A covariance model composed of M different states (nonterminals) is denoted as W_1, \dots, W_M . v, y and z are the indices for States W_v, W_y and W_z . These seven state types are associated with symbol emission and state transition probabilities. Each overall production probability is considered to be the independent product of an emission

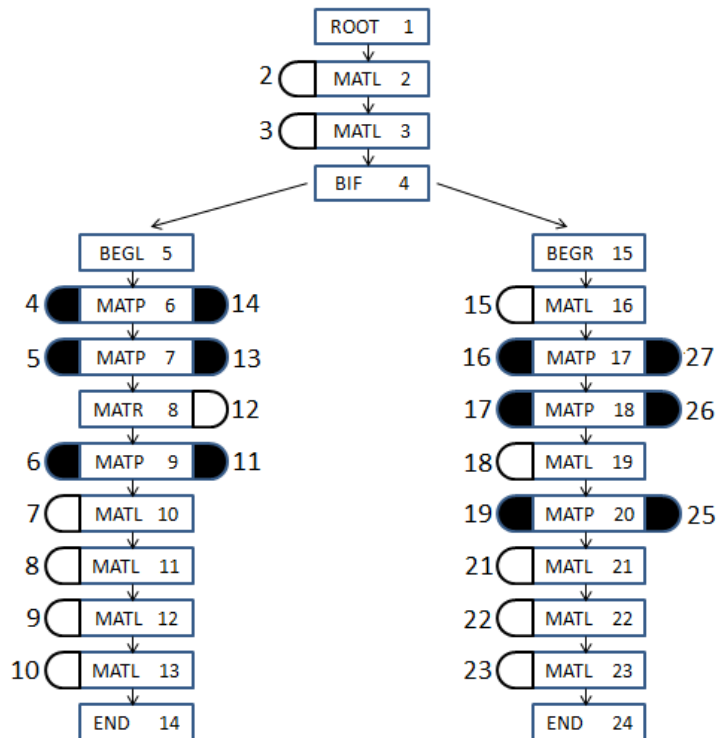
probability e_v and a transition probability t_v , both of which are position-dependent parameters that depend on the state v . For instance, a particular pair (P) state v produces x_i and x_j with probability $e_v(x_i, x_j)$ and transits to one of several possible new states y of various types with probability $t_v(y)$. A bifurcation (B) state splits into two new start (S) states with probability 1. The E state production terminates a derivation. [5].

Construction of a guide tree from consensus structure alignment:

The first step of building a CM is to produce a binary guide tree of nodes representing the consensus secondary structure. The guide tree is a parse tree for the consensus structure, with nodes as nonterminals and alignment columns as terminals. The guide tree deals with the consensus structure including the insertions and deletions for individual sequences. The guide tree, which is considered to be a skeleton for organizing a CM, has eight types of nodes each corresponding closely with the CM's final state types. An MATP node will contain a P-type state to model a consensus base pair. However in some cases it will also contain several other states to model infrequent insertions and deletions at or adjacent to this pair, MATL and MATR nodes containing L and R type states respectively to model single strands on the left and right side respectively. For a given consensus structure a consensus unpaired columns are assigned to MATL and MATR nodes. One ROOT node corresponding to the CM's S-type state is used at the head of the tree. Multiple stems are dealt with by assigning one or more BIF nodes that branch to sub trees starting with BEGL or BEGR head nodes. ROOT, BEGL, and BEGR start nodes are labeled differently because they will be expanded to different groups of states to avoid ambiguity.



A



B

Figure 2.10 Guide Tree from the Structural Alignment. A: the consensus secondary structure derived from the annotated alignment in Figure 2.8. Numbers in indicate alignment column coordinates: e.g. column 4 base pairs with column 14, and so on. B: guide tree corresponding to this consensus structure. The nodes of the tree are numbered 1 to 24 in preorder traversal. MATP, MATL, and MATR nodes are associated with the columns they generate: e.g., node 6 is a MATP (pair) node that is associated with the base-paired columns 4 and 14.

Source: [5]

Construction of a CM model from RNA alignment and guide tree.

As mentioned before CM must deal with insertions and deletions in individual sequences relative to the consensus structure. Each node in the master tree is expanded into one or more states in the CM and the next pair in the stem. When pairwise nodes expand, they have several insertions and deletions possibilities. A deletion may remove both bases in the base pair or solely 5' or 3' partner leaving the remaining unpaired partner as a bulge. Insertions in the base paired stem may occur on the 5' side of the pair, 3' side or both. For instance, node MATP is expanded into total six states, namely an MP, D (for complete deletion of the base pair), ML and MR (for a single base deletion that removes the 3' or 5' base, respectively), IL and IR (allow insertions on the 5' or 3' side of the pair, respectively) "M" and "I" denote the match and insert states respectively. MATL consists of ML and D as split states and IL as insert state. MATR consists of MR and D split states and IR insert state. The ROOT node is expanded to a start state S and insert state for either the 5' or 3' side, IL and IR. The left child start node BEGL under bifurcation is expanded just to a single S state whereas the right side child start node BEGR under bifurcation is expanded just to an S state and an insert-left (IL) state. Bifurcation and end nodes in the consensus tree becomes B and E states respectively in a CM. It is observed that since the insert state(s) have self-transitions, to allow insertions of more than one base. State transitions are then assigned as follows. For bifurcation nodes, the B state makes obligate transitions to the S states of the child BEGL and BEGR nodes. For other nodes, each state in a split set has a possible transition to every insert state in the same node, and to every state in the split set of the next node. An IL state makes a transition to itself, to the IR state in the same node (if present), and to every state

in the split set of the next node. An IR state makes a transition to itself and to every state in the split set of the next node [5].

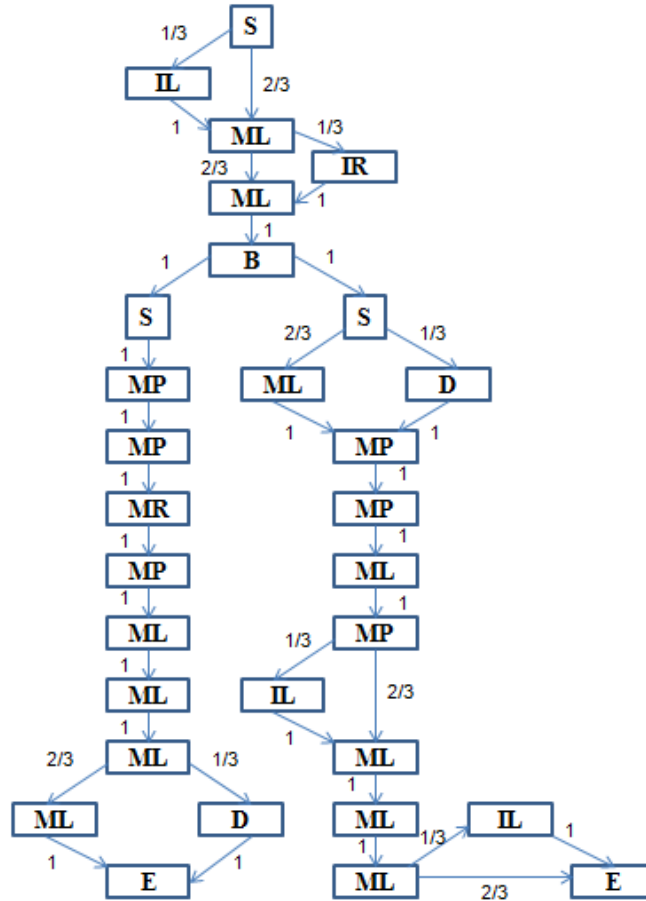


Figure 2.11 CM of the RNA Alignment from a Guide Tree. CM is built by Arranging the States according to the Guide Tree. The arrows and the numbers denote the direction of the transition from one state to another and their corresponding transition probabilities.

The transition probabilities are calculated based on the number of states the preceding states transits to. For instance, as seen in Figure 2.8, there is a deletion in the orc sequence after the ML state nucleotide U and below the residue G of mouse sequence. Hence it is observed that ML state transits to ML state with the probability of 2/3 whereas it transits to D state with the probability of 1/3 considering the three sequenced multiple alignment of the model.

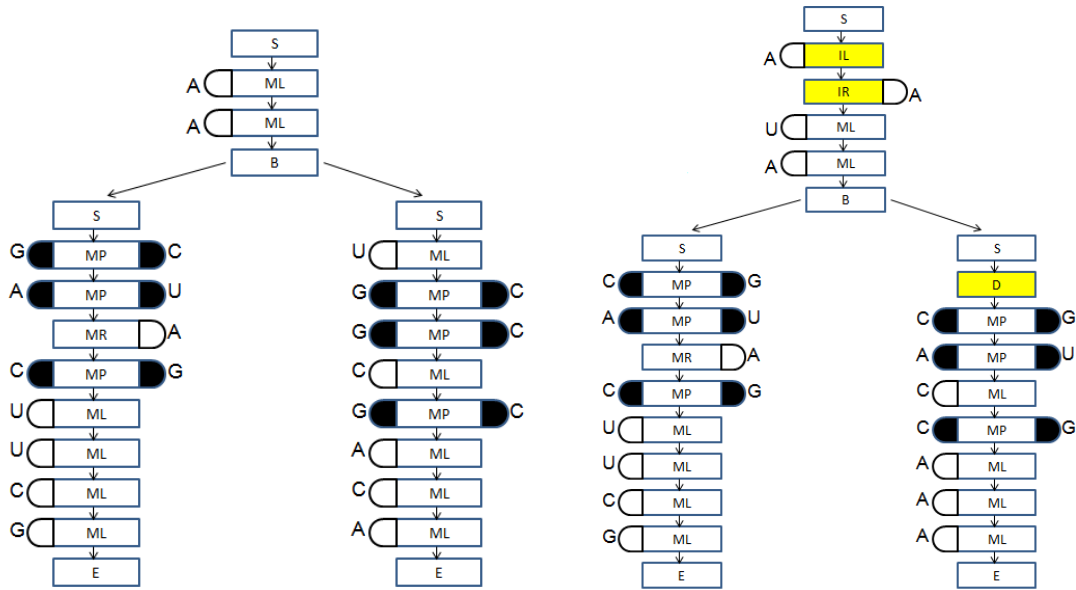


Figure 2.12 Parse Trees shown for Two Sequences-Structures from the Multiple Alignment and the CM. Given the alignment in Figure 2.9 and CM in Figure 2.11, for each sequence, each residue must be associated with a state in the parse tree. Each parse tree corresponds directly to a secondary structure – base pairs are pairs of residues aligned to MP states. Insertions and deletions relative to the consensus use nonconsensus states, shown in yellow.

Source: [6]

CM alignment Algorithms

A covariance model composed of M different states (nonterminals) is denoted as W_1, \dots, W_M . v, y and z are the indices for States W_v, W_y and W_z . These seven state types are associated with symbol emission and state transition probabilities as shown in Table 2.1. Δ_v^L and Δ_v^R are the number of symbols emitted to the left and right respectively by the state v . Let s_v be the state taking its value from P, L,R,D,S, B or E indicating one of the seven possible forms of production rule. Let C_v be the children of the state so that W_v can make transition to W_y and W_z . Let P_v be the parents of the state for the state W_y that make a state transition to W_v . A bifurcation state W_v always transmits to two S states W_y and W_z with probability 1. The children list C_v for a B state is a pair (y,z) for the two S children and the parent list P_y and P_z for both S state children is $\{v\}$ since only W_v transmits

to these states. The Start (S) and Delete (D) states are treated identically in alignment algorithms. However S states occur as the immediate children of bifurcations or as the root state W_1 , whereas D states occur within P,L and R nodes of CM [5].

i. Inside-outside algorithm

As mentioned earlier the inside algorithm calculates the probability score of a sequence with an SCFG. For an observed RNA sequence x , composed of L individual symbols $x_1, \dots, x_i, \dots, x_j, \dots, x_L$. the inside algorithm calculates the probability, likelihood $P(x|\theta)$ of the sequence given a covariance model θ summed over all possible structures for x . The inside algorithm recursively fills a three dimensional dynamic programming matrix with values $\alpha_v(i,j)$, which is the summed probability of all parse subtrees rooted at state v for the subsequence $x_i \dots x_j$. $\alpha_v(i+1,i)$ is the probability for null subsequences of length zero which occurs as the non emitting D, B and S states [5].

For L states the emission probability $e_v(x_i, x_j) = e_v(x_i)$.

For R states the emission probability $e_v(x_i, x_j) = e_v(x_j)$.

For non-emitting states D,S,B and E emission probability $e_v(x_i, x_j) = 1$

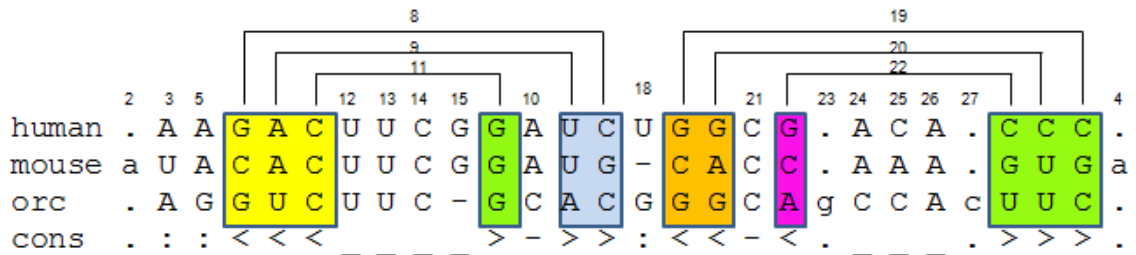


Figure 2.13a Representation of the Model used by carrying out the Infernal Algorithm. The figure shows alignment with the column numbered according to its state.

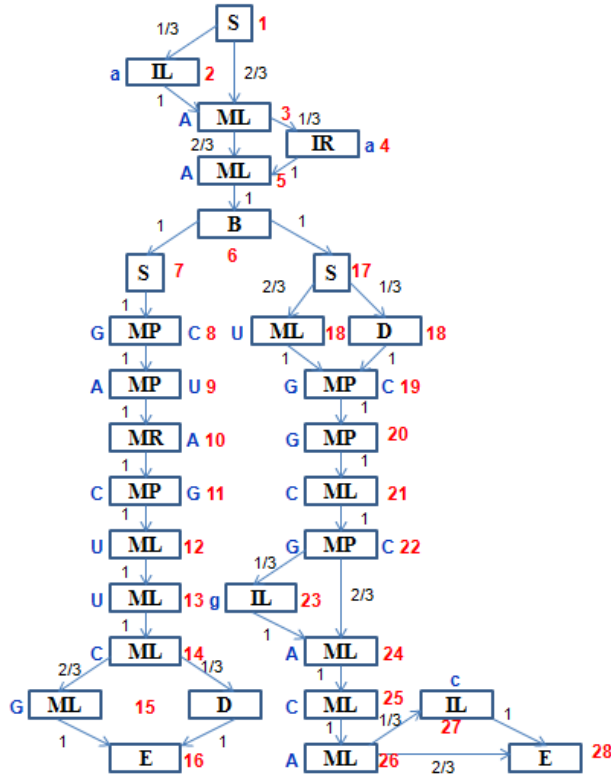


Figure 2.13b CM of the Alignment with Transition Probabilities. Numbers in red indicate the state numbers

The inside algorithm [5]:-

Initialization:

for $j=0$ to L , $v=M$ to 1 ; The matrix values $\alpha_v = (j+1, j)$ according to states is calculated

{ If $(S_v = E)$ for end states

$$\text{Value } \alpha_v = (j+1, j) = 1;$$

Elseif $(S_v = S, D)$ For Start and Delete states

$$\text{value } \alpha_v = (j+1, j) = \sum t_v(y) \alpha_y(j+1, j);$$

Elseif $(S_v = B)$ For Bifurcation states

$$\text{value } \alpha_v = (j+1, j) = \alpha_y(j+1, j) \alpha_z(j+1, j);$$

Else value $\alpha_v = (j+1, j) = 0 \dots$ When $S_v = L, R$ and P }

Matrix: At states 28 and : $\alpha_{28,16}$; $S_v=E$

		→ j				
		0	1	2	...	24
i	1	0	0	0	0	0
	2	1	0	0	0	0
	...		1
	24		
	25					1

Matrix: At state 6 $S_v=B$

		→ j				
		0	1	2	24
i	1	1	0	0	0
	2	1	0	0	0
	...			1	0
	24			
	25					1

Matrix: $S_v=R$

		→ j				
		0	1	2	24
i	1	0	0	0	0
	2	0	0	0	0
	...			0	0
	24			
	25					0

At state 1, 7 and 17: $\alpha_{1,7,17}$; $S_v=S$

		→ j				
		0	1	2	24
i	1	1	0	0	0	0
	2		1	0	0	0
	...			1
	24			
	25					1

At State, $S_v=D$

		→ j				
		0	1	2	24
i	1	1	0	0	0
	2	1	0	0	0
	...			1	0
	24			
	25					1

$S_v=P$

		→ j				
		0	1	2	...	24
i	1	0	0	0	0
	2			0	0

	24					0
	25					

Matrix: At $S_v=L$

		→				
		0	1	2	24
0	j	0	0	0	0
1	j	1	0	0	0
2	j		1	0	0
....	j			
24	j				
25	j					1

Figure 2.14 Matrices filled for all states from End to State 1 at the initialization step

Recursion:

for $j=I$ to L , $i= j$ to 1 , $v=M$ to 1 ; The matrix values $\alpha_v=(i,j)$ according to states

{ If $(S_v = E)$ for end states

Value $\alpha_v=(i,j) = 0$;

ElseIf $(S_v = P)$ and $j-i$

value $\alpha_v=(i,j) = 0$;

Elseif $(S_v = B)$ For Bifurcation states

{ for $(j=i-1$ to $k)$ {value $\alpha_v=(i,j) = \sum \alpha_y(i,k) \alpha_z(k+1,j)$ }; }

ElseWhen $S_v=L,R, S,D$

value $\alpha_v=(i,j) = e_v(x_i, x_j)$ [while $y = C_v$] $\sum t_v(y) \alpha_y(i + \Delta_v^L, j + \Delta_v^R)$ }

Matrix: At states 28 and 16 : $\alpha_{24,16}$; $S_v=E$ At state 27 α_{27} , $S_v=L$, $v=IL$

		→ j				
		0	1	2	24
↓ i	0	0	0	0	0	0
	1	1	0	0	0	0
	2		1	0	0	0
	0
	24					1
	25					

		→ j				
		0	1	2	24
↓	0	0	0	0	0
	1	1	0.33	0.33	0.33
	2		1	0	0.33
			1
	24				0.33
	25					1

At state 27 α_{27} , $S_v=L$, $v=IL$, $P(c)=0.33$, $y=E$, $t(y)=1$, $\Delta_v^L=1$, $\Delta_v^R=0$;

$\alpha_{27}(1,1) = P(c)*1* \alpha_{28}(2,1)=0.33$; $\alpha_{27}(2,2) = P(c)*1* \alpha_{28}(3,2)=0.33$; $\alpha_{23}(1,2) = P(c)*1* \alpha_{28}(4,3)=0.33$;
..... $\alpha_{23}(1,24)$

At states 26: α_{26} ; $S_v=ML$, $P(A)=1$, $y=IL$, $E,t(IL)=1/3,t(E)=2/3$, $\Delta_v^L=1$, $\Delta_v^R=0$;

$\alpha_{26}(1,1) = P(A)*(1/3(\alpha_{27}(2,1))+2/3(\alpha_{28}(2,1))) = 1*(0.33(1)+0.66(1))=0.99$;

Similarly $\alpha_{26}(2,2)$, $\alpha_{26}(1,2)$, $\alpha_{26}(3,3)$, $\alpha_{26}(2,3)$, $\alpha_{26}(1,3)$,..... $\alpha_{26}(1,24)=0.99$

At α_{26} , $S_v=ML$

		→ j				
		0	1	2	24
↓ i	0	0	0	0
	1	1	0.99	0.99	0.99
	2		1	0.99	0.99
			1
	24				1
	25					1

At states 25 and 24 ($\alpha_{25,24}$) $S_v=ML$, $P(C)=0.66$, $P(A)=0.66$ $t(y)=1$, $\Delta_v^L=1$, $\Delta_v^R=0$; Calculation similar to state 26

α_{25}

		→ j				
		0	1	2	24
↓ i	0	0	0	0
	1	1	0.66	0.66	0.66
	2		1	0.66	0.66
			1
	24				1
	25					1

α_{24}

		→ j				
		0	1	2	24
↓ i	0	0	0	0
	1	1	0.66	0.66	0.66
	2		1	0.66	0.66
			1
	24				1
	25					1

At state 22 $S_v=MP$, $P(GC=2/3)=0.66$, When $i=j$ value of cell =0; $\alpha_{22}(2,2)=0$, $y=IL, ML$, $t(IL)=1/3$, $t(ML)=2/3$,
 $\Delta_v^L=1, \Delta_v^R=1$; $\alpha_{22}(1,2)=P(GC)*(1/3(\alpha_{23}(2,2))+2/3(\alpha_{24}(2,2)))$; $=0.66 * [(0.33(0.33))+(0.66(0.66))]= \sim 0.54$. Illy $\alpha_{22}(3,3)$,
 $\alpha_{22}(2,3) \dots \alpha_{22}(1,2)=0.54$

At state 23 $S_v=IL$, $P(g)=0.33$

		→ j				
		0	1	2	24
i	0	0	0	0
	1	1	0.33	0.33	0.33	0.33
	2		1	0.33	0.33	0.33
			1	0.33	0.33
	24				1	0.33
	25					1

State 22(α_{22}), $S_v=MP$

		→ j				
		0	1	2	24
i	0	0	0	0
	1	1	0	0.54	0.54	0.54
	2		1	0	0.54	0.54
			1	0	0.54
	24				1	0
	25					1

State 21(α_{21}), $S_v=ML$, $P(C)=1$

		→ j				
		0	1	2	24
i	0	0	0	0
	1	1	1	1	1
	2		1	1	1
			1
	24				1
	25					1

At states 20 and 19 ($\alpha_{20,19}$), $S_v=MP$, $P(G-C)=1/3$ at α_{20} and 1 at α_{19} , $t(y)$ for both states =1, Calculations similar to α_{22}

State 20(α_{20}), $S_v=MP$

		→ j				
		0	1	2	24
i	0	0	0	0
	1	1	0	0.33	0.33	0.33
	2		1	0	0.33	0.33
			1
	24				1	0
	25					1

State 19(α_{19}), $S_v=MP$

		→ j				
		0	1	2	24
i	0	0	0	0
	1	1	0	0	0
	2		1	0	0
			1
	24				1	0
	25					1

At state 18, $S_v=ML$, D; When $S_v=ML$ $P(U)=1/3=0.33$ and when $S_v=D$, $P(D)=1$, both the states transits to child state MP at probabilities 1 $t(y)=1$.

State 18(α_{18}), $S_v=ML$

	0	1	2	24
0	0	0	0
1	1	0.33	0.33	0.33
2		1	0.33	0.33
....			1
24				1
25					1

State 18(α_{18}), $S_v=D$

	0	1	2	24
0	0	0	0
1	1	0.33	0.33	0.33
2		1	0.33	0.33
....			1
24				1
25					1

At the State 17 (α_{17}), $S_v=S$, the emission probabilities $P(S)=1$, $y=ML,D$, $t(ML)=2/3$, $t(D)=1/3$, $\Delta_v^L=0$, $\Delta_v^R=0$

$$\alpha_{17}(1,1)=P(S)*[(t(ML)*\alpha_{18}(1,1)) + (t(D)*\alpha_{18}(1,1))]=0.33$$

	0	1	2	24
0	0	0	0
1	1	0.33	0.33	0.33
2		1	0.33	0.33
....			1
24				1
25					1

At state 15 (α_{15}), $S_v=D$ When $S_v=ML$ $P(G)=2/3=0.66$ and when $S_v=D$, $P(D)=1$, both the states transits to child END

state E at probabilities 1 $t(y)=1$

State α_{15} : $S_v=D$

	0	1	2	24
0	0	0	0
1	1	0	0	0
2		1	0	0
....			1
24				1
25					1

State α_{15} : $S_v=ML$

	0	1	2	24
0	0	0	0
1	1	0.66	0.66	0.66
2		1	0.66	0.66
....			1
24				1
25					1

At state 14 (α_{14}), $S_v=ML$, $P(C)=1$, $t(ML)=2/3$, $t(D)=1/3$, $\Delta_v^L=1$, $\Delta_v^R=0$

$$\alpha_{14}(1,1)=P(S)*[(t(ML)*\alpha_{15}(1,1)) + (t(D)*\alpha_{15}(1,1))]=0.66 = 1*(0.66(1)+0.33(1))=0.99$$

State 14 (α_{14})

	0	1	2	24
0	0	0	0
1	1	0.99	0.99	0.99
2		1	0.99	0.99
....			1
24				1
25					1

State 13 (α_{13}) $S_v=ML, t(y)=1, P(U)=1,$

	0	1	2	24
0	0	0	0
1	1	1	1	1
2		1	1	1
....			1
24				1
25					1

State 12 (α_{12}) $S_v=ML, t(y)=1, P(U)=1$

	0	1	2	24
0	0	0	0
1	1	1	1	1
2		1	1	1
....			1
24				1
25					1

$S_v=MP, t(y)=1, P(G-C)=1 \Delta_v^L= \Delta_v^R= 1$

	0	1	2	24
0	0	0	0
1	1	0	1	1
2		1	0	1
....			1
24				1	0
25					1

$\alpha_{10} S_v=MR, t(y)=1, P(A)=2/3, \Delta_v^R=1$

	0	1	2	24
0	0	0	0
1	1	0.66	0.66	0.66
2		1	0.66	0.66
....			1
24				1
25					1

$\alpha_9 S_v=MP, t(y)=1, P(A-U)=1 \Delta_v^L= \Delta_v^R= 1$

	0	1	2	24
0	0	0	0
1	1	0	0.66	0.66
2		1	0	0.66
....			1
24				1	0
25					1

$\alpha_8 S_v=MP, t(y)=1, P(G-C)=1 \Delta_v^L= \Delta_v^R= 1$

	0	1	2	24
0	0	0	0
1	1	0	0	0
2		1	0	0
....			1
24				1
25					1

$\alpha_7 S_v=S, t(y)=1, P(S)=1 \Delta_v^L= \Delta_v^R= 0$

	0	1	2	24
0	0	0	0
1	1	0	0	0
2		1	0	0
....			1
24				1
25					1

At α_6 $S_v=B$ (Bifurcation), $P(B)=1$, Child states $y= \alpha_7(S)$ and $z= \alpha_{17}(S)$ or vice versa.

Calculation: $\alpha_6(1,1) = [\alpha_7(1,0)*\alpha_{17}(1,1)] + [\alpha_7(1,1)*\alpha_{17}(2,1)] = 0.33+0=0.33$

Similarly $\alpha_6(2,2) = [\alpha_7(2,1)*\alpha_{17}(2,2)] + [\alpha_7(2,2)*\alpha_{17}(3,2)] = 0.33$

$\alpha_6(1,2) = [\alpha_7(3,2)*\alpha_{17}(3,3)] + [\alpha_7(3,3)*\alpha_{17}(4,3)] = 0.33$

..... $\alpha_6(1,24) = 0.33$

	0	1	2	24
0	0	0	0
1	1	0.33	0.33	0.33
2		1	0.33	0.33
....			1
24				1
25					1

At α_5 $S_v=ML$, $t(y)=1$, $P(A)=2/3$, $\Delta_v^L=1$

At α_4 $S_v=IR$, $t(y)=1$, $P(a)=0.33$, $\Delta_v^L=0$

	0	1	2	24
0	0	0	0
1	1	0.66	0.66	0.66
2		1	0.66	0.66
....			1
24				1
25					1

	0	1	2	24
0	0	0	0
1	1	0.22	0.22	0.22
2		1	0.22	0.22
....			1
24				1
25					1

At α_3 $S_v=ML$, $y= \alpha_5$ (ML) and α_4 (IR), $t(ML)=2/3$, $t(IR)=1/3$, $P(A)=2/3$. $\Delta_v^L=1$

Calculations: $\alpha_3(1,1) = 0.66 * [2/3(1) + 1/3(1)] = 0.66 * 0.99 = \sim 0.65$

State α_3 $S_v=ML$

State α_2 $S_v=IL$, $P(a)=0.33$, $\Delta_v^L=1$, $t(y)=1$

	0	1	2	24
0	0	0	0
1	1	0.65	0.65	0.65
2		1	0.65	0.65
....			1
24				1
25					1

	0	1	2	24
0	0	0	0
1	1	0.33	0.33	0.33
2		1	0.33	0.33
....			1
24				1
25					1

At α_1 , $S_v=S$, $y= \alpha_3$ (ML) and α_2 (IL), $t(\text{ML})=2/3$, $t(\text{IL})=1/3$, $P(S)=1$. $\Delta_v^L= \Delta_v^L=0$
 $\alpha_1(1,1)= 1 * [2/3 (\alpha_3(1,1)) + 1/3(\alpha_2(1,1))] = (0.66 * 0.65) + (0.33 * 0.33) = \sim 0.54$

	0	1	2	24
0	0	0	0
1	1	0.54	0.54	0.54
2		1	0.54	0.54
....			1
24				1
25					1

Figure: 2.15 Matrices from End state to first state of Inside Algorithm Recursion

The final probability after completion $P(x|\theta)$ is in $\alpha_1(1,L)$ i.e. 0.54 in the above matrix. For ‘b’ bifurcation states and other ‘t’ states ($M=t+b$), the space order complexity of the algorithm is $O(L^2M)$ and time complexity is $O(tML^2 + bML^3)$. The outside algorithm calculates values $\beta_v(i,j)$ which are the probability of all parse trees rooted at state v that generate the complete sequence x excluding the subsequence x_i, \dots, x_j .

ii. CYK Algorithm for Database searching [5]

CYK is a variant of inside algorithm with max operation replacing the sums. For a given long sequence or a complete genome, one or more subsequences that match the RNA model is searched across the database. Let L be the length of the database sequence and D, the length of longest aligned subsequence. v,j and d are the indices in the dynamic programming matrix, where d is the length of subsequence i, \dots, j $d \leq D$. A row of scores of the best alignments are calculated for subsequences of lengths $0, \dots, D$ ending at a sequence position j. The CYK algorithm calculates the variable $\gamma_v(j,d)$ which returns the log of the probability $P(S, \pi | \theta)$ of the sequence S and the best parse π given the model θ .

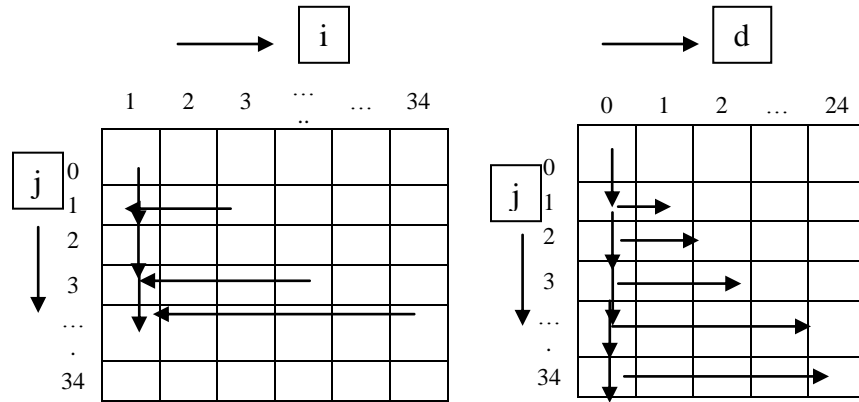


Figure 2.16 Matrices in CYK algorithm. The left matrix represents the order of cells filled which derives the starting point of subsequence. The right matrix represents the cells which gives a log odd score depending on the preceding cell in the direction shown by arrow marks

The left matrix is a standard CYK dynamic programming algorithm for a database sequence of length $L=34$ which is indexed by start position i and end position j . In a 3-dimensional matrix there are M different levels for M states, one per state in the model. The arrows indicate the order of calculation and these arrows for a search algorithm sweeps across the database search in the increasing order of j and the area within the 2 lines that cut through the table determines the part of matrix that is calculated when the maximum matching subsequence length is limited to three or four. The right matrix is the alternative coordinate system for the CYK calculation indexed by end position j and the subsequence length d , where $d=j-(i+1)$ [5].

Initialization:-

for $j=0$ to L , $v=M$ to 1; Values of matrix $\gamma_v(j,0)$ according to states calculated as follows

{If ($S_v=E$) ..For end state

$$\gamma_v(j,0) = 0;$$

Elseif ($S_v = D, S$)...for start and Delete states

$$\gamma_v(j,0) = \max_{y=C_v} [\gamma_v(j,0) + \log t_v(y)];$$

Elseif ($S_v=B$)..for bifurcation state

$$\gamma_v(j,0) = \text{Child state } C_v=(y,z): \gamma_y(j,0) + \gamma_z(j,0);$$

Elseif ($S_v= P,L,R$)

$$\gamma_v(j,0) = -\infty \quad \}}}$$

Recursion:-

for $j=1$ to L , $d = 1$ to D ($d \leq j$), $v=M$ to 1 ; Matrix $\gamma_v(j,d)$ calculated as follows

{If ($S_v=E$) $\gamma_v(j,d) = -\infty = \log(0)$;

Else if ($S_v=P$ and $d < 2$) $\gamma_v(j,d) = -\infty = \log(0)$;

Elseif ($S_v=B$)

$$\gamma_v(j,d) = C_v=(y,z): \max_{0 \leq k \leq d} [\gamma_y(j-k,d-k) + \gamma_z(j, k)]..k \text{ is the subsequence of } d;$$

Elseif ($S_v=L,R,S,D$)

$$\gamma_v(j,d) = \max_{y=C_v} [\gamma_v(j - \Delta_v^R, d - \Delta_v^L - \Delta_v^R) + \log t_v(y)] + \log e_v(x_i, x_j)$$

For Instance Matrix of State 'E' at recursion step is represented as

At $S_v = E$

\longrightarrow d

	0	1	2	24
j	0	0	0	0	0
	1	0	-∞		
	2	0	-∞	-∞	
	0	-∞	-∞	-∞
	34	0	-∞	-∞	-∞

Figure 2.17 Recursion Matrix of End State

Matrices are filled from States 28 and proceeds till state 1 like the Inside algorithm. However unlike inside algorithm the cells contain the log values of the probabilities according to the above algorithm. The value of the cells v , $\gamma_v(j,d)$ depends on one or more possible cells marked y , $\gamma_v(j,d-1)$ for different states y that state v connects to. When v generates a single residue left wise then the parse subtree rooted at state v for the subsequence of length d that ends at j is constructed by adding to subtrees for $y,j,d-1$. This similarly applies to calculation of R and P states [5].

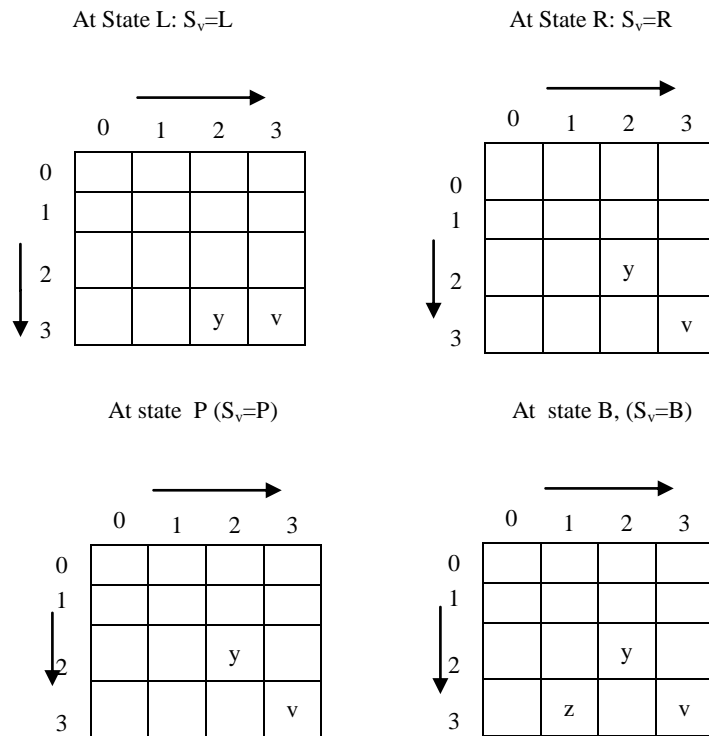


Figure 2.18 Representation of Recursion matrices of four states. The bifurcation state depends on start state scores for the previous rows. When v is the bifurcation state the calculation depends on choosing the best bifurcation point. y and z in the above figure shows the best bifurcation.

Source: [5]

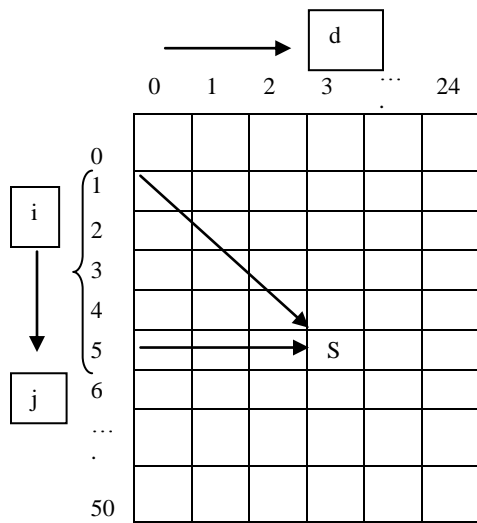


Figure 2.19 An Example showing a Hit in a CYK. S denotes a high hit score and i represent the subsequence.

The scores $\gamma_0(j,d)$ in row j are the log-odds scores of complete alignments to the models which indicates the parse tree starting from the root state ($v=0$) ending at position j . Infernal gives the bit score which are calculated as a log-odds score in log base two, given by the formula below [6].

$$S = \frac{P(\text{seq}|\text{CM})}{P(\text{seq}|\text{null})}$$

$P(\text{seq}|\text{CM})$ is the probability of the target sequence according to the CM obtained and $P(\text{seq}|\text{null})$ is denoted as the probability of the target sequence given a “null hypothesis” model of the statistics of random sequence. In INFERNAL, this null model is considered a simple one-state CM that determines the random sequences that are independent and identically distributed sequences with a specific residue composition, which by default is observed to be equally probable across the four RNA nucleotides ($P(A) = P(C) = P(G) = P(U) = 0:25$). The start point of the match i can be calculated as $i=j-d+1$ which is

obtained as the start point of the alignment For instance in the above figure the cell marked as S has a high score for the subsequence i, \dots, j , then 3 is the starting point of match i . After finding a high-scoring $\gamma_0(j,d)$, a CYK search algorithm reports the score and also the start position i and j of the subsequence that gives this high scores. Once a row j is calculated the best score $\gamma_0(j,d)$ for a $d \leq j$ is determined which if greater than any given threshold is stored in a list but if it overlaps with the previous hit in the list, the lower scoring hit is discarded. A non overlapping match is reported when any hit in the list whose end point j is less than the current minimum start point, $j-D$.

Termination:- $\log P(S, \pi | \theta) = \gamma_0(L,L)$ [5,6].

Due to memory efficiency most of the matrices were discarded and the alignments were not recovered whereas only the scores at the start and end positions were, due to which it was not possible to carry out CYK traceback. However Since CM parse trees represents an optimum alignment to the model and an optimal secondary structure prediction, algorithm is modified and is carried out to recover the optimum SCFG parse tree for the best matching subsequence. The traceback algorithm was implemented using second matrix of traceback pointers or by reconstructing the score calculations where all $D+1$ rows of traceback scores were stored in the memory assuming that the hit is traced back [5]. If both the scores and alignments are obtained then both CYK local and global search algorithm could be efficiently carried out. First a local search is carried out to find the matching subsequences and each of these subsequences are aligned one at a time to the model in a global alignment mode with tracebacks. The traceback is observed to start from $\gamma_0(j,d)$ for a high scoring subsequence of length d ending at j and works backwards. For global alignment with respect to the sequence the

traceback starts from $\gamma_0(L,L)$. The traceback is done by pushing and popping (j,d) on and off a push-down stack [5].

Traceback derives the secondary structure in the genome containing all 24 states in the above example. It begins from root state $\gamma_0(L,L)$, assumed to be State S, searches the highest hit score (log odds) in the root matrix, aligns the subsequence and moves down to the 2nd state (state IL in the model), traces the subsequence from the highest hit and moves down to the 3rd state and proceeding similarly all the way down to the end state E. The subsequences traced at each state of the model is aligned with the model. The memory complexity for a model of M_n non-bifurcation states and M_b bifurcation states is $O(M_n D + M_b D^2)$, and the memory requirement is independent of the database size. The time complexity for the models is $O(M_n LD + M_b LD^2)$ [5].

2.2.3 Accelerating ncRNA Search

According to the experiment it has been understood that RNA homology search with CMs was slow. To accelerate database search Infernal's cmsearch uses two rounds of filters with faster algorithms. These algorithms which accelerate the ncRNA search are implemented in the newer version of Infernal (Infernal v1.1). The first round of filtering was observed to be faster which included the database search using the approach of maximum likelihood (ML) HMMs. Since HMMs are unable to model the interactions between base-paired columns which CM can model, it makes them less sensitive and specific for RNA sequence analysis, However, they are more efficient to compute with, being faster than CMs and thus are useful for filtering. This HMM filter eventually allowed any good hits which are further searched with the second round of filter [5,6].

In the HMM filtering, given the model's parameters and a sequence of observations or states, the distribution over hidden states of the last latent variable at the end of the sequence is computed. The latent variable indicates the variables, not directly observed but inferred from other sequences. For instance when $y(1), \dots, y(t)$ are the model parameters and sequence of observations, $P(x(t) | y(1), \dots, y(t))$ is computed. Filtering is usually used when the sequence of latent variables considered as the underlying states that a process moves through at a sequence of points of time, with corresponding observations at each point in time. HMM filtering used Forward algorithm. The forward algorithm was interpreted to calculate probability of a state at a certain time, given the prior state and was studied to be closely related to viterbi algorithm which is a dynamic programming algorithm to find the most probable path though the model. (http://en.wikipedia.org/wiki/Hidden_Markov_model)

Viterbi algorithm:- According the book chapter Markov chains and Markov models by R. Durbin the algorithm was explained as follows. Suppose a Hidden Markov Model (HMM) of state space S consists of initial probabilities of the path π_i of being in state i and transition probabilities $a_{i,j}$ of transitioning from state i to state j to generate the output of y_1, \dots, y_t . The most likely state sequence x_1, \dots, x_t that produces the observations is given by the recurrence relations. The most probable path can be found by choosing the highest probability. The equation can be given as $\pi' = \operatorname{argmax}_{\pi} P(x, \pi)$. Considering the probability $v_k(i)$ to be the most probable path ending in state k , I be the observation for all states of k in space S , the probabilities calculated for all observations x_{i+1} can be calculated [5].

Initialization: When $i=0$; Initially beginning from the start state (0), $v_0(0)=1$, for $k>0$ $v_k(0)=0$.

Recursion: When $i=1$ to L By pointing to the components backwards the actual sequence can be found the process called back tracking.

At the i^{th} observation; $v_i(i) = e_i(i) * \max_k (a_{ki} * v_k(i-1))$;

Let ptr (for pointer) be the function that returns the maximum values of all atates till L .

$\text{ptr}(i)=\text{argmax}_k (a_{ki} * v_k(i-1))$; ‘argmax’ denotes the set of points of the given argument for which the given function attains its maximum value.

Termination: $P(x, \pi^*) = \max_k (a_{k0} * v_k(L))$; $\pi^*_L=\text{argmax}_k (a_{k0} * v_k(L))$;

The path for traceback can be retrieved by saving back pointers to track which i^{th} observation was used. (http://en.wikipedia.org/wiki/Hidden_Markov_model)

Starting from the End ($i=L$ to 1); $\pi^*_{i-1} = \text{ptr}_i (\pi^*_i)$.

Since state paths is known to produce the sequence x , all probabilities for all possible paths are added to obtain full probability of x which is given by the equation [5]:

$$P(x) = \sum_{\pi} p(x, \pi).$$

It is observed that number of all possible paths increases with the length of sequence. The full probability in the forward algorithm could be calculated using the similar dynamic programming to Viterbi algorithm where the maximizations are replaced by summation.

In the Forward algorithm the quantity $f_k(i)$ is observed to be corresponding to $v_k(i)$ in Viterbi. At the states k , the quantity was expressed by the equation below which

$$f_k(i) = P(x_1, \dots, x_i, \pi_i=k)$$

determines the probability of the observed sequence including x_1 when π_i is required to be equal to k and i^{th} recursion equation is given as below [5].

$$f_k(i+1) = e_l(i+1) \sum_k a_{kl} * f_k(i)$$

Initialization: At start state (0); for $k > 0$, $f_0(0)=1$ and $f_k(0)=0$;

Recursion: At i th observation from 1 to L; $f_k(i)=e_l(x_i) * \sum_k a_{kl} * f_k(i-1)$

Termination: Involved the final calculation of the total probability of the sequence x of length L containing all states, expressed as $P(x) = \sum_k a_{k0} * f_k(L)$ [5].

The dynamic programming technique called query-dependent banding (QDB) was used in second round of filtering, which precalculates regions of the CM dynamic programming matrix that have negligible probability. The calculation by QDB is considered to be dependent only on the query CM itself and not on database being searched. QDB is considered similar to CYK algorithm and was implemented in Infernal software to reduce the average case time complexity of CM alignment from $LD^{2.4}$ to $LD^{1.3}$ for query RNA containing D residues and a target database sequence of length L, resulting in a 4-fold RNA queries. Finally any hits that survive the first and second round of filtering are searched with the final round search strategy and is reevaluated again using the Inside algorithm, described above which determines the final scores of the hits in the database [6].

QDB algorithm:-

For any state v all possible paths could be enumerated down the model from v to the End state, each path possessing a product of transition probabilities used by the paths. And emitting n number of residues ($n=2$ for P, $n=1$ for L and R state in path). Sum of each path probabilities for each n , determined the probability distribution $\gamma_v(d)$, which was defined as the probability that the CM subgraph rooted at v would generate a subsequence of length d . A finite limit l was imposed on maximum subsequence during

calculation since the CM contained self-emitting loops as insert states and there is no finite limit observed on the subsequence. A recursive algorithm was designed to calculate $\gamma_v(d)$ working from leaves of the CM to the root and from smallest subsequence to the largest [9].

For $v=M-1$ (leaves) to 0 (root)

1. When $v=$ End state (E):- { for $d=1$ to 1, $\gamma_v(0)=1$; $\gamma_v(d)=0$ };
2. When $v=$ Bifurcation (B):- { for $d=0$ to 1, $\gamma_v(d)=\sum_{x=0}^d \gamma_y(x) * \gamma_z(d-x)$ };
3. When $v=$ S,P,L,R states:- { for $d=0$ to $\Delta_v^L + \Delta_v^R - 1$, $\gamma_v(d)=0$;
{ for $d= \Delta_v^L + \Delta_v^R$ to 1, $\gamma_v(d)=\sum_{y \in C_v} t(y) * \gamma_{y'}(d - (\Delta_v^L + \Delta_v^R))$ }

For instance while calculating $\gamma_v(d)$, when v is paired state it emits a pair of residues transiting to its child state C_v and the subgraphs rooted at y accounts for rest of the subsequence of lengths $d-2$. Hence $\gamma_v(d)$ is calculated as the sum over all states in $y \in C_v$ of transition probabilities $t(y)$ times the probability of a subsequence (length $d-2$ calculated by recursion) is generated by the subtree rooted at y [9].

Banded CYK database search algorithm for CM

A band $d_{min}(v) \dots d_{max}(v)$ of subsequence length were allowed for each state. As explained above, the CYK search algorithm recursively calculates $\gamma(j,d)$, which is the log probability of the most likely CM parse subtree rooted at state v generating the subsequence of length d as $s_i(j-d+1) \dots s_j$ ending at j of the target sequence s . Initialization is calculated at the smallest subgraphs of state E and shortest subsequence ($d=0$) iterating upwards and outwards to larger subtrees and longer subsequences to a window size of W which is preset. The loops over subsequences of length d which iterate

over the end position j on a target sequence are limited by banding to the range $d_{\min}(v)$ - $d_{\max}(v)$ [9].

Initialization : Bands are imposed:

for $j=0$ to L , $v=M$ to 1 and to 0

$$\gamma_v(j,d) \begin{cases} \text{for } d=0 \text{ to } (d_{\min}(v)-1), -\infty \\ \text{for } d=\min((d_{\max}(v)+1), (j+1)) \text{ to } W, -\infty; \end{cases}$$

W is defined as maximum size of a potential hit to a CM model.

Initialization at $d=0$,

For $j=0$ to L , $v=M$ to 1 till 0

$$\gamma_v(j,0) \begin{cases} \text{When } S_v = E, 0; \\ \text{when } S_v = D,S, \max_{y \in C_v} [\gamma_v(j,0) + \log t_v(y)]; \\ \text{when } S_v = B, C_v=(y,z): \gamma_y(j,0) + \gamma_z(j,0); \\ \text{Else when } S_v = P,L,R, -\infty; \end{cases}$$

Recursion

For $j=1$ to L , $d=\max(1, d_{\min}(v))$ to $\min(d_{\max}(v), j)$, $v=M$ to 1 till 0

$$\gamma_v(j,d) \begin{cases} \text{When } S_v = E, -\infty; \\ \text{When } S_v = B, k_{\min}=\max(d_{\min}(z), (d-d_{\max}(y))) \text{ and} \\ \quad k_{\max}=\min(d_{\min}(z), (d-d_{\min}(y))), \\ C_v=(y,z): \max_{k_{\min} \leq k \leq k_{\max}} [\gamma_y(j-k, d-k) + \gamma_z(j, k)] \\ \text{When } S_v = D,S, \max_{y \in C_v} [\gamma_v(j,d) + \log t_v(y)]; \\ \text{When } S_v = P,L,R, \max_{y \in C_v} [\gamma_v(j- \Delta_v^R, d-\Delta_v^L - \Delta_v^R) + \log t_v(y)] + \log e_v(x_i, x_j) \end{cases}$$

Like the CYK algorithm, the matrices are beginning from End state till the root state of the parse tree. As compared to CYK, The time and space requirement for QBD is negligible and was recorded to be $\Theta(Ml)$ since both M and l are linearly with length L in the residues of the query RNA, $\Theta(L^2)$. Whereas both banded CYK algorithm and normal CYK required the memory of $O(MW + bW^2)$ and $O(L(MW + bW^2))$ time for a model which contains b bifurcation states, window size W and target database length of L . Since M , B and W were reported to scale with the query RNA its time complexity was calculated as $O(LN^3)$. The experiments on the RNA homology search are described in the following chapters [9].

CHAPTER 3

METHODS

3.1 Preparation of Training Set

The training sequences containing RNA multiple alignments were extracted from Rfam database which is a collection or annotations of RNA families, including ncRNAs and other structured RNA each represented by multiple sequence alignment and consensus structures. Rfam open access database was hosted by Wellcome Trust Sanger Institute in collaboration with Janelia Farm. The seed alignments are the alignments that contains representative members of the ncRNA family and also contains the structural information that may or may not contain pseudoknot. Such alignment is used to create the SCFG, which is used to identify additional family members and add them to the alignment with the help of the Rfam software Infernal. Rfam divides ncRNAs into families based on its phylogeny.

wellcome trust
sanger
institute

HOME | SEARCH | BROWSE | FTP | BIOMART | BLOG | HELP

Rfam 11.0 (August 2012, 2208 families)

The Rfam database is a collection of RNA families, each represented by **multiple sequence alignments, consensus secondary structures** and **covariance models (CMs)**. [More...](#)

QUICK LINKS

- SEQUENCE SEARCH**
- VIEW AN RFAM FAMILY**
- VIEW AN RFAM CLAN**
- KEYWORD SEARCH**
- TAXONOMY SEARCH**
- JUMP TO**

QUERY RFAM BY KEYWORD

Search for keywords in text data in the Rfam database.

You can also use the [keyword search box](#) at the top of every page.

Figure 3.1 A Web Server of the Rfam Database.

Source: <http://rfam.sanger.ac.uk/>

The training set of ncRNA alignments were retrieved from four RFAM ncRNA families considered in this case, each containing a pseudoknotted structure and possessing its unique features.

Table 3.1 ncRNA Families of Rfam

No.	ncRNA Rfam family	Rfam ID	Feature
1	RNaseP_bact_a	RF00010	Bacterial RNaseP Class A
2	RNaseP_bact_b	RF00011	Bacterial RNaseP Class B
3	RNaseP_arch	RF00373	Archaeal RNaseP
4	Internal ribosome Entry Site (IRES) IRES-Cripavirus	RF00458	Production of capsid proteins
5	Group 1 intron catalytic site (Intron_gpI)	RF00028	Self splicing ribozymes
6	Hepatitis Delta Virus (HDV) ribozyme	RF00094	Viral replication

The seed alignments of each family consists of a multiple sequence alignment followed by a consensus structure. The training set of any family was prepared and saved in Stockholm format.

> File PDB_01157.ct. RNA SSTRAND database. External source: RCSB Protein Data Bank 2NOQ, number of molecules: 5. The secondary structure annotation was obtained with RNAview.

```

AAAAAUGUGAUCUUGCUUGUAAAUAACAAUUUUGAGAGGUUAAUAAAUAACAAGUAGUGCUAUUUUUGUAUU
UAGGUUAGCUAUUUAGCUUUACGUUCCAGGAUGCCUAGUGGCAGCCCCACAAUAUCCAGGAAGCCCUCUCU
GCGGUUUUUCAGAUUAGGUAGUCGAAAAACCUAAGAAAAUUUACCUGCUGCGCCGGCCAACUCCGUGCCAGC
AGCCGCGGUAAUACGGAGGGCGCGCUGCAUGGCCGUUCUGGUCAGCAUGGCCGGAUGCGUAGGAUAGGUGG
GAGCGCAAGCGCCGGUGAAAUAACCACCCUCCCC

```

Figure 3.4 FASTA Formatted Sequence of IRES of the Organism *S.cerevisiae*

When the above sequence was used as input and BLAST was set to run to find the homologous genome for the family. The BLAST output produce the alignments and gave the high scoring hit which was considered as significant genome for the particular ncRNA family for the organism.

Source: <http://www.rnasoft.ca/strand/>

Sequences producing significant alignments:

Select: [All](#) [None](#) Selected:0

Description	Max score	Total score	Query cover	E value	Max ident	Accession
Cricket paralysis virus nonstructural polyprotein and structural polyprotein genes, complete cds	351	351	59%	1e-93	100%	AF218039.1
Chain A, Structure Of Ribosome-Bound Cricket Paralysis Virus Ires Rna	350	350	59%	5e-93	100%	2NOQ_A
Chain 3, Structure Of The Ribosomal 80s-Eef2-Sordarin Complex From Yeast Obtained By Docking Atomic Mode	99.0	99.0	16%	2e-17	100%	1S1L_3
Chain E, Structure Of Ribosome-Bound Cricket Paralysis Virus Ires Rna	97.1	97.1	16%	8e-17	100%	2NOQ_E

Figure 3.5 BLAST Results. Score summary of the homologous organism genome sequence searched by BLAST in its database, in the decreasing order of identity.

Cricket paralysis virus nonstructural polyprotein and structural polyprotein genes, complete cds

Sequence ID: [gb|AF218039.1|AF218039](#) Length: 9185 Number of Matches: 1

Range 1: 6030 to 6219 [GenBank](#) [Graphics](#) [Next Match](#) [Previous Match](#)

Score	Expect	Identities	Gaps	Strand
351 bits(190)	1e-93	190/190(100%)	0/190(0%)	Plus/Plus
Query 1	AAAAATGTGATCTTGCTTGTAATAACAATTTGAGAGGTTAATAAATTACAAGTAGTGCT	60		
Sbjct 6030	AAAAATGTGATCTTGCTTGTAATAACAATTTGAGAGGTTAATAAATTACAAGTAGTGCT	6089		
Query 61	ATTTTGTATTTAGGTTAGCTATTTAGCTTTACGTTCCAGGATGCCTAGTGGCAGCCCCA	120		
Sbjct 6090	ATTTTGTATTTAGGTTAGCTATTTAGCTTTACGTTCCAGGATGCCTAGTGGCAGCCCCA	6149		
Query 121	CAATATCCAGGAAGCCCTCTCTGCGGTTTTTCAGATTAGGTAGTCGAAAAACCTAAGAAA	180		
Sbjct 6150	CAATATCCAGGAAGCCCTCTCTGCGGTTTTTCAGATTAGGTAGTCGAAAAACCTAAGAAA	6209		
Query 181	TTTACCTGCT	190		
Sbjct 6210	TTTACCTGCT	6219		

Figure 3.6 Optimum Alignment of the Highly Identical Genome Sequence with the Query RNA Sequence. The true or known ncRNA positions in the genome are also obtained from this alignment to compare with those predicted by RNATops and Infernal.

Similarly the genomes for 13 organisms corresponding to their respective RFAM families were retrieved using BLAST and tested for the ncRNA search.

Table 3.2 Genome belonging to ncRNA families.

ncRNA family	Organism	Genome	Genbank Accession
IRES_Cripavirus	<i>S. cerevisiae</i>	Cricket paralysis virus nonstructural polyprotein	AF218039.1
IRES_Cripavirus	<i>Plautia stali intestine virus</i>	Plautia stali intestine virus RNA	AB006531.1
RNaseP_arch	<i>Pyrococcus horikoshii</i>	Pyrococcus horikoshii OT3 DNA	BA000001.2
RNaseP_arch	<i>Pyrococcus furiosus</i>	Pyrococcus furiosus COM1	CP003685.1
RNaseP_bact_a	<i>Thermotoga maritima</i>	Thermotoga maritima MSB8	AE000512.1
RNaseP_bact_a	<i>Deinococcus radiodurans</i>	Deinococcus radiodurans, strain R1	AE000513.1
RNaseP_arch	<i>Methanocaldococcus jannaschii</i>	Methanocaldococcus jannaschii DSM 2661	L77117.1
RNaseP_arch	<i>Archaeoglobus fulgidus</i>	Archaeoglobus fulgidus , strain DSM 4304	AE000782.1
RNaseP_bact_b	<i>Bacillus subtilis</i>	Bacillus subtilis BEST7003 DNA, complete genome	AP012496.1
Intron_gpI	<i>Tetrahymena thermophila</i>	Tetrahymena thermophila strain ATCC 30382 18S ribosomal RNA	JN547815.1
RNaseP_bact_a	<i>Thermus thermophilus</i>	Thermus thermophilus HB27, complete genome	AE017221.1
Intron_gpI	<i>Homo Sapiens</i>	Azoarcus sp. BH72	AM406670.1
HDV_Ribozyme	<i>Hepatitis Delta Virus</i>	Hepatitis delta virus isolate 59045-CAR delta antigen gene	JX888110.1

3.3 Test with RNATOPS

The Genomes retrieved from RNASTRAND were further tested for pseudoknots with a profile based RNA structure search program, RNATOPS (RNA via Tree decOmPoSition) which detects RNA pseudoknots in genomes as mentioned in Chapter 2. The files were uploaded in the web server of RNATOPS called RNATOPS-W.

The screenshot shows the homepage of the RNATOPS-W web server. At the top, there is a header for 'RNA INFORMATICS UNIVERSITY OF GEORGIA' with a search bar and 'Web RNA@UGA' links. Below this is a red navigation bar with buttons for HOME, RESEARCH, PUBLICATIONS, PEOPLE, PHOTOS, NEWS, and MEMBERS. A left sidebar contains several red buttons for 'SOFTWARE', 'DATA SETS', 'RELATED', and 'OTHER', each with a list of links. The main content area has a title 'RNATOPS-W: A Web Server for RNA Pseudoknot Search' followed by a paragraph describing the web server's capabilities. Below this is a red link 'Link to the RNATOPS-W Server', a 'Citation' section with a reference to Wang et al. (2009), and a 'Related Publications' section with a reference to Huang et al. (2008).

Figure 3.7 University of Georgia RNATOPS-W Home Page. The left column gives the options to download software and algorithms. This page provides link to RNATOPS-W web server for analysis.

Source: <http://rna-informatics.uga.edu/?f=software&p=RNATOPS-w>

RNATOPS-W

A Web Server for RNA Pseudoknot Search



RNATOPS-W is a web server version of the program [RNATOPS](#) (version: rnatops.v1.1), a profile based RNA structure search program that can detect RNA pseudoknots in genomes. It has a filtering function for search speed-up and comes with the options of automatic (HMM) filter selection and manual filter selection by the user.

To use the RNATOPS-W server, you need to provide as input:

- (1) a structure profile in the pasta format, and
- (2) genome sequences in the fasta format,

each can be in a file or in the text box to be uploaded to the server. ([Click for some examples](#))

By default, the server selects from the profile an HMM filter and searches with the filter. You can also opt to select your own filter (in an additional step before submitting your search request). Your own filter can be either an HMM or a substructure filter. You may request the whole structure search result, the filtering result, or both. You may also change the default parameter values for the search. After submitting your search request, you will be given an ID to retrieve your search result at a later time from this page or by following a specified web link.

Recommended web browsers to use with RNATOPS-W are: IE (version 7.0 or above) and Firefox (version 2.0 or above). It may also support Safari (version 2.0.4 or above).

Retrieve Results of a Previous Search with ID:

New Search: Inputs and Parameter Setting

Structure profile: File (choose a file) Input
 No file chosen

Target genomes: File (choose a file) Input
 No file chosen

Filter: Automatic HMM filter selection Manual selection Neither

Output: Results of whole structure search Results of filtering Both

Parameters: All default Adjust parameters

Figure 3.8 RNATOPS-W Open Access Web Server. Server is maintained by University of Georgia

Source: <http://128.192.141.226:8080/rnatops-w/>

A function of automatically addition of HMM filter is an addition to RNATOPS. The training set in Stockholm format was converted into “pasta format” using RNAPasta, a JAVA application on windows available as a precompiled java archive file (.jar). RNA Pasta is a Java application used to calculate a variety of statistics related to RNA stem-loop and pseudoknot structures. The pasta file was further used as input file and

contained a multiple structural alignment for a set of training RNA sequences. It was observed that RNATOPS-W accepts the structure profile input in 2 different versions of pasta format, namely one-line version and two-line version, one of them shown the figure below.

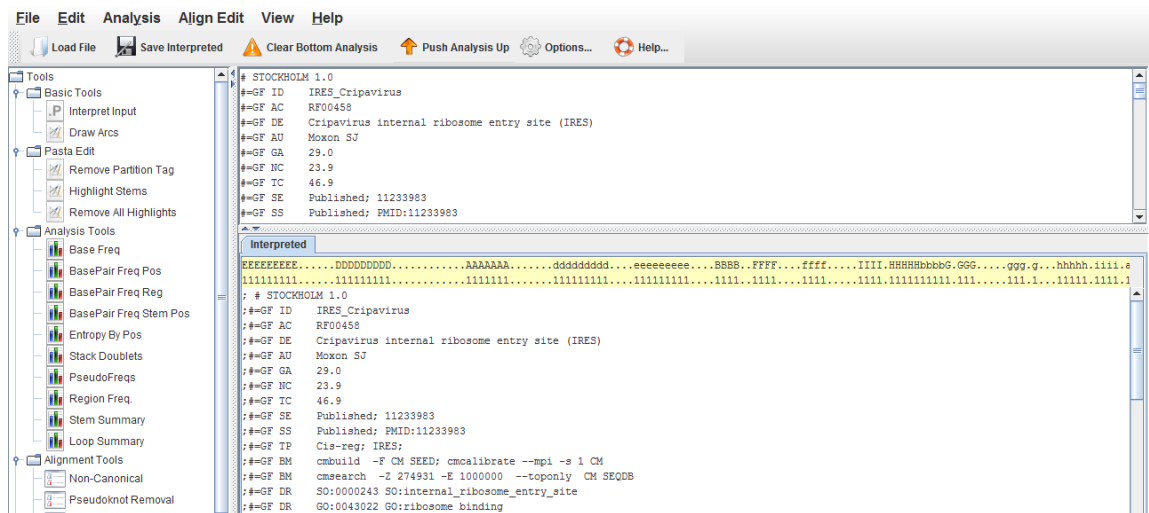


Figure 3.9a Pasta Format of the Multiple Alignment in the RNApasta tool. Stockholm format of the ncrRNA training file loaded in RNApasta tool which gives pasta formatted structure profile.

The structural graph described in section 2.2 produced from consensus sequence represented the first line of Pasta output. The rest of the lines below the consensus structures were the RNA sequences structurally aligned to the consensus structures.

```

EEEEEEEEEE.....DDDDDDDDDD.....AAAAAAA.....ddddddddd.....eeeeeeee
e....BBBB..FFFF....ffff.....IIII.HHHHHbbbbG.GGG....ggg.g...hhhh.iiii.
a.aaaaaa...KKKKK...JJJJ.CCCCCjjjj..kkkkk.....cccc.....
111111111.....111111111.....1111111.....111111111....11111111
1....1111..1111....1111....1111.1111111111.111....111.1...11111.1111.
1.111111...11111...1111.1111111111..11111.....11111.....
>Drosophila_C_virus.1
GUUAAGAUGUGAUCUUGCUUCCUU--
AUACAAUUUUUGAGAGGUUAAUAAGAAGGAAGUAGUGCUAUCUUAU-
AAUUAGGUUAACUAAUUUAGUUUACUGUUCAGGAUGCCUAU-UGGCAGCCCCA-UAA-UAUCCAGGACAC-
CCUCUCUGCUUCUUAUAUGAUUAGGUUGUCAUUUAGAA--UAAGAAAAUAACCUGCUAACUUUCA
>Black_queen_cell_vir.1
CCAACA AUGUGAUCUUGCUUGCGGA-
GGCAAAAUUUGCACAGUAUAAAUCUGCAAGUAGUGCUAUUGUUGG-
AAUACCCGUACCUAAUUUAGGUUUACGCUCCAAGAUCGGUGGAUAGCAGCCCUAUCAA-UAUCUAGGAGAA-
CUGUGCU-AUGUUUAGAAGAUAUAGGUAGUCUCUAAACA---GAACAAUUUACCUGCUGAACAAAUU
>Himetobi_P_virus.1
GAAA AUGUGUGAUCUGAUUAGAAG--UAAGAAAUUCCUAG-
UUUAUAUUUUUUAAUACUGCUACAUUUUU-
AAGACCCUAGUUUUUAGCUUUACCGCCCAGGAUGGGGUG-CAGCGUCCUG-CAA-UAUCCAGGGCAC-
-CUAGGUGCAGCCUUGUAGUUUAGUGGACUUUAGGCU--AAAGAAUUUCACUAGCAAUAUUAAU
>Rhopalosiphum_padi_v.1
AGUGUUGUGUGAUCUUGCGCGAU-----AAAUGCUGACG---
UGAAAACGUUGCGUAUUGCUACAACACU-----UGGUUAGCUAUUUAGCUUUACUAAUCAAGACGCCGUC-
GUGCAGCCCAC-AAAA-GUCUAGAU-----
CGUCACAGGAGAGCAUACGCUAGGUCGCGUUGACUAUCCUUUAUUAU-GACCUGCAAUAUUAAC
>Triatoma_virus.1
UUGACUAUGUGAUCUUGCUUUCG----
UAAUAAAAUUCUGUACAUAAGGUCGAAAGUAUUGCUAUAGUUAAGGUUGCGCUUGCCUAAUUUAGGCAUAC
UUCUCAGGAUGGCGCG-UUGCAGUCCAA-CAAG-AUCCAGGGACUGUACAGAAUUUUC-
UAUACCUCGAGUCGGGUUU-GGAA--UCUAAGGUUGACUCGCUGUAAAAUAAU
>Plautia_stali_intest.1
CUGACUAUGUGAUCUUAAUUAAAAUUAGGUUAAAAUUUCGAGGUUAAAAUAGUUUUAAUUAUUGCUAUAGUCU
U-AGAGGUCUUGUAUUAUUUACUUUACCACACAAGAUGGACCG-GAGCAGCCCUC-CAA-
UAUCUAGUGUAC--CCUCGUCGUCGCUCAAACAUAUAGUGGUGUUGUGCGA--
AAAGAAUCUCACUUCAAGAAAAAGAA

```

Figure 3.9b The Pairing Structure and RNA Multiple Alignment in Pasta format. RNAPasta produces two strings, one of which is upper and lower case letters, the second of which is a numerical index or subscript. Since every base-pair/ single residue occur only once in the structure, every letter (upper case and lowercase) is indexed as 1. For instance E1-e1, D1-d1, etc. D1-d1 and A1-a1 base pairing region form a pseudoknot.

The two input files, Pasta formatted structure profile and the genome file were submitted and along with those an option for automatic filter available in the program was selected with the default parameters, and submitted for which RNATOPS filters the

conserved regions and gives the hits. Similarly the ncRNA structure profile in Pasta format was prepared for the training set from 4 RFAM Families as input. Secondly above retrieved genomes belonging to their corresponding RNA families were loaded in RNATOPS-W as input along with Pasta structure profile.

3.4 ncRNA search with Infernal

Infernal v1.0.2 was downloaded and installed from <http://infernal.janelia.org/>. The Multiple RNA sequence alignment with secondary structure annotation, of the six RFAM families were stored in Stockholm format as mentioned above. Thirteen genomes were tested for ncRNA search. ‘cmbuild’ and ‘cmsearch’ were the two core programs used for searching and aligning ncRNA model to the genomes.

3.4.1 Building a model using cmbuild program

The cmbuild reads an RNA multiple sequence alignment in a Stockholm format containing consensus secondary structure annotation. To construct the architecture of CM cmbuild uses consensus secondary structure and saves the CM in a .cm file. Six .cm files were built of each RFAM families respectively. While cmbuild constructs a model from an input alignment it is observed that the selection is decided upon which columns of the input alignment are defined as match (called “consensus”) and insert columns. It is understood that to implement this strategy the weights for each sequence are computed to down weight closely related sequences and up weight distantly related ones. [6]. Then, for each position, the sum of the weights of the sequences that include a residue at the position was computed and if that sum was nearly half the total number of sequences

- "rel entropy: HMM": the total relative entropy of the model ignoring secondary structure divided by the number of consensus columns [6].

```

INFERNAL-1 [1.0.2]
NAME      IRES_Cripavirus
GA        29.00
TC        46.90
NC        23.90
STATES    622
NODES     171
ALPHABET  1
ELSELF    -0.08926734
WBETA     1e-07
NSEQ      6
EFFNSEQ   2.409
CLEN      201
BCOM      bin/cmbuild IRES/IRES_Cripavirus_new.cm IRES/IRES_Cricket_Cripavirus_new_stockholm.txt
BDATE     Fri Apr 12 20:09:46 2013
NULL      0.000 0.000 0.000 0.000
MODEL:

      [ ROOT  0 ]
      S  0  -1 0   1   4  -7.034  -8.280  -0.072  -4.733
      IL 1  1 2   1   4  -2.817  -4.319  -0.613  -2.698           0.000 0.000 0.000 0.000
      IR 2  2 3   2   3  -1.925  -0.554  -4.164
      [ MATR  1 ]
      MR 3  2 3   5   3  -8.516  -0.017  -6.834           0.387 -0.426 -1.159 0.585
      D  4  2 3   5   3  -6.390  -1.568  -0.620
      IR 5  5 3   5   3  -1.925  -0.554  -4.164           0.000 0.000 0.000 0.000
      [ MATR  2 ]
      MR 6  5 3   8   3  -8.516  -0.017  -6.834           1.514 -1.754 -1.799 -0.835
      D  7  5 3   8   3  -6.390  -1.568  -0.620
      IR 8  8 3   8   3  -1.925  -0.554  -4.164           0.000 0.000 0.000 0.000
      A

      [ BIF   27 ]
      B  81  80 3   82  362
      [ BEGL  28 ]
      S  82  81 1   83   1  0.000
      [ BIF   29 ]
      B  83  82 1   84  296
      [ BEGL  30 ]
      S  84  83 1   85   4  -0.042  -6.929  -6.337  -6.977
      [ MATP  31 ]
      MP 85  84 1   89   6  -9.181  -9.120  -0.020  -7.896  -8.176  -8.571  -2.315  -
      2.158 -2.521  1.669 -2.110 -2.791  1.475 -0.781 -2.532  1.370 -2.638  0.606  1.506 -2.199
      -0.496 -1.86
      6
      ML 86  84 1   89   6  -6.250  -6.596  -1.310  -1.005  -6.446  -3.975  0.660  -
      0.612 -0.293 -0.076
      MR 87  84 1   89   6  -6.988  -5.717  -1.625  -5.695  -0.829  -3.908  0.660  -
      0.612 -0.293 -0.076
      D  88  84 1   89   6  -9.049  -7.747  -3.544  -4.226  -4.244  -0.319
      IL 89  89 5   89   6  -2.579  -2.842  -0.760  -4.497  -5.274  -4.934  0.000
      0.000 0.000 0.000
      IR 90  90 6   90   5  -2.408  -0.496  -5.920  -4.087  -5.193
      0.000 0.000 0.000
      B

```

Figure 3.11 CM of IRES Cripavirus ncRNA built by the cmbuild Program. Shows ROOT, MATR, BIF, BEGL and MATP nodes with their expanded states of the model. the model continues going all the way though other states (MATR, MATL, MATP, BEGL, BEGR, BIF, etc) till the END as shown in Appendix C. The nodes are numbered according to its position in the structure. To the left are the states represented by S, MP,ML, MR, D, IL,IR followed by with the state IDs The negative integers denote the CM bit scores or log-odds (log of their emission and transition probabilities).

As mentioned in the above text and according to the above figure, node MATP is expanded into total 6 states, namely an MP, D, ML and MR IL and IR. MATL consists of ML and D as split states and IL as insert state. MATR consists of MR and D split states and IR insert state. The ROOT node is expanded to S, IL and IR. The left child start node BEGL under bifurcation is expanded just to a single S state whereas the right side child start node BEGR under bifurcation is expanded just to an S state and an insert-left (IL) state. The position-specific scores were assigned for the four possible residues at single-stranded positions, the 16 possible base pairs at paired positions and for insertions and deletions. [6]. These scores were observed to be the log-odds scores derived from the observed counts of residues, base pairs, insertions and deletions in the input alignment, combined with prior information derived from structural RNA alignments [6]. The CM model contained 44 MP, 122 ML, 79 MR, 126 IL, 82 IR, 14 S, 16 B, 164 D and 27 E states, which determines the multi-branched structure.

3.4.2 Searching the CM against a sequence database using cmsearch.

After the six CMs belonging to six RFAM families each from their ncRNA multiple alignments were constructed they were made to search against the corresponding genome sequences using Infernal's cmsearch program. The CM calibration was optional but computationally expensive for a complex RNA structure. Calibration was performed to obtain E-values that estimate the statistical significance of hits in a database search, determining appropriate hidden Markov model (HMM) filter thresholds for accelerating search. More importantly apart from searching the model, the alignments were needed to be built with the model of the large family, and moreover cmsearch is understood and

proved to be the only INFERNAL program that uses E-values and HMM filters. Hence calibrations were not needed to be carried out even since it was time consuming. Also it was observed that the amount of time taken by calibration depended on size of the RNA family being modeled. Therefore without calibrating the model, directly, the six models each, constructed from cmbuild were taken as input Infernal's cmsearch program to scan the genome, belonging to the corresponding family, for the ncRNA model. Default filter threshold cutoffs were used though not accelerating the search [6].

Apart from cmbuild and cmsearch, the other optional programs could be used in Infernal are as mentioned below:-

- **cmemit:** reads the CMs in a .cm file and generates a number of sequences from the CM(s).
- **cmstat:** Calculates and displays the statistics of the covariance models determining statistics on calibrated or non-calibrated CM files. Moreover the program prints general statistics of the model and the alignment built.
- **cmscore:** Used to align and score the homologous sequences and display summary statistics on timings and scores.
- **cmalign:** Used for aligning the homologous sequences to the covariance model (CM) in cmfile, and outputs a multiple sequence alignment.

Since the goal of this thesis was to analyze the model alignment with the genome sequence, the above four programs were not necessarily carried out. The alignments produced by cmsearch program were further analyzed using a Perl script to find out the misalignments.


```

*****
* Searched done : with RNATOPS V1.1 *
* By RNA-Informatics @ UGA *
* Total no of hits: 1 *
* Total time used 2.5e-05 hours *
* Time: 20:31:00 EDT 2013-03-26 *
*****

```

```

*****
* Whole Structure Result *
* *
* Profile file : Cripavirus_cricket_pasta *
* Profile length : 210 *
* *
* Genome file : Cricket_paralysis_virus.txt *
* Number of sequences : 1 *
* Total length of sequences : 9185 *
* *
* Search parameters setting: *
* Pseudocount = 0.001 *
* Num of stem candidates = 10 *
* Score threshold for hits = 0 *
* Num of nt overlap between stems = 2 *
* Candidate representatives only = Yes *
* Shortest candidate representatives = Yes *
* IShiftNumMergeCand = No *
* Nts allowed in null loops = 3 *
* Pcoeff = 2 *
* Search with jump = Yes *
* Search step size = 1 *
* Search reversed complement sequence = No *
*****

```

Whole structure search hit 1

```

-----
gi|8895506|gb|AF218039.1|
Plus search result
Hit Positions: 6027-6215
Alignment score = 118.894

```

Folded structure

```

    1 EEEEEEEEE.....DDDDDDDD.....AAAAA.A.....DDDDDDDD... 61
    1 11111111.....11111111.....1111.1.....11111111... 61
6028 GCAAAAATGTGATCTTGCTTGTAATAACAATTTTGAGAGGTTAATAAATTACAAGTAGTG 6088

    61 .EEEEEEEE...BBB..FFF...FFF.....IIII.HHHHHBBBBGGGG....G 121
    61 .11111111...1111..1111....1111....1111.111111111111.....1 121
6088 CTATTTTGTATTTAGGTTAGCTATTTAGCTTTACGTTCCAGGATGCCTAGTGGCAGCCC 6148

    121 GGG..HHHHH.IIII.A.AAAA...KKKKK...JJJJ.CCCCCJJJJ..KKKKK..... 181
    121 111..1111.1111.1.1111...1111...1111.11111111..1111..... 181
6148 CACAATATCCAGGAAGCCCTCTCTGCGTTTTTCAGATTAGGTAGTCGAAAAACCTAAGA 6208

```

```

181 ....CCCCC 190
181 ....11111 190
6208 AATTTACCT 6217

Structure alignment

1 EEEEEEEEE.....DDDDDDDD.....AAAAA1A.....DDDDDDDD... 61
1 111111111.....111111111.....1111111.....111111111... 61
6028 GCAAAAUGUGAUCUUGCUUGUAAAUAACAAUUUUGAGAGGUAAAUAUUACAAGUAGUG 6088

61 .EEEEEEEEEE...BBBB..FFFF....FFFF.....IIII.HHHHHBBBBGGGG.....G 121
61 .111111111...1111..1111....1111.....1111.1111111111111.....1 121
6088 CUAUUUUUGUAUUUAGGUUAGCUUUUAGCUUUACGUUCCAGGAUGCCUAGUGGCAGCCC 6148

121 GGG..HHHHH.IIII.ArAAAAA...KKKKK...JJJJ.CCCCCJJJJ.--KKKKK... 181
121 111..11111.1111.1r11111...11111...1111.1111111111...11111... 181
6148 CACAAUAUCCAGGAAGCCUCUCUGCGUUUUUCAGAUUAGGUAGUCG--AAAACCUAA 6208

181 .....CCCCC 192
181 .....11111 192
6208 GAAUUUUACCU 6219

*****
* Searched done : with RNATOPS V1.1 *
* By RNA-Informatics @ UGA *
* Total no of hits: 1 *
* Total time used 0.00292778 hours *
* Time: 20:31:11 EDT 2013-03-26 *
*****

```

Figure 4.1 RNATOPS Result for search of IRES Cripavirus against Cricket Paralysis virus Genome

According the output generated by RNATOPS, the pseudoknots were accurately detected without compromising computation time and all combinations of stems for pseudoknot alignment were feasibly considered through a non-conventional tree decomposition-based dynamic programming explained in Chapter 2, section 2.3. When requested, upon use of the filtering step RNATOPS-W gave the output of the corresponding filtering result which consisted of a header, a tail, and list of filtering hits. The header and the tail contained the information regarding the method of filter generation(automatic or manual), filter type (HMM or substructure), filter positions in the original profile, name of the genome searched, number of sequences in the genome, total length of the sequences in the genome, number of filtering hits, the total time used in

filtering, etc. Each filtering hit contains the following information containing the name (and directed) of the sequence in which the hit is found, positions of the hit, score and alignment to the filter, the extended sequence flanking the hit and positions of the extended sequence. Followed by the filtering, RNATOPS-W gave the result of whole structure search, which was applied on extended sequences flanking filtering hits. Like filtering result, it consisted of a header, a tail, and a list of whole structure search hits. The header and the tail contained the information namely, the name and length of the structural profile, name of the searched genome, number of sequences and total length of the genome, total number of whole structure search hits, parameter settings and total time used in the whole structure search. Each whole structure search hit contained the information regarding the genome sequence (and direction) in which the hit is found, hit positions in the sequence, the fold of the hit sequence, annotated with regard to the consensus structure in the profile and the structure alignment, along with the score, of the hit sequence to the consensus structure in the profile.

For most of the genomes RNATOPS accurately searched the ncRNA in the genomes and aligns the pseudoknot accurately. However it failed to search the instance ncRNA of interest in the target genomes of distant RFAM family or in other words when genome differed in structure significantly from those in the training set. When few of the genomes were tested for their corresponding ncRNAs it was observed that in some of the cases RNATOPS failed to give output of the whole structure search whereas in other cases which involved ncRNAs and genomes belonging to distant families RNATOPS failed to search the ncRNAs. One of the instances of those cases are shown below. When bacterial ncRNA of RNaseP arch family was searched in the genome of

Methanocaldococcus_jannaschii organism RNATOPS failed to produce the whole structure search including the alignment.

```
*****
* Filtering Result *
* *
* Profile file : arch_new_pasta *
* Profile length : 721 *
* *
* Filter generation: automatic seleted *
* Filter type : HMM *
* Filter info : positions from 687 to 714 *
* Genome file : Methanocaldococcus_jannaschii.txt *
* Number of sequences : 1 *
* Total length of sequences : 1664957 *
*****
```

Filtering hit 1

```
-----
>gi|6626255|gb|L77117.1|(81670-81689)[81367-81698]
Plus search result
Hit Positions: 81670-81689
Alignment score = 1.46672
Alignment to the filter
CAGAAG--GGCTTAAG-AAGGGT---
mmmmmmmmddmmmmmmmmmmmmmmmmmmddd
Extension positions: 81367-81698
Extension of the hit
GAAATTTAAAGTGCTATGTCTTTGATTTACCAAATGTTATTGAAGAAACCAAAAAATTTA
TCAAAAAATACAATGCAAAAAACGTCTTCACAATTACTGGAGATTTTTATAAGGATGATA
TCGGAAAGGGCTACGATATAATATTCTGCTCATATAATCCAGGTGGAAAAAATCCAAAGA
TTGCAGAGAAGGTTTATAATGCCTTAAATGAAGGAGGTTTATTTATAAATAAGCAATTCT
TTCCAGATAAGGAAGAGGGTATTGAAGACTATATAAACAACATGGAATGGAACCTTCTCTA
AACCAGAAGGGCTTAAGAAGGGTAAAATAAGA
```

Filtering hit 2

```
-----
>gi|6626255|gb|L77117.1|(179247-179267)[178944-179276]
Plus search result
Hit Positions: 179247-179267
Alignment score = 0.516452
Alignment to the filter
CAAAAGGTGG---ATTCT-TACACT---
mmmmmmmmmmmmdddmiimmmmmmmmmddd
Extension positions: 178944-179276
Extension of the hit
CTAAAGAGATTTTAGGTAAGCAGTTAAACATTACAGATGTCTTCCAAGAAGGAGAGTTAG
TCGATACAATTGGAGTTACAAAAGGTAAAGGATTCCAAGGACAAGTTAAAAGATGGGGAG
TTAAAATACAATTTGGTAAGCACGCAAGAAAAGGAGTAGGAAGACACGTTGGTTCTATTG
GTCCATGGCAACCAAGATGGTTATGTGGAGTGTCCAATGCCAGGTCAAATGGGATACC
ACCAAAGAAGTGAATACAACAAGAGAATATTAAGATTGGAAACAATGGGGATGAAATTA
CACCAAAAGGTGGATTCTTACACTACGGGGTTA
```

```

*****
* Searched done : with RNATOPS V1.1 *
* By RNA-Informatics @ UGA *
* Total no of hits: 6 *
* Total time used 0.00434444 hours *
* Time: 20:09:14 EDT 2012-10-11 *
*****

*****
* Whole Structure Result *
* *
* Profile file : arch_new_pasta *
* Profile length : 721 *
* *
* Genome file : Methanocaldococcus_jannaschii.txt *
* Number of sequences : 1 *
* Total length of sequences : 1664957 *
* *
* Search parameters setting: *
* Pseudocount = 0.001 *
* Num of stem candidates = 10 *
* Score threshold for hits = 0 *
* Num of nt overlap between stems = 2 *
* Candidate representatives only = Yes *
* Shortest candidate representatives = Yes *
* IShiftNumMergeCand = No *
* Nts allowed in null loops = 3 *
* Pcoeff = 2 *
* Search with jump = Yes *
* Search step size = 1 *
* Search reversed complement sequence = No *
*****

*****
* Searched done : with RNATOPS V1.1 *
* By RNA-Informatics @ UGA *
* Total no of hits: 0 *
* Total time used 0.0919333 hours *
* Time: 20:20:16 EDT 2012-10-11 *
*****

```

Figure 4.2 RNATOPS output for the ncRNA search of RNaseP arch ncRNA against the genome *Methanocaldococcus jannaschii* DSM 2661 of organism *Methanocaldococcus jannaschii*.


```

.....
181 AauUaACuuGCaaAaAAaAAu 201
    AAUU AC+UGC+A A ++ A+
6208 AAUUUACCGCUACAUUUCA 6228

#
# Post-search info for CM 1: IRES_Cripavirus
#
# rnd  mod  alg  cfg  beta  bit sc cut  num hits  surv fract
# ---  ---  ---  ---  ---  ---  ---  ---  ---
#   1  hmm  fwd  loc   -    3.00    5    0.0835
#   2   cm  cyk  loc 1e-10  0.00    6    0.1083
#   3   cm  ins  loc 1e-15  0.00    6    0.0252
#
#   run time
# -----
#         00:00:02
//
#
# CPU time: 2.06u 0.00s 00:00:02.06 Elapsed: 00:00:02

```

Figure: 4.3a Infernal Output given for the Genome Cricket Paralysis virus of Organism *S. cerevisiae* tested for ncRNA from family IRES Cripavirus.

The matching pairs of symbols <>, (), [], or {} represented the base pairs. These different symbols indicated the “depth” of the helix in the RNA structure. For instance <> denoted simple terminal stems; () as “internal” helices of all terminal stems; [] as internal helices enclosing a multifurcation that includes at least one annotated () stem and {} accounted for all internal helices which encloses deeper loop residues which are indicated by commas, ... Hairpin loop residues were indicated by underscores, _; simple stem loops as <<<<____>>>>. The dashes – indicate the bulge and interior loop residues. Unstructured single stranded residues completely outside the structure (unenclosed by any base pairs) were annotated by colons, :. Tildes ~ were observed in alignments of a known (query) structure annotation to a target sequence of unknown structure. All other symbols _-,:~ indicated single stranded residues.

Annotation parameters displayed in the output describe the following[6].

- ‘**rnd**’:- round of searching each row pertains to. three rounds, two filter rounds plus the final round. The hits reported by **cmsearch** are those that survive both filters and the final round of searching.
- ‘**mod**’ :- model each round of searching. For the first round used CP9 HMM to filter, for the last two rounds the Covariance Model (CM) is used.
- ‘**Alg**’ search algorithm used by each round. The HMM filter round always uses the Forward (“fwd”) HMM algorithm. The QDB (Query-Dependent Banding) filter round always uses the CYK CM algorithm. The final round uses the Inside (“ins”) CM algorithm by default, but the CYK algorithm can be used instead with the **cyk** option.
- ‘**cfg**’ : configuration of the model during each round. In this case they are all locally configured. Global configuration is enabled with the **-g** option.
- ‘**beta**’ :- tail loss probability for the Query-Dependent Banding (QDB) calculation which uses the probabilistic query CM to precalculate regions of the dynamic programming lattice that have negligible probability, independently of the target database. This is the amount of probability mass allowed outside each band on the DP matrix.
- ‘**cutoffs**’ :- is the cutoffs used for each round of searching, in **E value** and in **bitscore (bit sc)**, a log-odds score in log base two. The predicted survival fraction (**surv**) and running time (**run time**) for each round are based on the E-value cutoffs.

Similarly the misalignments in other 12 ncRNA search experiments were given by the Infernal tool and perl script was designed to calculate those misalignments as summarized in the table. The results are further summarized from the original outputs given by Infernal and RNATOPs tools as given in Appendix B and D. The BS, TSS and PA are the abbreviations for Single base-stem, Total Stem-Stem and Partial alignments respectively.

Table 4.1a Result summary of ncRNA search with RNATOPS and Infernal. The table shows summary of first eight out of 13 organisms.

Organism	RNASTRAND ID	Genome (Accession)	RFAM ncRNA type	True ncRNA positions (BLAST)	RNATOPS ncRNA positions	Infernal ncRNA positions	Infernal misalignments
<i>S.cerevisiae</i>	PDB_01157	Cricket paralysis virus nonstructural polyprotein (AF218039.1)	IRE5 Cripavirus (RF00458)	6030-6219	6027-6215	Plus; 6028-6228	BS=58; TSS=3, PA=5
<i>Plautia stali intestine virus</i>	PDB_01129	Plautia stali intestine virus RNA (AB006531.1)	IRE5 Cripavirus (RF00458)	6003-6146	6002-6191	Plus; 6003 - 6204	BS=73; TSS=15, PA=5
<i>Pyrococcus horikoshii</i>	ASE_00253	Pyrococcus horikoshii OT3 DNA (BA000001.2)	RNaseP_arch (RF00373)	168172-168500	Filtered: 168471-168496	Plus; 168175 - 168497	BS=147; TSS=11; PA=12
<i>Pyrococcus furiosus</i>	ASE_00249	Pyrococcus furiosus COM1 (CP003685.1)	RNaseP_arch (RF00373)	1382446-1382117	Filtered: 527680-527705	Minus; 1382443 - 1382120	BS=114; TSS=4; PA=6
<i>Methanococcus jannaschii</i>	ASE_00196	Methanocaldococcus jannaschii DSM 2661 (L77117.1)	RNaseP_arch (RF00373)	643507-643746	Filtered: 643428-643762	Plus; 643504 - 643761	BS=117; TSS=7; PA=4
<i>Archaeoglobus fulgidus</i>	ASE_00009	Archaeoglobus fulgidus, strain DSM 4304 (AE000782.1)	RNaseP_arch (RF00373)	86275-86047	Filtered: 2092331-2092356	Minus; 86278-86044	BS=115; TSS=10; PA=6
<i>Thermotoga maritima</i>	ASE_00424	Thermotoga maritima MSB8 (AE000512.1)	RNaseP_bact_a (RF00010)	752885 - 753222	NIL	Plus; 752885 - 753222	BS=145; TSS=9; PA=5

Table 4.1b Result summary of ncRNA search with RNATOPS and Infernal. The table shows summary of remaining five out of 13 organisms

Organism	RNASTRAND ID	Genome (Accession)	RFAM ncRNA type	True ncRNA positions (BLAST)	RNATOPS ncRNA positions	Infernal ncRNA positions	Infernal misalignments
<i>Deinococcus radiodurans</i>	ASE_00101	Deinococcus radiodurans, strain R1 (AE000513.1)	RNaseP_bact_a (RF00010)	904571- 905017	NIL	Plus; 904571 – 905017	BS=194; TSS=12; PA=4
<i>Thermus thermophilus</i>	ASE_00428	Thermus thermophilus HB27 (AE017221.1)	RNaseP_bact_a (RF00010)	695567- 695957	NIL	Plus; 695570 - 695939	BS=215; TSS=21; A=6
<i>Bacillus subtilis</i>	ASE_00032	Bacillus subtilis BEST7003 DNA, complete genome (AP012496.1)	RNaseP_bact_b (RF00011)	2163512-2163112	NIL	Minus; 2163505 - 2163125	BS=81; TS=3; PA=10
<i>Tetrahymena thermophila</i>	PDB_00082	Tetrahymena thermophila strain ATCC 30382 18S ribosomal RNA (JN547815.1)	Intron_gPI (RF00028)	2960 - 3116	Filtered: 2831-3596, No Whole structure search	Plus; 2848 - 3189	BS=171; TSS=10; PA=3
<i>Homo Sapiens</i>	PDB_00908	Azoarcus sp. BH72 (AM406670.1)	Intron_gPI (RF00028)	3559264 - 3559459	2141126-2142199, no whole structure	3559255 – 3559454	BS=125; TSS=8; PA=6
<i>Hepatitis Delta Virus</i>	PDB_00335	Hepatitis delta virus isolate 59045-CAR delta antigen gene (JX888110.1)	HDV_Ribozyme	687-731	684-768	Plus; 685-773	BS=27; TSS=3; PA=1

CHAPTER 5

DISCUSSION

The ncRNA search against the genome was performed using the two tools RNATOPS and Infernal which used heuristic techniques to accurately search RNA pseudo-knotted structure. The performances of each tool were analyzed.

It was observed that RNATOPS apparently detected ncRNA with less computation time due to HMM filter function incorporated by the tool, by speeding up the whole structural search. All applicable combinations of stem, including the pseudoknot pairs, were considered by RNATOPS, through a graph model and tree decomposition-based dynamic programming. Out of the 13 ncRNA search experiments, RNATOPS almost produced an invalid alignments with ten genomes due to strong substructure found in the training data. For instance, when compared to true ncRNA positions obtained from BLAST, RNATOPS yielded invalid alignments of RNaseP_arch ncRNAs with genomes of *Pyrococcus horikoshii*, *Pyrococcus furiosus* COM1 and *Archaeoglobus fulgidus*, and the invalid alignments of Intron Group1 ncRNAs with genomes of *Tetrahymena thermophile* and *Azoarcus*, thereby giving filtered hits without performing the whole structure (Table 4.1a and b). It was further interpreted that the stem in the RNA was assumed to contain non-canonical base pairing due to which the candidates were not accurately identified. RNATOPS was interpreted to perform a local search where its heuristic algorithm is assumed to produce only k pairs of candidate regions for each individual stem in the structure to align to and for small k , the real candidate of the stem were not included and perhaps brought the inaccuracy to the search result [4]. In addition to it the current version of RNATOPS used in the above

experiments apparently failed to search ncRNA of interest in the target genome possessing different structure compared to those in the training set. For example in the above experiments, RNATOPS failed to search ncRNA belonging to RNaseP_bact_a family when tested in genomes of *Thermotoga maritime*, *Deinococcus radiodurans*, *Thermus thermophiles* and the ncRNA belonging to RNaseP_bact_a class when tested in the *Bacillus subtilis* genome.

```

Drosophila_C_virus.1          GUUAAGAUGUGAUCUUGCUUCCUU..AUACAAUUUUGAGAGGUUAAUUAAG
Black_queen_cell_vir.1      CCAACAAUGUGAUCUUGCUUGCGGA.GGC AAAUUUGCACAGUAAAAAU
Himetobi_P_virus.1         GAAAAUGUGUGAUCUGAUUAGAAG..UAAGAAAAUCCUAG.UUUAAAAU
Rhopalosiphum_padi_v.1     AGUGUUGUGUGAUCUUGCGCGAU.....AAAUGCUGACG...UGAAAA
Triatoma_virus.1           UUGACUAUGUGAUCUUGCUUUCG...UAAUAAAAUUCUGUACUAAAAAG
Plautia_stali_intest.1     CUGACUAUGUGAUCUUAUAAAAUUAGGUUAAAAUUCGAGGUAAAAAUA
#=GC SS_cons                <<<<<<<<.....<<<<<<<<......AAAAAA...... 18

Drosophila_C_virus.1          AAGGAAGUAGUGCUAUCUUAU.AAUUAGGUUAACUAAUUUAGUUUUACUG
Black_queen_cell_vir.1      CUGCAAGUAGUGCUAUUGUUGG.AAUCACCGUACCUAAUUUAGGUUUACGC
Himetobi_P_virus.1         UUUUUAAUACUGCUACAUUUUU.AAGACCCUAGUUAAUUUAGCUUUACCG
Rhopalosiphum_padi_v.1     CGUUGCGUAUUGCUACACACU....UGGUUAGCUAAUUUAGCUUUACUA
Triatoma_virus.1           UCGAAAGUAUUGCUAUAGUUAAGGUUGCGCUUGCCUAAUUUAGGCAUACU
Plautia_stali_intest.1     GUUUUAAUUGCUAUAGUCUU.AGAGGUCUUGUAUAAUUUACUUACCA
#=GC SS_cons                >>>>>>>>.....>>>>>>>>......BBB.<<<<.....>>>>>>>>.....< 5

Drosophila_C_virus.1          UUCAGGAUGCCUUAU.UGGCAGCCCCA.UAA.UAUCCAGGACAC.CCUCUC
Black_queen_cell_vir.1      UCCAAGAUCGGUGGAUAGCAGCCCUAUCAA.UAUCUAGGAGAA.CUGUGC
Himetobi_P_virus.1         CCCAGGAUGGGGUG.CAGCGUCCUG.CAA.UAUCCAGGGCAC..CUAGG
Rhopalosiphum_padi_v.1     AUC AAGACGCCGUC.GUGCAGCCAC.AAAA.GUCUAGAU...CGUCA
Triatoma_virus.1           CUCAGGAUGGCGCG.UUGCAGUCAA.CAAG.AUCCAGGGACUGUACAGA
Plautia_stali_intest.1     CACAAGAUGGACCG.GAGCAGCCUC.CAA.UAUCUAGUGUAC..CCUCG
#=GC SS_cons                <<<. <<<<<<<bbbb<<<<.....>>>.>.....>>>>>>>>.....>>>>>>>>......a.aaaaa 12

Drosophila_C_virus.1          UGCUCUUUAUUGAUUAGGUUGUCAUUUAGAA..UAAGAAAAUAACCGC
Black_queen_cell_vir.1      U.AUGUUUAGAAGAUUAGGUAGUCUCUAAACA..GAACAAUUUACCGC
Himetobi_P_virus.1         UGCAGCCUUGUAGUUUAGUGGACUUUAGGCU..AAAGAAUUUACUAGC
Rhopalosiphum_padi_v.1     CAGGAGAGCAUACGCUAGGUCGCGUUGACUAUCCUUUAUUAU.GACCGC
Triatoma_virus.1           AUUUUCC.UAUACCUAGAGUCGGUUU.GGAA..UCUAAGGUUGACUCGC
Plautia_stali_intest.1     UGCUCGCUCAAACAUUAAGUGGUUGUGCGA..AAAGAAUCUACUUA
#=GC SS_cons                ...<<<<<...<<<<<<<<CCCC>>>>>>>>.....>>>>>>>>.....cccc.. 9

Drosophila_C_virus.1          UAACUUCAA
Black_queen_cell_vir.1      UGAACAAAU
Himetobi_P_virus.1         AAUUAUAAU
Rhopalosiphum_padi_v.1     AAUUAUAAAC
Triatoma_virus.1           UGUAAUAAU
Plautia_stali_intest.1     AGAAAAAGAA
#=GC SS_cons                ..... 0

```

Figure 5.1 ncRNA Multiple Alignment of IRES Cripavirus Family in Stockholm Format. The test in red, blue and green determines the first, second and third pseudknot in the subsequence neglected by Infernal.

Since Infernal implements the covariance model giving repetitive tree like SCFG architecture, it has the property to neglect the pseudoknot pairs from the input consensus structure. This was further verified when CM of that particular family contained 44 match pairs while the observed true structure contained 60 base pairings (44 non pseudoknots + 16 pseudoknot pairs). The consensus sequence and structure obtained from Infernal were further aligned with the genome structure and sequence, retrieved from RNASTRAND and PDB sources and it was observed that Infernal produced misalignments at three levels namely single base-stem, total stem-stem and partial alignments, respectively. With respect to the computation time Infernal's CM consumed longer computation time according to the complexity of the consensus secondary structure. However the approach of HMM filter, along with banded CYK algorithm incorporated in the tool, reduced the computational time and accelerates the ncRNA search. Unlike RNATOPS, Infernal successfully detected ncRNAs in all the 13 genomes belonging to its corresponding ncRNA family, irrespective of its structural diversity. Also it was interpreted that the CM-CYK based programs performed global ncRNA search which results in complete and valid alignments.

CHAPTER 6

CONCLUSIONS

Based on the experiments and analysis of the ncRNAs searches against the 13 genomes, it was interpreted and concluded that although both the tools RNATOPS and Infernal possessed limitations, Infernal performed better than RNATOPS with respect to successfully detecting ncRNAs in all the genomes irrespective of its computation time. Newer version of Infernal is observed to have incorporated improvement in their algorithms which utilizes two rounds of search and upon using the HMM filter the ncRNA search is accelerated reducing the computational time. Even though CM neglected the pseudoknot base pairs due to its intrinsic property of SCFG tree architecture, it successfully produced hits of ncRNA positions, overlapping to those of the true positions (Table 4.1), thereby increasing the sensitivity.

APPENDIX A

STOCKHOLM FORMAT OF THE TRAINING SET

Training set containing the RNA multiple alignment of the RNaseP_arch family is shown here.

```

# STOCKHOLM 1.0
#=GF ID      RNaseP_arch
#=GF AC      RF00373
#=GF DE      Archaeal RNase P
#=GF AU      Griffiths-Jones SR
#=GF GA      53.0
#=GF NC      52.8
#=GF TC      53.2
#=GF SE      Brown JW, The Ribonuclease P Database, PMID:9847214
#=GF SS      Published; PMID:9847214
#=GF TP      Gene; ribozyme;
#=GF BM      cmbuild -F CM SEED; cmcalibrate --mpi -s 1 CM
#=GF BM      cmsearch -Z 274931 -E 1000000 --toponly -g CM SEQDB
#=GF DR      URL; http://jwbrown.mbio.ncsu.edu/RNaseP/home.html
#=GF DR      SO:0000386 SO:RNase_P_RNA
#=GF DR      GO:0008033 GO:tRNA processing
#=GF DR      GO:0004526 GO:ribonuclease P activity
#=GF DR      GO:0030681 GO:multimeric ribonuclease P complex
#=GF RN      [1]
#=GF RM      9759486
#=GF RT      Ribonuclease P: unity and diversity in a tRNA processing ribozyme.
#=GF RA      Frank DN, Pace NR;
#=GF RL      Annu Rev Biochem 1998;67:153-180.
#=GF RN      [2]
#=GF RM      9847214
#=GF RT      The Ribonuclease P Database.
#=GF RA      Brown JW;
#=GF RL      Nucleic Acids Res 1999;27:314.
#=GF CC      Ribonuclease P (RNase P) is a ubiquitous endoribonuclease, found
#=GF CC      in archaea, bacteria and eukarya as well as chloroplasts and
#=GF CC      mitochondria. Its best characterised activity is the generation
#=GF CC      of mature 5'-ends of tRNAs by cleaving the 5'-leader elements of
#=GF CC      precursor-tRNAs. Cellular RNase Ps are ribonucleoproteins.
#=GF CC      RNA from bacterial RNase Ps retains its catalytic
#=GF CC      activity in the absence of the protein subunit, i.e. it is a
#=GF CC      ribozyme. Isolated eukaryotic and archaeal RNase P RNA has not
#=GF CC      been shown to retain its catalytic function, but is still essential for
#=GF CC      the catalytic activity of the holoenzyme. Although the archaeal and
#=GF CC      eukaryotic holoenzymes have a much greater protein content than
#=GF CC      the bacterial ones, the RNA cores from all the three lineages are
#=GF CC      homologous -- helices corresponding to P1, P2, P3, P4, and P10/11
#=GF CC      are common to all cellular RNase P RNAs. Yet, there is
#=GF CC      considerable sequence variation, particularly among the
#=GF CC      eukaryotic RNAs.
#=GF WK      http://en.wikipedia.org/wiki/RNase_P
#=GF SQ      70

#=GS M.sedula.1          AC      AF121773.1/1-303
#=GS Sulfolobus_tokodaii_.1 AC      BA000023.2/326017-326320
#=GS S.acidocaldarius.1 AC      L13597.1/422-736
#=GS M.formicicum.1      AC      AF121774.1/9-309
#=GS M.thermautotrophicus.3 AC      AF192356.1/9-302
#=GS T.litoralis.1      AC      AF192365.1/1-317
#=GS Picrophilus_torridus.1 AC      AE017261.1/818125-818417
#=GS Thermoplasma_volcani.1 AC      BA000011.4/537155-537457
#=GS uncul.archa.ER-E.1 AC      AF192352.1/6-318
#=GS marine_metag.18    AC      AACY022727466.1/349-648

M.sedula.1              UAGGGGAGCCUACAGGGGGCCA.CGG.....
Sulfolobus_tokodaii_.1 ..GGGAGCCCUAGAUUGGGCUA.CGG.....
S.acidocaldarius.1     UAGGGGAGCCUACAGGGGGUUA.CGGA.....
M.formicicum.1         .....AGCCGAAGGGCAGCUAC.CGGUUUCUAUAGAUUUAAUGUCUGUA
M.thermautotrophicus.3 .....AGCCGAAGGGCAGCUGA.CGCCCCAU.....
T.litoralis.1         .....GGGGGCU.GGGGCCCU.CGGGUA.....
Picrophilus_torridus.1 ...AAAAGCCCGAGGGCAACCGA.CGCCG.....
Thermoplasma_volcani.1 UGAGAAAGCACGAGGGCAACUGA.CGCC.....
uncul.archa.ER-E.1     ...GCAAGCCGAAGGGCAGCUAC.CGGUUUCUAUAGAUUUAAUGUCUGUA
marine_metag.18       ...GGGAGUGGGAGGACCGCUGA.CAGAGC.....

```



```

M.formicicum.1 .....GAU.....
M.thermautotrophicus.3 .....UCA.....
T.litoralis.1 .....AAG.GCU.....
Picrophilus_torridus.1 .....AAA.....
Thermoplasma_volcani.1 .....GAA.....
uncul.archa.ER-E.1 .....UCA.....
marine_metag.18 .....GAA.....
#=GC SS_cons .....<<.....
#=GC RF .....gaa.ccg.....

M.sedula.1 UCG..GCGA..CGGAUGG.....GAGGGUGUGAGAGA.....
Sulfolobus_tokodaii_.1 ACCCAAGUAAUUGGGUUAC.....CUAAAUGAGAGC.....
S.acidocaldarius.1 CUAG..GUAA..CUAGGCU.....ACAUAAAUGAG.....
M.formicicum.1 .....CCG.....UCUGAUUGA.....
M.thermautotrophicus.3 .CCC..UUAA..AGGA.....GCAGACUGA.....
T.litoralis.1 CCCG..GCGA..CGGGAGCUGAGUUAACCCGACAGCAAU.....
Picrophilus_torridus.1 .....A.....GAUGACGU.....
Thermoplasma_volcani.1 .....GG.....GGAGACGUU.....
uncul.archa.ER-E.1 .CCC..UCAA..GGA.....GCAGACUGA.....
marine_metag.18 .....AG.....ACGGAGGUU.....
#=GC SS_cons .<<<.....>>>.>>.....>>.....
#=GC RF cCcc..GuaA..ggGgcg.....gcuGAaaaa.....

M.sedula.1 .....CCUA.....
Sulfolobus_tokodaii_.1 .....UAUC.....
S.acidocaldarius.1 .....ACCUA.....
M.formicicum.1 .....CACCA.....
M.thermautotrophicus.3 .....CACCA.....
T.litoralis.1 .....CCCG.....
Picrophilus_torridus.1 .....UCCC.....
Thermoplasma_volcani.1 .....UCUG.....
uncul.archa.ER-E.1 .....CAUCA.....
marine_metag.18 .....UCCA.....
#=GC SS_cons .....>>>>.....
#=GC RF .....aucgg.....

M.sedula.1 U.C.GU..GGGUUGAAACG.GCAG...AUCUCCCC.UUGAGCAAGU...
Sulfolobus_tokodaii_.1 A.U.GC..CAGUUGAAACG.GUAG...UCUCUCCU..GGAGCAAGU..G
S.acidocaldarius.1 U.U.AUACCGGCGUGAAACG.GCAG...UCCUCCCA..GGAGCAAGU..A
M.formicicum.1 G.G.AGGAACGGUGAAACG.GCCA...AUCCACGG..GAUGCAAGGUA
M.thermautotrophicus.3 G.G.AGGACCGGUGAAACG.GCCA...UUCGCGG..GAUGCAAGGACA
T.litoralis.1 A.G.GGGAGCGGUGAAACG.GCCG...UCCC GCGG..GGUGCAAGGCCG
Picrophilus_torridus.1 C.G.GCAGCCGAUGAGAA..CUCU...CCCCGCAU..GGAGCAAGUCUA
Thermoplasma_volcani.1 G.G.AGAUCCGAUGGGAA..GCCU...UCCUGGGU..GGAGAAAGUCUA
uncul.archa.ER-E.1 G.G.AGGACCGGUGAAACG.GCCA...UUCACGG..GAUGCAAGGACA
marine_metag.18 C.G.UGG.UCGAUGAGACG.AGCA...ACCCCAUG..GGAGCAAGCCCA
#=GC SS_cons >.>.>>.>>>.....>>>>.....>>>>.....>>>>.....<<<<.....
#=GC RF g.g.ggaacgGaUGAAACG.gCcg...uCCccccg..GGuGCAAGucca

M.sedula.1 AG.....GGGAG...GAU..AGG.GCAAAUUA.....
Sulfolobus_tokodaii_.1 GG.....GAAAA...GAUGAGAA.GGGACCCGA.....
S.acidocaldarius.1 AG.....GAGGG...GAUGAGUU.GAGGUUCA.....
M.formicicum.1 AACA.....CUGCC...AGUGAAUC..CUGUAUGA.....
M.thermautotrophicus.3 AAUA.....CUGCC...UGUGAUUA..CUGUAGGA.....
T.litoralis.1 AGAUA.....GGGGC...UAUGAGUCCCCGGUGUGA.....
Picrophilus_torridus.1 AGAU.....GGGCC...CGUGAACA.CCAUCCCGA.....
Thermoplasma_volcani.1 AAAU.....GGCCC...UGUGAAGC..UGCCGUUA.....
uncul.archa.ER-E.1 AAUG.....CUGCC...UGUGAUUA..CUGUAGGA.....
marine_metag.18 GACA.....GCCCU...GACGACGCGGCACGCCGA.....
#=GC SS_cons .....<<<<<.....bbbbbb.....
#=GC RF Aa.....ggGgc...gauGagaa.aagguaugA.....

M.sedula.1 .....CU.CCC..CUGA.....
Sulfolobus_tokodaii_.1 .....UU.UUC.CCCGA.....
S.acidocaldarius.1 .....CC.CUC.CUAAG.....
M.formicicum.1 .....GG.CAG..AGGU.....
M.thermautotrophicus.3 .....GG.CAG..AGGU.....
T.litoralis.1 .....GC.CCC.GUGGU.....
Picrophilus_torridus.1 .....GG.CCC..AGGU.....
Thermoplasma_volcani.1 .....GG.GCC..GGGU.....
uncul.archa.ER-E.1 .....GG.CAG..AGGU.....

```

```

marine_metag.18 .....GG.GGC..GGGU
#=GC SS_cons .....>>>>.....
#=GC RF .....gc.Ccc..aggu

M.sedula.1 U..ACGCAGAGCCUAAUCCCCCA.....
Sulfolobus_tokodaii_.1 G..ACGCUAAGUCAAAUCCCAA.....
S.acidocaldarius.1 G..ACGCUUAGUAGAAUCCCCU.....
M.formicicum.1 AACUCGCAUAGAUGAAUGCUGCC.....
M.thermautotrophicus.3 AGUCCGCCGAGAUGAAUGCUGCC.....
T.litoralis.1 AGGCCGCUCAGUCGAAUGCCCCA.....
Picrophilus_torridus.1 AAGACGCAUAGUCGAAUGUUGCC.....
Thermoplasma_volcani.1 GAGACGACUAGUCGAAUGUUGCC.....
uncul.archa_ER-E.1 AGUCCGCCGAGAUGAAUGCUGCC.....
marine_metag.18 AGGGUGCACAGUUGAAUGCUGUC.....
#=GC SS_cons .....>>>>.....>>>>>>.....
#=GC RF .....aggaCGCuuAGucGAAUGCcgcc.....

M.sedula.1 .....AG..UACAGAAGCUGGGUU
Sulfolobus_tokodaii_.1 .....AA..UACAGAAGCCGGGUU
S.acidocaldarius.1 .....AAA..UACAAAAGCUGGGUU
M.formicicum.1 .....ACC..AACAGAAGGUGGGUU
M.thermautotrophicus.3 .....GA..AACAGAAGGUGGGUU
T.litoralis.1 .....UUAA..UACAGAAGGCGGGCU
Picrophilus_torridus.1 .....AGCCUUAAGGUG..AACAGAAGGGGGCUU
Thermoplasma_volcani.1 .....AGCCCGAAAGGGUG..AACAGAAGGGGGCUU
uncul.archa_ER-E.1 .....G..AACAGAAGGUGGGUU
marine_metag.18 .....ACCGUUUUUCGGG..AACAGAAGGGGGCUU
#=GC SS_cons .....aaaaaaaa
#=GC RF .....aaa..aACAGAAGGgGGcUU

M.sedula.1 AUUGU.UAGGCUCCC.....C
Sulfolobus_tokodaii_.1 AUGCU.AGGGCUCCC.....
S.acidocaldarius.1 AUUGU.UAGGCUCCC.CUUAU
M.formicicum.1 A..CUCUCGGCA.....
M.thermautotrophicus.3 A..CUCUCGGCA.....
T.litoralis.1 A.UAG.CCCCU.....
Picrophilus_torridus.1 A..CUCCGGGCAUUU.....
Thermoplasma_volcani.1 A..CUCCGUGGAUUCUCAUCU
uncul.archa_ER-E.1 A..CUCUCGGCAUGC.....
marine_metag.18 A..CUCCCCACUCCC.....
#=GC SS_cons .....>>>>>>>>>>>>.....
#=GC RF A..cucccgCAccC.....
//

```

APPENDIX B

OUTPUTS GIVEN BY RNATOPS TOOL

The original output produced by RNATOPS for different ncRNA searches against each of the 13 genomes.

```

*****
* Filtering Result *
* *
* Profile file : IRES_Cripavirus-2_Pasta *
* Profile length : 210 *
* *
* Filter generation: automatic seleted *
* Filter type : HMM *
* Filter info : positions from 107 to 137 *
* Genome file : Plautia_stali.txt *
* Number of sequences : 1 *
* Total length of sequences : 8797 *
*****

```

Filtering hit 1

```

>gi|2344756|dbj|AB006531.1|(6108-6135)[5991-6206]
Plus search result
Hit Positions: 6108-6135
Alignment score = 13.0547
Alignment to the filter
TGG-A-CCGGAGCAGCCCTCCAATATCTAG
mmmdidimmmmmmmmmmmmmmmmmmmmmmmmmmmmmmm
Extension positions: 5991-6206
Extension of the hit
GAAGAAGAAAGCTGACTATGTGATCTTATTAATAATTAGGTTAAATTCGAGGTTAAAAAT
AGTTTTAATATTGCTATAGTCTTAGAGGTCTTGTATATTTATACTTACCACACAAGATGG
ACCGGAGCAGCCCTCCAATATCTAGTGTACCCTCGTGCCTCGCTCAAACATTAAGTGGTGT
TGTGCGAAAAGAATCTCACTTCAAGAAAAGAATTT

```

```

*****
* Searched done : with RNATOPS V1.1 *
* By RNA-Informatics @ UGA *
* Total no of hits: 1 *
* Total time used 2.22222e-05 hours *
* Time: 20:24:15 EDT 2013-04-30 *
*****

```

```

*****
* Whole Structure Result *
* *
* Profile file : IRES_Cripavirus-2_Pasta *
* Profile length : 210 *
* *
* Genome file : Plautia_stali.txt *
* Number of sequences : 1 *
* Total length of sequences : 8797 *
* *
* Search parameters setting: *
* Pseudocount = 0.001 *
* Num of stem candidates = 10 *
* Score threshold for hits = 0 *
* Num of nt overlap between stems = 2 *
* Candidate representatives only = Yes *
* Shortest candidate representatives = Yes *
* IShiftNumMergeCand = No *
* Nts allowed in null loops = 3 *
* Pcoeff = 2 *
* Search with jump = Yes *
* Search step size = 1 *
* Search reversed complement sequence = No *
*****

```

Whole structure search hit 1

```

gi|2344756|dbj|AB006531.1|
Plus search result

```

Hit Positions: 6002-6191
Alignment score = 77.7878

Folded structure

```
1 EEEEEEEEE.....DDDDDDDD.....AAAAA.....DDDDDDDD. 61
1 11111111.....11111111.....111111.....11111111. 61
6003 CTGACTATGTGATCTTATATAAATTAGGTTAAATTCGAGGTTAAAAATAGTTTTAATAT 6063

61 ...EEEEEEEE...BBB..FFF...FFF....IIII.HHHHHBBBGGG... 121
61 ...11111111...1111..1111...1111...1111.111111111111... 121
6063 TGCTATAGTCTTAGAGGCTTGTATATTTATACTTACCACACAAGATGGACCGGAGCAGC 6123

121 .GGGG..HHHH.HIII.AAAAAA...KKKK...JJJJ.CCCCCJJJ..KKKK... 181
121 .1111..1111.1111.111111...1111...1111.1111111111..1111... 181
6123 CCTCAAATATCTAGTGTACCCTCGTGCCTCAAACATTAAGTGGTGTGTGCGAAAG 6183

181 ....CCCC 191
181 .....1111 191
6183 AATCTCACTT 6193
```

Structure alignment

```
1 EEEEEEEEE.....DDDDDDDD....---.....AAAAA.....-.DDDDDDDD 61
1 11111111.....11111111.....111111.....11111111 61
6003 CUGACUAUGUGAUCUUAUAAAAU-UAGGUUAAAUUUCGAGG-UAAAAAUAGUUUUAU 6063

61 D...EEEEEEEE.--.BBB..FFF...FFF....IIII.HHHHHBBBGGG 121
61 1...11111111.....1111..1111...1111...1111.111111111111 121
6063 AUUGCUAUAGUCUUAUA--GGUCUUGUAUUAUUUAUCUACCACACAAGAUGGACCGGAG 6123

121 ....GGGG..HHHH.HIII.AAAAAA...KKKK...JJJJ.CCCCCJJJ..KKKK 181
121 ....1111..1111.1111.111111...1111...1111.1111111111..1111 181
6123 CAGCCCUCAAUAUCUAGUGUAC-CCUCGUGCUCGCUCAACAUAAGUGGUGUGGCG 6183

181 K.....CCCC 196
181 1.....1111 196
6183 AAAAGAAUCUCACUU 6198
```

```
*****
* Searched done : with RNATOPS V1.1 *
* By RNA-Informatics @ UGA *
* Total no of hits: 1 *
* Total time used 0.00308333 hours *
* Time: 20:24:26 EDT 2013-04-30 *
*****
```

2.

```
*****
* Filtering Result *
* *
* Profile file : RNaseP_arch_Pasta *
* Profile length : 721 *
* *
* Filter generation: automatic selected *
* Filter type : HMM *
* Filter info : positions from 687 to 714 *
* Genome file : Pyrococcus_horikoshii_OT3.txt *
* Number of sequences : 1 *
* Total length of sequences : 1738505 *
*****
```

Filtering hit 1

```
-----
>gi|47118297|dbj|BA000001.2|(51520-51545)[51217-51554]
Plus search result
Hit Positions: 51520-51545
Alignment score = 0.280348
Alignment to the filter
CAAATGCTGGCTTACGCAACGAATCC
```


TGAATCTGTAACCTCTTCCAATTCCTTTTTATCAACTCTAGCAATATCAGCAATCTCAT
CCAGAGTCCTGGGAACCTTTAATAACCTACAAGCAGCGTAAACACATGCCGCCATAACGC
TCTCAATAGATCTACCTAATAAGTCCCTTTCTCACTGCCTCTGTACAGCCTTGCAG
CTTCTTCTCTACATGTCTTGGAAAGTTTAACTGAGCAGTAATCTATCCAACCTACTTA
GGCAAAGCTAGGTTCTCTCTGCTGCATCACTA

Filtering hit 6

>gi|393188278|gb|CP003685.1|(1218880-1218903)[1218577-1218912]
Plus search result
Hit Positions: 1218880-1218903
Alignment score = 0.246043
Alignment to the filter
CAGAAGCGGCC-GC-TGGCAGGACA---
mmmmmmmmmmmmmdiidmmmmmmmmmmddd
Extension positions: 1218577-1218912
Extension of the hit
GATATAAATCGGGCCCTTCCACCTGGAGTCAAAAATCTATCGGTATACAACCTGGGTGAT
GGAATTACAATGCCCTCAGAGTTTTTCAGTTCCTCTACACTCCTATTTTCAAGTCTGGG
AATAACTAAAGTATTCCTCTGTATTTCCTCAACCCTTTTGTAGTTAAAACCCCTCAAC
CCAAGAGCATTGCCAACATTGTTAGAATCTCCAATCTGGCTTGCTCTCCCTAGTGGT
TCGCAAGCTTTATGACTCCACTGTATTCTCCTCTCGCTGTTTCATGTAAGAACCTTCCTTC
TCACAGAAGCGGCCGCTGGCAGGACATAGTGAGCG

Filtering hit 7

>gi|393188278|gb|CP003685.1|(1751878-1751896)[1751575-1751905]
Plus search result
Hit Positions: 1751878-1751896
Alignment score = 0.717729
Alignment to the filter
CAGAAGGT---TFACT-ACTCCT---
mmmmmmmmdddmmmmmmmmmmddd
Extension positions: 1751575-1751905
Extension of the hit
GGCTATACAAAGAGCCACCAAGTTATAGTGACGTAAGTGACCTACACCCAGCCGGTGA
GGATGGGCTGCTCCACTTCTTGAAGAAGCTGGAATAATCCTCAACAAGAACCTACTTCCA
TGGGATCCACTTGAAAAAGTCAACGAGCCAGTGGATTAAGAATTGGAGTTCAAGAGATG
ACAAGAGTTGGAATGATGGAAGATGAAATGAGAGAAATCGCCCACTTCATCAAGAGAGTT
CTAATAGATAAGGAAGATCCAAAGAAGGTCAGAAAGGATGTCTACTACTTCAGGCTAGAG
TACCAGAAGGTTTACTACTCCTTCGACTATG

Filtering hit 8

>gi|393188278|gb|CP003685.1|(1795843-1795862)[1795540-1795871]
Plus search result
Hit Positions: 1795843-1795862
Alignment score = 0.484224
Alignment to the filter
CAGAAAGGG-----CT-CCTCCTCCC
mmmmmmmmmmdddmmmmmmmmmm
Extension positions: 1795540-1795871
Extension of the hit
CGGGAAGGAATATCCCAGCCGGGAATCATTCTCAGTATTTGTTTTAGGGGCCCATTT
TTCTCATCGCTTCAAGCTGAGCATACATGTCTTAAGTGAATTTACCCCTTAAGAAATC
TTCAATGTCTTCTCTTTTATTTCAACTTCTTCTCTAACTCCTTAAATTTTCCAGCA
ATCCTTGAATATCTCCAAGTCCAAGAGTCTTGAAACAAATCTCGGTGGGTCAAAGGGCT
CGATATCATCTATCTTTTCTCCCGTACCTATGAATTTTATTGGAGCCCTGTTGCGGCAA
CTGCAGAAAGGGCTCCTCCTCCCTTGGCAGAT

Filtering hit 9

>gi|393188278|gb|CP003685.1|(1807216-1807234)[1806913-1807243]
Plus search result
Hit Positions: 1807216-1807234
Alignment score = 0.56657
Alignment to the filter
CAGAAGG-----ATACGCTAGGCT---
mmmmmmmmdddmmmmmmmmmmddd
Extension positions: 1806913-1807243
Extension of the hit

* The whole structure search result is not available. *

4.

* Filtering Result *
* *
* Profile file : RNaseP_arch_Pasta *
* Profile length : 721 *
* *
* Filter generation: automatic seleted *
* Filter type : HMM *
* Filter info : positions from 687 to 714 *
* Genome file : Methanocaldococcus_jannaschii.txt *
* Number of sequences : 1 *
* Total length of sequences : 1664957 *

Filtering hit 1

>gi|6626255|gb|L77117.1|(81670-81689)[81367-81698]
Plus search result
Hit Positions: 81670-81689
Alignment score = 1.46672
Alignment to the filter
CAGAAG--GGCTTAAG-AAGGGT---
mmmmmmddmmmmmmmmmmmmddmmmmmmdd
Extension positions: 81367-81698
Extension of the hit
GAAATTTAAAGTGCTATGTCTTTGATTACCAAATGTTATTGAAGAAACCAAAAAATTTA
TCAAAAAATACAATGCAAAAAACGTCTTCACAATTACTGGAGATTTTTATAAGGATGATA
TCGAAAGGGCTACGATATAATATTCTGCTCATATAATCCAGGTGGAAAAATCCAAAGA
TTGCAGAGAAGGTTTATAATGCCTTAAATGAAGGAGGTTTATTATAAATAAGCAATTCT
TTCCAGATAAAGGAAGGGTATTGAAGACTATATAACAACATGGAATGGAACCTTCTCTA
AACCAGAAGGGCTTAAGAAGGGTAAAATAAGA

Filtering hit 2

>gi|6626255|gb|L77117.1|(179247-179267)[178944-179276]
Plus search result
Hit Positions: 179247-179267
Alignment score = 0.516452
Alignment to the filter
CAAAGGTGG---ATTCT-TACT---
mmmmmmmmmmdddmimmmmmmmmmdd
Extension positions: 178944-179276
Extension of the hit
CTAAGAGATTTTAGGTAAGCAGTTAAACATTACAGATGCTTCCAAGAAGGAGAGTTAG
TCGATACAATTGGAGTTACAAAAGGTAAGGATTCGAAGGACAAGTTAAAAGATGGGGAG
TTAAAATACAATTTGGTAAGCACGCAAGAAAAGGAGTAGGAAGACACGTTGGTCTATG
GTCCATGGCAACCAAGATGGTTATGTGGAGTGTCCAATGCCAGGTCAAATGGGATACC
ACCAAAGAACTGAATACAACAAGAGAATATTAAGATTGGAAACAATGGGGATGAAATTA
CACAAAAGGTGGATTCTTACTACTACGGGGTTA

Filtering hit 3

>gi|6626255|gb|L77117.1|(643731-643753)[643428-643762]
Plus search result
Hit Positions: 643731-643753
Alignment score = 16.7975
Alignment to the filter
CAGAAGGCGGGCTATAG-CCCCA---
mmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmdd
Extension positions: 643428-643762
Extension of the hit
GCGACACCGCGGCCACTTTTTATTATTTAGAAAATAACATTTATATATTCAAATCTTA
AAGTTAAGCGGGTAAGGGGGCTGGTGACTTTCCCTCTTTAAGAGGGGAGGAAGTCCGC
CCACCCCATTTATGGGCAGCGTCCCCTGAGAAGGGGCGGGAGATGCAGCAGAAACGACAC
GGCTCCGGAAGAGATGACGATGATAGTGAAAGTTGAGGACTTCCGGAACCGGTGAAAC


```
* Shortest candidate representatives = Yes *
* IShiftNumMergeCand = No *
* Nts allowed in null loops = 3 *
* Pcoeff = 2 *
* Search with jump = Yes *
* Search step size = 1 *
* Search reversed complement sequence = No *
*****

*****
* Searched done : with RNATOPS V1.1 *
* By RNA-Informatics @ UGA *
* Total no of hits: 0 *
* Total time used 0.165964 hours *
* Time: 20:43:32 EDT 2013-04-30 *
*****
```

APPENDIX C

COVARIANCE MODEL OF IRES CRIPAVIRUS FAMILY

The covariance model (CM) was obtained by the cmbuild program of infernal. It shows nodes, states and log odds of the transition probabilities from one state to another, and emission probabilities of the nucleotides. Here the model shows 33 nodes and 102 states out of 622 states and 171 nodes, respectively.

INFERNAL-1 [1.0.2]

NAME IRES_Cripavirus
 GA 29.00
 TC 46.90
 NC 23.90
 STATES 622
 NODES 171
 ALPHABET 1
 ELSELF -0.08926734
 WBETA 1e-07
 NSEQ 6
 EFFNSEQ 2.409
 CLEN 201
 BCOM bin/cmbuildIRES/IRES_Cripavirus_new.cm
 IRES/IRES_Cricket_Cripavirus_new_stockholm.txt
 BDATE Tue Feb 12 11:50:09 2013
 NULL 0.000 0.000 0.000 0.000

MODEL:

				[ROOT	0]				
S	0	-1	0	1	4	-7.034	-8.280	-0.072	-4.733
IL	1	1	2	1	4	-2.817	-4.319	-0.613	-2.698
0.000	0.000	0.000	0.000	0.000					
IR	2	2	3	2	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000	0.000					
					[MATR	1]			
MR	3	2	3	5	3	-8.516	-0.017	-6.834	
0.387	-0.426	-1.159	0.585						
D	4	2	3	5	3	-6.390	-1.568	-0.620	
IR	5	5	3	5	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000	0.000					
					[MATR	2]			
MR	6	5	3	8	3	-8.516	-0.017	-6.834	
1.514	-1.754	-1.799	-0.835						
D	7	5	3	8	3	-6.390	-1.568	-0.620	
IR	8	8	3	8	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000	0.000					
					[MATR	3]			
MR	9	8	3	11	3	-8.516	-0.017	-6.834	
1.225	-0.890	-0.874	-0.791						
D	10	8	3	11	3	-6.390	-1.568	-0.620	
IR	11	11	3	11	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000	0.000					
					[MATR	4]			
MR	12	11	3	14	3	-8.516	-0.017	-6.834	
0.783	-1.079	-1.243	0.468						
D	13	11	3	14	3	-6.390	-1.568	-0.620	
IR	14	14	3	14	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000	0.000					
					[MATR	5]			
MR	15	14	3	17	3	-8.516	-0.017	-6.834	
1.127	-1.251	-1.373	0.013						
D	16	14	3	17	3	-6.390	-1.568	-0.620	
IR	17	17	3	17	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000	0.000					
					[MATR	6]			
MR	18	17	3	20	3	-8.516	-0.017	-6.834	
1.184	-0.804	-1.394	-0.369						
D	19	17	3	20	3	-6.390	-1.568	-0.620	

IR	20	20	3	20	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 7]

MR	21	20	3	23	3	-8.516	-0.017	-6.834
0.784	-0.560	-1.195	0.218					
D	22	20	3	23	3	-6.390	-1.568	-0.620
IR	23	23	3	23	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 8]

MR	24	23	3	26	3	-8.516	-0.017	-6.834
1.488	-1.703	-1.755	-0.758					
D	25	23	3	26	3	-6.390	-1.568	-0.620
IR	26	26	3	26	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 9]

MR	27	26	3	29	3	-8.516	-0.017	-6.834
0.694	-1.437	0.547	-0.859					
D	28	26	3	29	3	-6.390	-1.568	-0.620
IR	29	29	3	29	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 10]

MR	30	29	3	32	3	-8.516	-0.017	-6.834
0.777	-1.077	-1.242	0.475					
D	31	29	3	32	3	-6.390	-1.568	-0.620
IR	32	32	3	32	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 11]

MR	33	32	3	35	3	-8.516	-0.017	-6.834
-0.404	1.248	-1.682	-0.844					
D	34	32	3	35	3	-6.390	-1.568	-0.620
IR	35	35	3	35	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 12]

MR	36	35	3	38	3	-8.516	-0.017	-6.834
-0.984	-1.224	1.406	-1.268					
D	37	35	3	38	3	-6.390	-1.568	-0.620
IR	38	38	3	38	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 13]

MR	39	38	3	41	3	-8.516	-0.017	-6.834
-0.079	-0.520	-1.277	0.958					
D	40	38	3	41	3	-6.390	-1.568	-0.620
IR	41	41	3	41	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 14]

MR	42	41	3	44	3	-8.516	-0.017	-6.834
-0.473	0.383	-1.315	0.654					
D	43	41	3	44	3	-6.390	-1.568	-0.620
IR	44	44	3	44	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 15]

MR	45	44	3	47	3	-8.516	-0.017	-6.834
-1.590	1.646	-2.423	-1.510					
D	46	44	3	47	3	-6.390	-1.568	-0.620
IR	47	47	3	47	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

[MATR 16]

MR	48	47	3	50	3	-8.516	-0.017	-6.834
1.782	-2.647	-2.570	-2.108					
D	49	47	3	50	3	-6.390	-1.568	-0.620
IR	50	50	3	50	3	-1.925	-0.554	-4.164
0.000	0.000	0.000	0.000					

					[MATR	17]			
MR	51	50	3	53	3	-8.516	-0.169	-3.216	
0.102	-0.129	0.053	-0.036						
D	52	50	3	53	3	-6.390	-1.568	-0.620	
IR	53	53	3	53	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000						
					[MATR	18]			
MR	54	53	3	56	3	-8.366	-0.019	-6.684	
-1.095	-1.539	-1.878	1.544						
D	55	53	3	56	3	-7.332	-0.610	-1.562	
IR	56	56	3	56	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000						
					[MATR	19]			
MR	57	56	3	59	3	-8.516	-0.017	-6.834	
-0.085	-0.573	-1.290	0.982						
D	58	56	3	59	3	-6.390	-1.568	-0.620	
IR	59	59	3	59	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000						
					[MATR	20]			
MR	60	59	3	62	3	-8.516	-0.017	-6.834	
0.845	-1.081	-0.581	0.088						
D	61	59	3	62	3	-6.390	-1.568	-0.620	
IR	62	62	3	62	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000						
					[MATR	21]			
MR	63	62	3	65	3	-8.516	-0.017	-6.834	
1.131	-1.274	-0.732	-0.333						
D	64	62	3	65	3	-6.390	-1.568	-0.620	
IR	65	65	3	65	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000						
					[MATR	22]			
MR	66	65	3	68	3	-8.516	-0.017	-6.834	
1.517	-1.252	-1.812	-1.205						
D	67	65	3	68	3	-6.390	-1.568	-0.620	
IR	68	68	3	68	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000						
					[MATR	23]			
MR	69	68	3	71	3	-8.516	-0.017	-6.834	
0.363	-1.169	0.484	-0.201						
D	70	68	3	71	3	-6.390	-1.568	-0.620	
IR	71	71	3	71	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000						
					[MATR	24]			
MR	72	71	3	74	3	-8.516	-0.017	-6.834	
1.488	-1.703	-1.755	-0.758						
D	73	71	3	74	3	-6.390	-1.568	-0.620	
IR	74	74	3	74	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000						
					[MATR	25]			
MR	75	74	3	77	3	-8.516	-0.147	-3.406	
0.789	-0.545	-0.627	-0.090						
D	76	74	3	77	3	-6.390	-1.568	-0.620	
IR	77	77	3	77	3	-1.925	-0.554	-4.164	
0.000	0.000	0.000	0.000						
					[MATR	26]			
MR	78	77	3	80	2	-2.827	-0.219		
0.467	-0.995	-1.165	0.740						
D	79	77	3	80	2	-5.306	-0.037		
IR	80	80	3	80	2	-1.362	-0.711		
0.000	0.000	0.000	0.000						
					[BIF	27]			
B	81	80	3	82	362				
					[BEGL	28]			

S	82	81	1	83	1	0.000								
							[BIF	29]						
B	83	82	1	84		296								
							[BEGL	30]						
S	84	83	1	85	4	-0.042	-6.929	-6.337	-6.977					
							[MATP	31]						
MP	85	84	1	89	6	-9.181	-9.120	-0.020	-7.896	-8.176	-			
8.571	-2.315	-2.158	-2.521	1.669	-2.110	-2.791	1.475	-0.781	-2.532	1.370	-			
2.638	0.606	1.506	-2.199	-0.496	-1.866									
ML	86	84	1	89	6	-6.250	-6.596	-1.310	-1.005	-6.446	-			
3.975	0.660	-0.612	-0.293	-0.076										
MR	87	84	1	89	6	-6.988	-5.717	-1.625	-5.695	-0.829	-			
3.908	0.660	-0.612	-0.293	-0.076										
D	88	84	1	89	6	-9.049	-7.747	-3.544	-4.226	-4.244	-			
0.319														
IL	89	89	5	89	6	-2.579	-2.842	-0.760	-4.497	-5.274	-			
4.934	0.000	0.000	0.000	0.000	0.000									
IR	90	90	6	90	5	-2.408	-0.496	-5.920	-4.087	-5.193				
0.000	0.000	0.000	0.000											
							[MATP	32]						
MP	91	90	6	95	6	-9.181	-9.120	-0.020	-7.896	-8.176	-			
8.571	-3.089	-2.921	-3.380	1.469	-2.566	-3.732	1.763	-3.271	-3.383	1.614	-			
3.408	-0.652	1.945	-3.074	-0.514	-0.768									
ML	92	90	6	95	6	-6.250	-6.596	-1.310	-1.005	-6.446	-			
3.975	0.660	-0.612	-0.293	-0.076										
MR	93	90	6	95	6	-6.988	-5.717	-1.625	-5.695	-0.829	-			
3.908	0.660	-0.612	-0.293	-0.076										
D	94	90	6	95	6	-9.049	-7.747	-3.544	-4.226	-4.244	-			
0.319														
IL	95	95	5	95	6	-2.579	-2.842	-0.760	-4.497	-5.274	-			
4.934	0.000	0.000	0.000	0.000	0.000									
IR	96	96	6	96	5	-2.408	-0.496	-5.920	-4.087	-5.193				
0.000	0.000	0.000	0.000											
							[MATP	33]						
MP	97	96	6	101	6	-9.181	-9.120	-0.020	-7.896	-8.176	-			
8.571	-3.517	-2.957	-3.991	1.971	-3.415	-4.189	1.146	-3.705	-3.750	1.986	-			
3.668	0.295	1.681	-3.567	-0.777	-2.759									
ML	98	96	6	101	6	-6.250	-6.596	-1.310	-1.005	-6.446	-			
3.975	0.660	-0.612	-0.293	-0.076										
MR	99	96	6	101	6	-6.988	-5.717	-1.625	-5.695	-0.829	-			
3.908	0.660	-0.612	-0.293	-0.076										
D	100	96	6	101	6	-9.049	-7.747	-3.544	-4.226	-4.244	-			
0.319														
IL	101	101	5	101	6	-2.579	-2.842	-0.760	-4.497	-5.274	-			
4.934	0.000	0.000	0.000	0.000	0.000									
IR	102	102	6	102	5	-2.408	-0.496	-5.920	-4.087	-5.193				
0.000	0.000	0.000	0.000											

APPENDIX D

INFERNAL ORIGINAL OUTPUTS WITH ALIGNMENTS

The original output produced by the Infernal tool for search of ncRNAs in thirteen genomes. After every output is the representation of the Infernal consensus sequence-structure alignment of an ncRNA with that of the genome showing the misalignments.

Consensus sequence and structure of Infernal aligned to that of *Pyrococcus furiosus* COM1 genome of organism *Pyrococcus furiosus*. The dots (.) in the consensus sequence of infernal denote the single residues or unaligned nucleotides

```
GggAGCccgAGgGcgGcUgACgGcggaAacgCuGAGGAAAgUCCacCCuCCacgcca.A
(((.((((((((((.....(((.....

((((((((({{{{...(((.....)))).((.(.....[[][[[.
GCGAGGGGGCUGGGGGCUGUCGGGCUCGUGCCCAGGAAGUUCGCCACCGCACCGGGG

cggggaccccuGuAAGggggugggcccAGAggcccGGcaAcggcAcAGAAAcGaaACgcg
(((((((.....)))(((((.....)))((.....(((.....(((

(((((((.....)))(((((.....)))((.....(((.....(((
CCGCGGUGCCGUAAGGCACCGGCCGAGAGGCCGGCAACGGCACAGAAACGACACGUCC

ccccGaaaAu...GAUGAu.ccgaaAuaA.....g
(((((((.....((.....)))

(((((((.....(((.....(((.....))).....)).
UCGGGGAUGUGGAUGAAAGCGGUGAAGGCUCCCGUGACGGGAGCCGAGUUAACCGCA

guGAaggauCcgggggAacgGaUGAAAcGgccgaCCccccgGGaGCAAGucc.AAaaagg
).....))).....)).....)).....)).....)).....)).....)).....)).....))
.....)).....)).....)).....)).....)).....)).....)).....)).....)).....
GACAAUCCCGAGGGGAGCGGUGAAACGGCCGUCCCGCGGGGUGCAAGGCCGAGUUAGGGC

gccgaUGAaaa.aauauauGAggccc.agGuaggaCGCauAGucGAAUgCcgCc.gagaA
(((.....))).....)).....)).....)).....)).....)).....)).....)).....

(...((.[[]]]]]]])).....)).....)).....)).....}}}}}}.....
CGAUGAGUUCGCCGUGAGGCCCGUGGUAGGCCGCUAGUCGAAUGCUCGCCGUAGUACA

CAGAAGgaGGgUUA.CUCcggGCaccC
.....)).....)).....)).....))

...)).....)).....)).....)).....)).....)).....)).....)).....)).....
GAAGGCgGCUAUAGCCCCUCGCCUA
```

Infernal analysis of *Pyrococcus furiosus* COM1 genome of the organism *Pyrococcus furiosus* to test for RNaseP_arch ncRNA.

```
# cmsearch :: search a sequence database with an RNA CM
# INFERNAL 1.0.2 (October 2009)
# Copyright (C) 2009 HHMI Janelia Farm Research Campus
# Freely distributed under the GNU General Public License (GPLv3)
# -----
```


1382205 GCCGAUGAGUUcCCGGUGUGAGGCCCGUGGUAGGCCGCUUAGUCGAAUGCUCcCGUAGUA 1382146

-----.]-]]]]-]]]
263 CAGAAGgaGGgUUA.CUCcggGCaccC 288
CAGAAGG GGG UA :: C::C+C:C
1382145 CAGAAGCGGGCUAuAG-CCCCUCGC 1382120

```
#
# Post-search info for CM 1: RNaseP_arch
#
# rnd  mod  alg  cfg  beta  bit sc cut  num hits  surv fract
# ---  ---  ---  ---  ----  -
# 1  hmm  fwd  loc  -      3.00  503  0.0822
# 2  cm   cyk  loc  1e-10  0.00  79   0.0137
# 3  cm   ins  loc  1e-15  0.00  84   0.0010
#
# run time
# -----
# 00:04:58
# CPU time: 296.87u 0.01s 00:04:56.88 Elapsed: 00:04:5
```

Consensus sequence and structure of Infernal aligned to that of *Pyrococcus furiosus* COM1 genome of organism *Pyrococcus furiosus*. The dots (.) in the consensus sequence of infernal denote the single residues or unaligned nucleotides

```
GggAGCccgAGgGcgGcUgACgGcggaaAcgCuGAGGAAAgUCCacCCuCCacgccga.A
((( (((((((((((((((((...(((.....))))))..... (((.....

((((((((((((({{({{({{...(((.....))))))..... (((.....

GCGAGGGGGCUGGGGGCUGUCGGGCUCGUGCCCAGGAAGUCCGCCACCGCACCGGGG

cggggaccccuGuAAGgggguggggccgAGAgggcccGGcaAcggcAcAGAAAcGAaACcgc
((((((((((.....))))))(((.....))))))((...(((.....))))))(((

((((((((((.....))))))(((.....))))))((...(((.....))))))(((

CCGCGGUGCCGUAAGGCACCGGCCGAGAGGCCGGGCAACGGCACAGAAACGACACGUCCC

ccccGaaaAu...GAUGAu.ccgaaAuaA.....g
((((((.....)))))).....

((((((.....)))))).....

UCGGGGGAUGUGGAUGAAAGCGGUGAAGGCUCcCGGUGACGGGAGCCGAGUUAACCCGCA

guGAaggauCcggggAacgGaUGAAAcGgccgaCCccccgGgCAAGucc.AAaaagg
).....))))))..)).....))))..)).....))))..)).....))))..)).....

.....))))))..)).....))))..)).....))))..)).....))))..)).....

GACAAUCCCAGGGGAGCGGUGAAACGGCCGUCCC GCGGGUGCAAGGCCGAGUUAGGGC

gccgaUGAaaa.aauauauGAggccc.agGuaggaCGCauAGucGAAUgCcgCc.gagaA
((((.....)))).....

(...(((.....]]]]]])))))).....)))).....}}}}}}.....
```


CAGAAGG GGG UA :: C:::CA::C
643736 CAGAAGGCGGGCUAuAG-CCCCCAUAC 643761

```
#
# Post-search info for CM 1: RNaseP_arch
#
#  rnd  mod  alg  cfg  beta  bit sc cut  num hits  surv fract
#  ---  ---  ---  ---  ----  -
#  1    hmm  fwd  loc  -      3.00      9      0.0016
#  2    cm   cyk  loc  1e-10   0.00      2      0.0004
#  3    cm   ins  loc  1e-15   0.00      2      8.2e-05
#
#   run time
#  -----
#   00:00:44
//
#
# CPU time: 43.95u 0.01s 00:00:43.96 Elapsed: 00:00:44
```

The substrings **[37]** and **[4]** in the query position and target positions respectively indicate that 34 consensus residues and 4 target residues were left unaligned; the target does not appear to have the consensus structure in this region. Similarly it applies to substrings **[12]** and

Consensus sequence and structure of Infernal aligned to that of *Methanocaldococcus jannaschii* DSM 2661 genome of organism *Methanocaldococcus jannaschii*

```
AGCccgAGgGcgGcUgACgGcg...gaaAcgCuGAGGAAAgUCCacCCuCCacgcccgaAc
.((((((((((((((...((((.....)))))).....((((.....(
((((((((([[[[[[...((((((.....)))))).....(((.....(((
AGGGGGCUGGUGACUUCCCCUCUUUAAGAGGGGAGGAAGUUCGCCACCCCAUUUAUG
ggggac*[12]*gugggcccgAGAggccc.GG.caAcggcAcAGAAAcGAaACcgcccccg
((((((.....))((((.....))))).((.....((((.....(((((((
(((((.((((.....))))))((((((((.....((((((((((((((...
GGCAGCGUCCCCUGAGAAGGGGCGGAGAUAGCAGCAGAAACGACACGGCUCCGGAAGAGA
Gaa.aAuGAUGAu.ccgaaAuaAgguGAaggauCcgggggAacgGaUGAAAcGgccga..
((((.....((.....)).....))))))..)).....))))))...
.....((.....)).....))))))..)).....)))))))))
UGACGAUGAUAGUGAAAGUUGAGGACUUCGGGAGAACCGGUGAAACGGGCAUCUCCCCUG
..CCccccg..GGaGCAAGucc*[37]*ggaCGCauAGucGAAUgCcgCc..gagaACAG
..))))))..))..((.....)).....)))))).....
))))))..((.....)).....]]]]].....))))
CCCGGGUGCAAGCCGUUUCGGCGCUUAGCCGAAUGUCACCGAAAUACAGAAGCGGG
AAGgaGGgUUA.
.....
))..))))))
CUAUAGCCCCCA
```

[37] and *[12]* in the sequence of Infernal consensus structure are considered the residues which are unaligned are while aligning with the genome sequence they were mostly neglected or treated as single residues. However when aligned with a stem or a base pair it produces a misalignment.

Infernal analysis of *Thermotoga maritima* MSB8 genome of the organism *Thermotoga*

maritima to test for RNaseP_bact_a ncRNA

```
# cmsearch :: search a sequence database with an RNA CM
# INFERNAL 1.0.2 (October 2009)
# Copyright (C) 2009 HHMI Janelia Farm Research Campus
# Freely distributed under the GNU General Public License (GPLv3)
# -----
# command: bin/cmsearch RnaseP/RNaseP_bact_a.cm
RnaseP/Thermotoga_maritima.txt
# date: Fri Sep 7 12:47:19 2012
# num seqs: 1
# dbsize (Mb): 3.721450
#
# Pre-search info for CM 1: RNaseP_bact_a
#
# rnd mod alg cfg beta bit sc cut
# --- --- --- ---
1 hmm fwd loc - 3.00
2 cm cyk loc 1e-10 0.00
3 cm ins loc 1e-15 0.00

CM: RNaseP_bact_a
>gi|12057205|gb|AE000512.1|

Plus strand results:

Query = 1 - 381, Target = 752885 - 753222
Score = 230.21, GC = 68

{{{
1 cgggcccGccGGgCGGcCGcGgccccgacuauaaaggucg.ggccGAGGAAAGUCCGGaCu 59
:G::G::G::C:GGCGG:CGCGG: ::C+ A+ G::: :CCGAGGAAAGUCCGGACU
752885 GGAGAGGAGCAGGCGGUCGCGGGGGCGCAC-ACCUGCGCuUCCCGAGGAAAGUCCGGACU 752943

[[------[[[[[<<<<<<_____>>>>>><<<<<<_____>>>>->((---(((,,,,,,
60 CCacAGaGcAGGguGguGGaUAACauCCaccGggGugAccCgagGGAAAGUGCCACAGA 119
C GAGC GGUG::GG:UAAC+:CC:::GGGUGACCC: :GGA AG:GCCA AGA
752944 CUG--GAGCGGGUGCCGGUAACGCCCGGGAGGGUGACCCU-CGGACAGGGCCAUA GA 753000

,,,,,,,<<<<<<<<<<_____>>>>>>>>-->,,,,,, <<<<<<_____>>>>
120 .AAaAgACCgCCcccgcgguagcgcgggGGuAAGGGUGAAAAGuGggGUAAGAGccCa 178
AA AAGACC:CCC GU+G +GGG:AAGGGUG AA GGUGGGGUAAGAGCCCA
753001 gAAGAAGACCGCCC-----GUUG----AGGGCAAGGGUGGAACGGUGGGGUAAGAGCCCA 753051

>>, <<<<<<<<<_____>>>>>>>>->,))--))]]]]]]]]],,, <<<<-----<
179 CCAGcggccccgGuGAcggggcgcgGCuaGgcAAaCCCCacCCGGaGCAAGGCCAAAaAaG 238
```


UGGAGCGGGGUGCCGGGUAACGCCCGGGAGGGGUGACCCUCGGACAGGGCCAUAGAGAAG

.AAAaAgACCgCCccccgguagcgcgggGGuAAGGGUGAAAaGGuGggGUAAGAGccCa
.....((((((((((.....)))))))).).....((((((((.....))))))
....((((((((.....)))))).....((((((((.....))))))((((((((.....
AAGACCGCCCGUUGAGGGCAAGGGUGGAACGGUGGGGUAAGAGCCACCAGCGUCGGGGC

CCAGcgccccgGuGAcggggccgGCuaGGcAAaCCCCacCCGGaGCAAGGCCAAaAaG
).....((((((((.....)))))).....((((((((.....)))))).....((((.....(
..)))))))).).....((((((((.....)))))).....((((.....((((.....]]]]].
AACCCGGCGGCUUGGCAACCCCCACCUGGAGCAAGGCCAAGCAGGGGGUUGGGUCGCUCC

caggcguaaaugGgcugcucgcCca.AgccugCgGGUaGGCCGcuaGAGgCcgCgGcA
((((.....((((.....)))))).....)))).....)))).....((((.....(
))))).).....)))).....)))).....)))).....)))).....)))).....}}}
CCCUAUUCCCCGGGUUGGCCGCUUGAGGUGUGCGGUAACGCACACCCAGAUUGAUGAC

AcGgcgGcCccAGAuAUAUGgcCG
.))))).....)))))
}}}}.....)))))
CGCCACGACAGAAUCCGGCUUAU

Infernal analysis of Bacillus subtilis BEST7003 DNA, complete genome of the Bacillus subtilis organism to test for RNaseP_bact_b ncRNA

```
# cmsearch :: search a sequence database with an RNA CM
# INFERNAL 1.0.2 (October 2009)
# Copyright (C) 2009 HHMI Janelia Farm Research Campus
# Freely distributed under the GNU General Public License (GPLv3)
# -----
# command: bin/cmsearch RnaseP/bact_b.cm
RnaseP/Bacillus_subtilis_BEST7003.txt
# date: Tue Nov 13 18:31:36 2012
# num seqs: 1
# dbsize (Mb): 8.086084
#
# Pre-search info for CM 1: RNaseP_bact_b
#
# rnd mod alg cfg beta bit sc cut
# --- --- --- --- -----
# 1 hmm fwd loc - 3.00
# 2 cm cyk loc 1e-10 0.00
# 3 cm ins loc 1e-15 0.00

CM: RNaseP_bact_b
>gi|407962962|dbj|AP012496.1|

Plus strand results:

Query = 258 - 306, Target = 1398985 - 1399038
Score = 8.14, GC = 43

,,<<<<<<_____>>>>>>>>>>>><<<<<<.....____>>>>>>
```



```

3 cm ins loc 1e-15 0.00 619 0.0020
#
# run time
# -----
# 00:20:30
//
#
# CPU time: 1225.60u 0.04s 00:20:25.63 Elapsed: 00:20:30

```

Consensus sequence and structure of Infernal aligned to that of *Bacillus subtilis* BEST7003 DNA complete genome of organism *Bacillus subtilis*.

```

AgUuucgGguaAucGCgcgguacaU.uuguaccguGAGGAAAGUCCAUGCucGCACag.g
...(((((((((((.(.(((((.)))))))).))))))....(((((((.(
(((([[[[[[[[.((((((.....)))))))).))))....(((.(
ACGUUCGGGUAUUCGUCGAGAUCUUGAAUCUGUAGAGGAAAGUCCAUGCUCGCACGGUG

CUGaGAUGccuGUAGUGUucGuGCuagggGAAAaAAUAgccuaGGcAcgccuaUuagggc
(.....))))).(((.(((((((.)))))))).))))....(((.
(.....))))).(((.(((((((.)))))))).))))....(((.
CUGAGAUGCCCGUAGUUCGUGCCUAGCGAAGUCAUAAGCUAGGGCAGUCUUUAGAGGC

gUgACGGCgggaaAAaaGgCUAAggCuUugGccAuGcCuaAgUAucccUGAAAGugCCAc
))..)((.((((....(((....(((....)))..))))).))))....(((.
))..)((.((((....(((....(((....)))..))))).))))....(((.
UGACGGCAGGAAAAAGCCUACGUCUUCGGAUAUGGCUGAGUAUCCUUGAAAGUGCCACA

AGuGACGaAgccuuuugggGAAAcccaaagggUGGAACGcGGuAAaCCCaCgaGCgaGcA
.....(((((((.)))))))).))))....(((.
.....(((((((.)))))))).))))....(((.
GUGACGAAGUCUCACUAGAAAUGGUGAGAGUGGAACGCGGUAACCCUCGAGCGAGAAA

ACCCAAAUuauGGUAGGGGaaacccgauaaaagGAAauGAAcgaAuuaucggggCGgacaa
.((((....))))).))))....(((.
((((....))))).))))....(((.
CCCAAUUUUGGUAGGGGAACCUUUAACGGAAUUAACGGAGAGAAGGACAGAAUGCU

aAguccgcAGAuAGAUgaUuacGgccaagagcaauua.....aaaagcucuuu
..))))).))))....(((.
))))).]]]]]]].(((((((((((((((.)))))))).))))))
UUCUGUAGAUAGAUGAUUGCCGCCUGAGUACGAGGUGAUGAGCCGUUUGCAGUACGAUGG

ggaACAAAACAUGGCuUAcAga
).))))).
.....))))).
AACAAAACAUGGCuuACAGAACG

```


APPENDIX E

PERL SCRIPT TO CALCULATE THE MISALIGNMENTS

The misalignment produced by Infernal was calculated using the Perl script. The input file contains the consensus structure given by Infernal and the true structure as seen in PDB in the dot-parentheses format.

Input file :-

```
(((((.....((((((((.....)))))).....))))))
)).....((((.....))).....(((.....(((.....)))..)))..
)).....((((.....(((.....)))..))).....
....

((((.....((((((((.....{{{...}}}))))).....))))
).....[[[[[.(((.....))).....(((((((([[[]]]))(((.....)))..)))
)))]}.}}}]...(((.....<<<<<<..)))..)).....>>>>>>.....
....
```

Code:-

```
open(IN,"inf_align.txt");
@data=<IN>;
chomp(@data);
close IN;

$seq1 = $data[0];
$seq2 = $data[2];

print "$seq1\n";
print "$seq2\n";

$count=0;

for($i=0; $i<length($seq1); $i++)
{
    $x = substr($seq1, $i, 1);
    $y = substr($seq2, $i, 1);

    if(((($x eq '.')&&($y ne '.')) || (($x ne '.')&&($y eq
    '.')))
    {
        $count++;
    }
}
print "$count\n";
```

REFERENCES

- [1] Eddy SR, Durbin R. RNA sequence analysis using covariance models. *Nucleic Acids Res.* 1994; 22: 2079-2088.
- [2] Wang Y, Huang Z, Wu Y, Malmberg RL, Cai L. RNATOPS-W: a web server for RNA structure searches of genomes. *Bioinformatics.* 2009; 25(8): 1080–1081.
- [3] Rivas E, Eddy SR. A dynamic programming algorithm for RNA structure prediction including pseudoknots. *J mol Biol.* 1999; 285: 2053-2068.
- [4] Huang Z, Wu Y, Robertson J, Feng L, Malmberg RL, Cai L. Fast and accurate search for non-coding RNA pseudoknot structures in genomes. *Bioinformatics.* 2008; 24(20): 2281–2287.
- [5] Durbin R, Eddy S, Krogh A, Mitchison G. *Biological sequence analysis, Probabilistic models of proteins and nucleic acids.* Cambridge University Press; 1998; 233-295.
- [6] Nawrocki EP, Kolbe D, Eddy SR. *Infernal User's Guide Sequence analysis using profiles of RNA secondary structure consensus* (<http://infernal.janelia.org/Version 1.0.2>).
- [7] Achawanantakun R, Sun Y. Shape and secondary structure prediction for ncRNAs including pseudoknots based on linear SVM. *BMC Bioinformatics.* 2013;14 Suppl 2:S1. doi: 10.1186/1471-2105-14-S2-S1.
- [8] Han B, Dost B, Bafna V, Zhang S. Structural alignment of pseudoknotted RNA. *J Comput Biol.* 2008; 15(5): 489-504.
- [9] Nawrocki EP, Eddy SR. Query-dependent banding (QDB) for faster RNA similarity searches. *PLoS Comput Biol.* 2007; 3(3): e56.