An Algorithmic Framework for the Study of Behavior of siRNA Sequences

Shailendra Singh
Asst. Prof. CSE Dept., PEC University of Technology, Chandigarh,

Amardeep Singh
Associate Prof., Computer Engg. Dept., UCoE, Punjabi University, Patiala,

ABSTRACT
The study about biological sequences is gaining momentum nowadays. An increasing number of researchers have proposed framework for the implementation of various algorithms for biomolecules sequence alignment and secondary structure prediction. A comparative study can also enhance the results but alignment and prediction algorithms vary widely in terms of both sensitivity and selectivity across different lengths and homologies. The independent benchmarking of these algorithms is rarely performed for siRNA sequences. Sequence alignment and determination of the secondary structure of biomolecules are very important for understanding their functions and behavior. This paper presents the implementation of Nussinov algorithm using Bio-Perl for the study of behavior of siRNA sequences.

Keywords: Sequences, siRNA, Nussinov Algorithm, Bio-Perl

1. INTRODUCTION
A biomolecule is a molecule that naturally occurs in living organisms. All known forms of life are composed solely of bio-molecules. Biomolecules play a key role in the synthesis of protein from deoxyribonucleic acid (DNA). It is also known for its structural and catalytic roles in the cell. For the purpose of sequence alignment and structure determination, it can simply be described as a flexible single-stranded biopolymer. The biopolymer is made from a sequence of four different nucleotides (in the case of RNA and its classes): adenine (A), cytosine (C), guanine (G), and uracil (U). Intermolecular base pairs can form between different nucleotides, folding the sequence onto itself. The most stable and common of these base pairs are GC, AU, and GU, and their mirrors, CG, UA, and UG. Biomolecules play many important regulatory, catalytic and structural roles in the cell. Reliable alignment and determination of bimolecular structure as mentioned by Crick (1966) from their primary sequences is highly desirable.

The challenge in the alignment and determination of structure is the proper implemented algorithmic framework. Use of algorithm for the prediction has been increased because determining the secondary structure of biomolecules through laboratory methods such as nuclear magnetic and x-ray crystallography is very costly. And because of it the biomolecules secondary structure determination remains one of the most compelling, yet elusive areas of computational biology and bioinformatics. The secondary structure is obtained from the primary structure. The following section describes the siRNA biomolecule taken for the study.

1.1 siRNA
Small interfering RNA (siRNA), sometimes known as short interfering RNA or silencing RNA, are a class of 20-25 nucleotide-long double-stranded RNA molecules that play a variety of roles in biology. siRNA is involved in the RNA interference (RNAi) pathway where the siRNA interferes with the expression of a specific gene. In addition to their role in the RNAi pathway, siRNAs also act in RNAi-related pathways, e.g. as an antiviral mechanism or in shaping the chromatin structure of a genome. The complexity of these pathways is only now being elucidated. siRNA has undergone a major change in the last decade. There is still much speculation in scientific community as to what extent siRNAs are responsible for the complexity in higher organisms which can hardly be explained with only proteins as regulators (Eddy, 2004). The following section describes the sequences and structure of biomolecules.

1.2 Sequences and Structures
Biological sequence may be looked as words over alphabets of nucleotides. Naturally occurring siRNAs form subsets of the set of all possible words. The sequence of a biological molecule is the specification of its atomic composition represented by the alphabets e.g. A, C, T, G, U, etc. Bio-molecular structure is the structure of biomolecules. The structure of these molecules is frequently decomposed into primary structure (that is also known as sequence), secondary structure, tertiary structure, and quaternary structure. The scaffold for this structure is provided by secondary structural elements which are hydrogen bonds within the molecule. It also leads to several recognizable "domains" of protein (Eddy, 2004) structure (like alpha helices, beta sheets for proteins) and nucleic acid structure (like hairpin loops, bulges and internal loops).

2. ALGORITHMS
Lots of algorithms have been designed for determining the biomolecules secondary structure. One such algorithm is CONTRAfold and it is able to determine biomolecules secondary structure from the sequence. CONTRAfold also considers the number of non-canonical base pairings (G-U pairings), helices, bulge loops and CG/GC base-pair stacking interactions in its assessment of a Biomolecules structure. The dynamic programming is an important method of algorithm that allows us to treat sequences containing up to 2000 nucleotides, however such method neither consider pseudo-knots nor find sub-optimal solutions. A more recent method partially solves the last problem. The phylogenetic methods use covariation analysis to identify conserved paired bases among a set of homologous sequences. This is a satisfying procedure that gives excellent results, including

pseudo-knots identification. The following section discusses the modified Nussinov algorithm for sequence alignment and structure determination for siRNA sequences.

2.1 Nussinov Algorithm

The Nussinov method (Nussinov, 1978, 1980) is a simple dynamic programming algorithm to align the biological sequences and secondary structure determination. It searches for a secondary structure scheme that maximizes the number of base pairs and considers other factors, such as base stacking and strength of the base pairs (Gorodkin, 1997). This search is carried out on the recursive principle of looking for a possible first solution, and then building on it and refining it in steps. It also attempts to optimize the number of A-U and C-G base pairings within an RNA structure. We first consider this algorithm as it illustrates the general structure of more sophisticated folding algorithms. Shown below are the recursion equations used to compute the optimal scoring structure:

$$\gamma(i.j) = \max \begin{cases} \gamma(i+1,j) \\ \gamma(i,j-1) \\ \gamma(i+1,j-1)+\delta(I,j) \\ \max_{i<k<j} [\gamma(i,k) + \gamma(k+1,j)] \end{cases}$$

Where $\gamma(i, j)$ represents the score for the optimal structure from base i to base j in the sequence, and $\delta(i, j)$ represents the score of a pairing between base i and base j. The first observation one makes is that we are taking a maximum over the 4 different terms on the right. Each of the terms on the right represents a choice of non-pairing, or bifurcation. The first case arises if we choose to leave base i unpaired, and then compute the optimal structure for the sequence from base i+1 to base j, while the second case leaves base j unpaired and computes the optimal structure for the sequence from base i to base j-1. The third case is when we pair base i and base j, and we add the score from the pairing to the score of the optimal structure from base i+1 to base j-1. The fourth case occurs if we choose to begin a bifurcation at a base k which is between bases i and j. In this case, the recursion determines the optimal scoring structure over all possible locations k of the bifurcation, by recursion on the substructures on either side of the bifurcation point k.

3. METHODOLOGY

Finding the best secondary structure for a sequence is just like finding the best sequence alignment in terms of complexity. This means that for an exhaustive solution of the problem, there is no algorithm that will complete it in an amount of time that can be expressed as a polynomial function of the length of the sequence (Nussinov, 1980). To appreciate this better consider a sequence of length N. To find the best secondary structure, defined simply as one that contains the maximum number of base pairs, as we need to consider pairing the first base with every other base in the sequence. This means N calculations. The siRNA sequence chosen for the example is AAAGCCCUU. This is a simple sequence with length L=9 and can have many secondary structures. The algorithm has three main steps: constructing and initializing the matrix; filling up the elements of the matrix using a recursion algorithm; and finally constructing the trace back.

The first step is simple. The sequence is written both horizontally across the top and vertically at the left. This define a matrix called G (i, j), each element of which is labelled by one residue i from the top and one residue from j from the left (Fig. 1.1). The initialization of the matrix is carried out as follows: Set G(i,i-1) = 0 for i = 2 to L, and set G(i,i) = 0 for i = 1 to L.

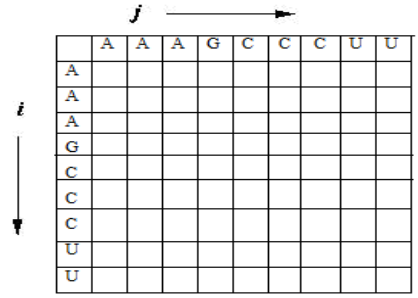


Fig. 1.1 Matrix G (i, j)

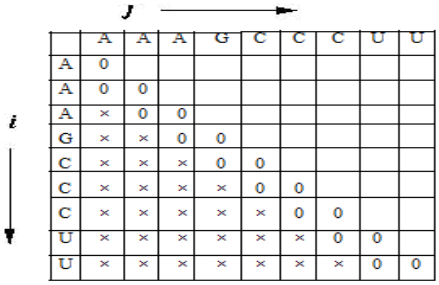


Fig. 1.2 Matrix G (i, j) after initialization

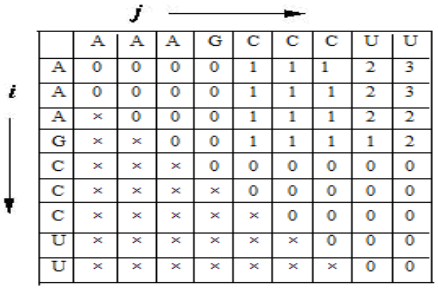


Fig. 1.3 The trace-back step

This sets the main diagonal and the diagonal just below both to zero (Pipas 1975), thereby ensuring that base pairs do not occur between a residue and the immediate next one. In addition, the entire lower left half of the matrix is blanked out; since we don't use this. The result of this operation is shown in Fig. 1.2. The next step is to recursively fill up the matrix. We start from (i = 1, j = 2) and apply the following set of operations to each cell, incrementing both i and j by 1 each time. When we come to (i = L-1, j = L), we start again by setting (i = 2, j=3), and so on until the matrix is filled. The final part of this operation is to compare the four members calculated for G (i, j) and choose the largest of them. This is used to update the matrix cell. Once the matrix has been filled by recursively applying the algorithm in step two, the score of the best possible secondary structure is easily obtained by inspection, as the highest value in the matrix. We start at the cell at the top right hand corner, which have

largest possible score. The indices of this cell are (1, L) where L is the length of the sequence (Nussinov, 1980). In this way the best secondary structure is determined as shown in Fig. 1.3.

3.1 Algorithmic Framework with Bio-Perl

Bio-Perl is a popular programming language extensively used in areas such as bioinformatics and web programming. It has become popular with biologists because it is so well-suited to several bioinformatics tasks and also an application. A program can be as simple as the following Bio-Perl language program to print some siRNA sequence data onto the computer screen.

PROGRAM

```
#!/usr/bin/Bioperl -w
# Printing a sequence:
print
"GCAUCAAUCGAUCGACGACUGUCAUAUAAUCUGA
UCGAAUG\n";
_
#!/usr/bin/Bioperl -w
$rna =
"GCAUCAAUCGAUCGACGACUGUCAUAUAAUCUGA
UCGAAUG";
# Printing a sequence:
print "$rna\n";
```

OUTPUT:

Print 'ACCUGGUAACCCGGAGAUUCCAGCU';

For implementation, a Bio-Perl program is written that does the following:

- It takes (as input) exactly one command-line argument, which is a siRNA sequence (in upper- or lower-case, or a mixture of both).
- It validates the sequence to ensure it contains no characters that are not A, C, G or U (if any other characters are present, it should print an error and die.
- It prints (on standard output) the maximum possible number of Watson-Crick base pairs in the "optimal structure" for the sequence (where "optimal structure" means the secondary structure that contains the most Watson-Crick base pairs).

A common implementation strategy is to begin by writing what is called pseudo-code. Pseudo code is an informal program, in which there are no details, and formal syntax is not followed. It does not actually run as a program; its purpose is to flesh out an idea of the overall design of a program in a quick and informal way. For example, in an actual Bio-Perl program (Tisdall, 2001) one may write a bit of code called a subroutine, in this case, a subroutine that gets an answer from a user typing at the keyboard. Such a subroutine may look like as below:

```
sub getanswer {
Print "Type in your answer here:"
My $answer = <STDIN>;
chomp $answer;
return $answer;
}
```

But in pseudo code, one must just say:
getanswer

Here's an example of pseudo code for the program:

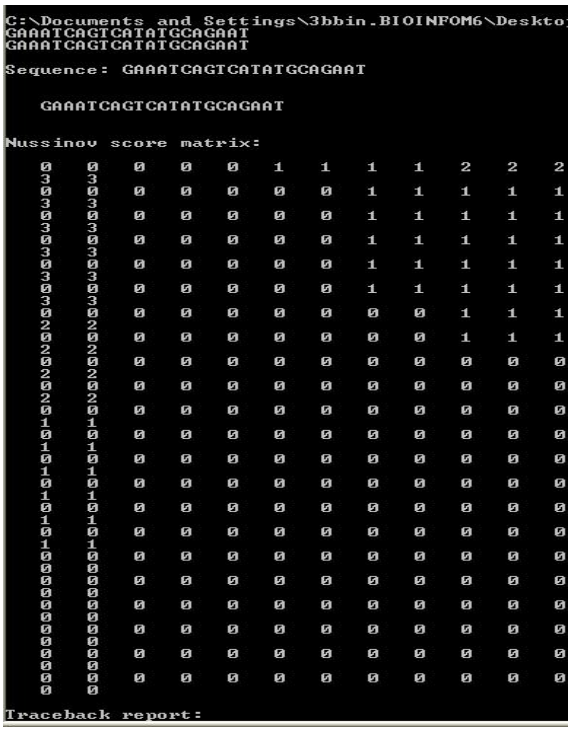
```
get the name of siRNAfile from the user
read in the siRNA from the siRNAfile
for each regulatory element
if element is in siRNA, then
add one to the count
print count
```

4. RESULTS AND DISCUSSION

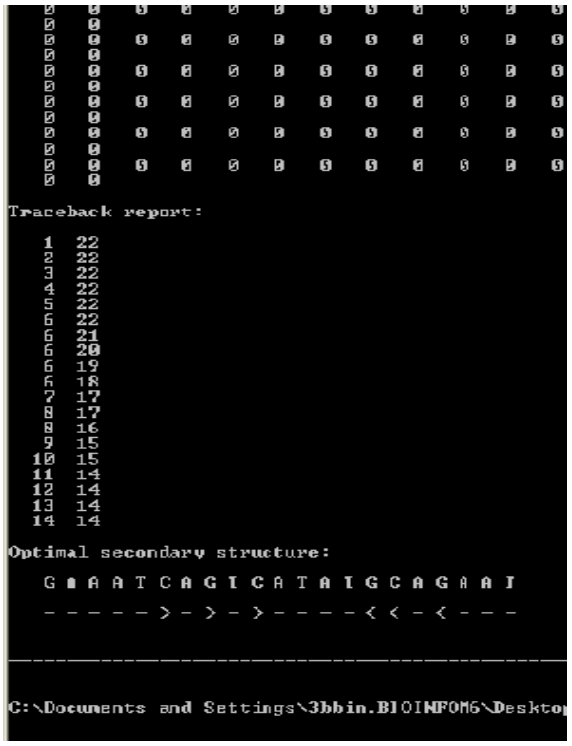
The following input table (Table 1) contains various siRNA sequences with various attributes and the Fig. 1.4 shows the output generated by the Bio-Perl program for the behavior of siRNA sequences as its substructures.

Table: 1 (siRNA sequences)

siRNA SEQUENCES
CAUUCUGCUACUUCACUGUCA
GACAAGAAGAUGGUGGAGAAG
AAAAGACAAUAGUCCCUUGGA
GAUAAGGAUGUAAAGAUUGAG
UAUUCUGUUGGAGCAGAAUCC
GAUAUCACAUCAGUGGUUCCA
GAUGCUAUUCAGGAUGCAGUU
CAAAACUAUCGUACCAUGGAA
UAAUCCUGAUUUAACUGGAUU
CAGAACUCACCAGUCACAUCA
UACAUCUUAAUGGUCAGAAUU
GAUCAUCUAUGACCGGAAAUU
CAUCUAUGACCGGAAAUUCCU
GAAGAGUCACAGUUUGAGAUG
GACAAGAACGAACCCUUCCUU
GAUCUACGAGUGGAUGGUCAA
AACAACAGUAAAUUUGCUAAG
CAAAGAUGGCCUCUACUUUAC



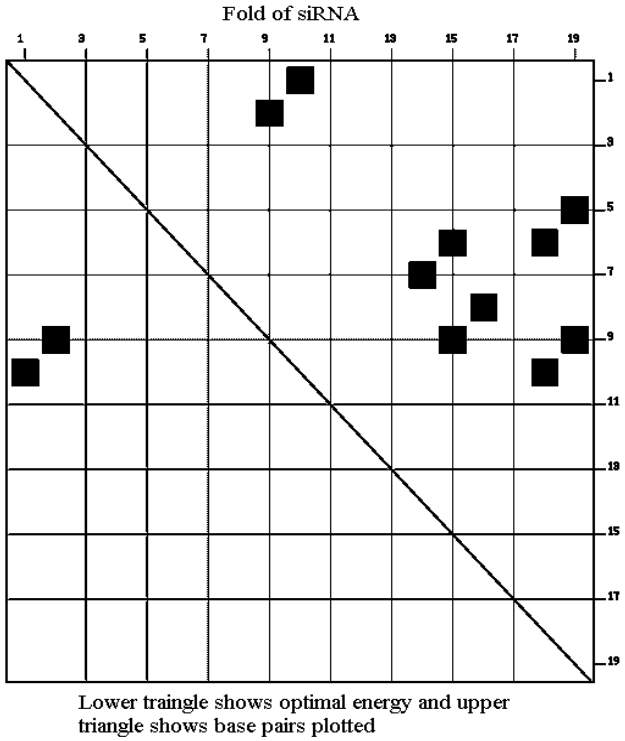
(a)



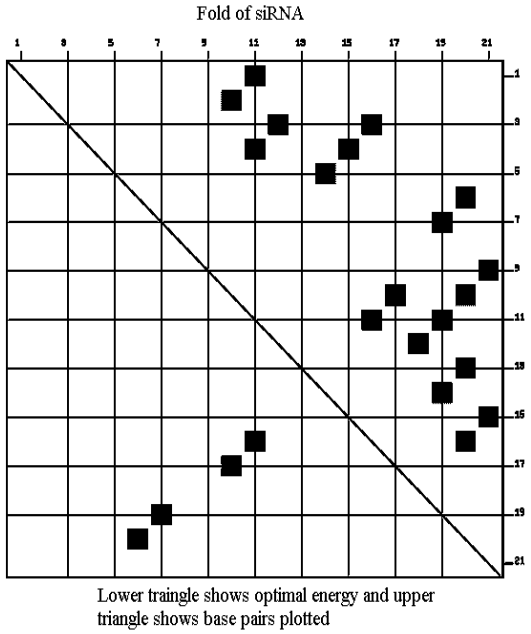
(b)

Fig. 1.4 (a), (b) Output of Bio-Perl program

Nussinov algorithm is one kind of dynamic programming algorithm and its scoring approach works well. It finds the secondary structure with the maximal number of base pairs. siRNA sequences have been used for the input of the Nussinov algorithm which gives various optimal substructures in the form of output. The behaviour of various siRNA sequences has also been verified by the other tool available. In the above energy dot plot the upper triangular region, a dot in row i and column j represents a base pair between the i^{th} and j^{th} bases. The black dots represent all the optimal folding. The colored, smaller dots represent the superposition of all possible suboptimal folding within p% of the minimum free energy, where p is the maximum percent deviation from the minimum free energy. The color ranges are red, blue and yellow; representing base pairs that are in folding within p/3%, 2p/3% and p% of the minimum free energy, respectively. Thus the yellow dots represent base pairs that are least likely to form. The upper half of the triangle in the Fig. 1.6 (a) shows the base pair plotted. And lower half shows the optimal energy. So it plotted the 10 base pairs. The upper half of the triangle in Fig. 1.6 (b) plotted 19 base pairs. In the above minimum energy plots first siRNA Sequence having optimal energy 6.0 while other one is having 1.0 So, first siRNA Sequence is better than second sequence and it will provide better optimal solution.



(a)



(b)

Fig. 1.6(a), (b) Energy dot plot for siRNA sequences

5. CONCLUSIONS

In the field of bioinformatics, the prediction of biomolecules secondary structure is certainly going to have a very bright future. The modifications in the Nussinov Algorithm have just stepped in to the field of bioinformatics and have long way to go in the field. In the future work there are lots of things that can be applied on the Nussinov Algorithm. The framework may further be explored to deal with pseudoknots. As this framework cannot deals with pseudoknots because pseudoknots violate the recursive definition of the optimal score.

6. REFERENCES

[1] Crick, F. H. 1966. Codon – Anticodon Pairing: The Wobble Hypothesis. J. Mol. Biol. 19, 548–55.

[2] Eddy, S. R. 2004. How do RNA Folding Algorithms Works? Nature Bio-Technology, Vol. No. 22(11).

[3] Eddy, S. R. 2004. What is Dynamic Programming? Nature Biotechnology, Vol. 22, No. 7.

[4] Eddy, S. R. 1994. Durbin R.: RNA Sequence Analysis Using Covariance Models. Nucleic Acids Res, Vol. 22, 2079-2088.

[5] Gorodkin J, Heyer L. J., and, Stormo G. D. 1997. Finding the Most Significant Common Sequence and Structure Motifs in a Set of RNA Sequences. Nucl. Acids Res. 25, 3724-3732.

[6] Nussinov, R. et al. 1978. Algorithms for Loop Matching. SIAM Journal of Applied Mathematics, 35, 68–82.

[7] Nussinov, R. and Jacobson, A. 1980. Fast Algorithm for Predicting the Secondary Structure of Single-Stranded RNA. Proc Natl Acad. Sci. USA, 77, 6903-6910.

[8] Pipas, J., McMahon, J. 1975. Method for Predicting RNA Secondary Structure. Proc Natl. Acad Sci USA, 72.

[9] Tisdall James. 2001. Beginning Perl for Bioinformatics. Publisher: O'Reilly.

[10] Rivas E., Eddy S. R. 1999. A Dynamic Programming Algorithm for Biomolecules Structure Prediction Including Pseudoknots. Journal of Biomolecules of Molecular Biology, 285, 2053-2063.

[11] Hengwu Li, Daming Zhu. 2005. Algorithm for Predicting RNA Secondary Structure Including Pseudoknots. Journal of Communication and Computer, USA. 1548-7709.

[12] Goldgof Greg. 2004. RNA Secondary Structure Prediction and Runtime Optimization. Proceedings of Genome Informatics, Vol. 15, 473-478.

[13] Erlangung Zur. 2006. Prediction of Structural Non-Coding RNAs by Comparative Sequence Analysis. CLEI Electronic Journal, Vol. 8 No. 2.

[14] Sharma Gaurav, Ozgun Harmanci A. 2007. Probabilistic Methods for Improving Efficiency of RNA Secondary Structure Prediction across Multiple Sequences. IEEE, 34-38.

[15] Zhou Hong, Zeng Xiao. 2005. A Three-Phase Algorithm for Computer Aided siRNA Design. Informatica, 357–364.

[16] Wang Zhuozhi, Zhang Kaizhong. 2004. Multiple RNA Structure Alignment. Proceedings of the IEEE Computational Systems Bioinformatics Conference (CSB 2004).

[17] Dowell D. 2006. Efficient Pair-wise RNA Structure Prediction and Alignment Using Sequence Alignment Constraints. Journal of BMC Bioinformatics.