1

1 RNA

RNA

- In this section, we are going to try to examine the sequence of RNAs, and predict its structure.
- In RNA, thymine (T) is substituted with uracil (U).
- There are many types of RNA including mRNA, tRNA and rRNA.

2

RNA

- mRNA is transcribed from DNA, then translated to proteins.
- They are called coding RNA. They *do* code for proteins.
- RNA can also have catalytic and structural functions, like proteins.
- These are called RNA genes, or non-coding RNA. The most important examples are tRNA and rRNA.
- RNA can have regulatory properties; e.g. microRNA.

3

RNA

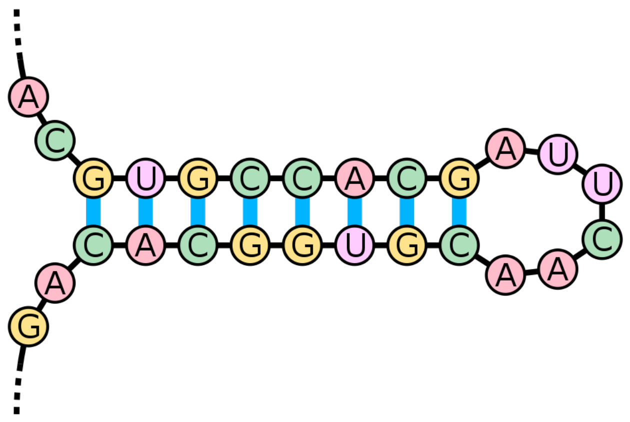
- RNA can also come in double-stranded form.
- The RNA world hypothesis proposes that the common ancestor to all life used RNA to carry genetic information and to catalyze biochemical reactions.

4

2 RNA Secondary Structure

RNA Stucture

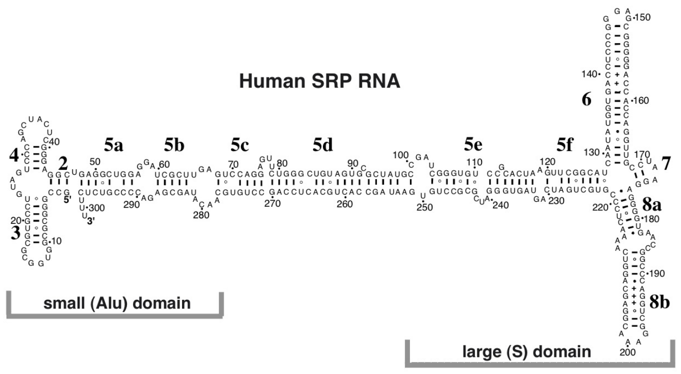
Some primary features of RNA secondary structure include single stranded regions, stems, bulges and loops.



picture from http://en.wikipedia.org/wiki/File:Stem-loop.svg

5

RNA Structure



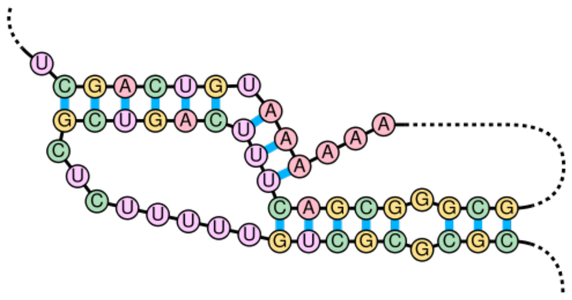
Andersen, E.S., Rosenblad, M.A., Larsen,, N., Westergaard, J.C., Burks, J., Wower, I.K., Wower, J., Gorodkin, J., Samuelsson, T., and Zwieb, C. (2006) The tmRDB and SRPDB resources. Nucleic Acids Res. 34, D163-D168.

picture from http://de.wikipedia.org/wiki/Datei:RF00017_hsap.jpg

6

Pseudoknots

- There is also a secondary structure called the pseudoknot.
- This results from a region of a loop pairing with a complementary region outside the loop.



picture from http://en.wikipedia.org/wiki/File:Pseudoknot.svg

7

3 Secondary Structure Prediction

RNA secondary Structure Prediction

- Often times, we ignore this structure with secondary prediction, because it is very hard.
- However, without pseudoknots, we've got an $O(n^3)$ algorithm, while allowing pseudoknots makes the problem exponential.

8

- After we've found a possible conformation of the RNA structure, we would like to evaluate it, to see if it is good.
- We can assign a score, much like we would with an alignment.
- As a first try, it seems the more basepairs we obtain the better, so why don't we let the number of basepairs be our score?

9

4 Nussinov's Algorithm

RNA Secondary Structure Prediction

- Now we must *find* the structure with the highest score.
- One method to do this is called *Nussinov's Algorithm* which uses dynammic programming.
- A little bit like a sequence alignment, although we are aligning it with itself.

10

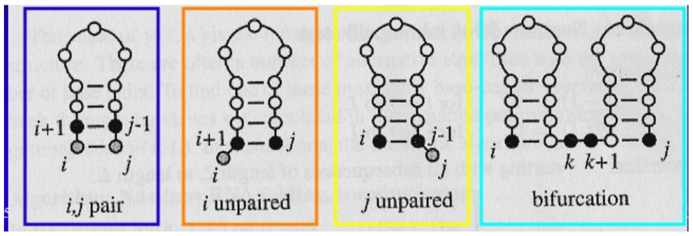
Nussinov's Algorithm

- Let $w = a_1 a_2 \cdots a_n$ be our input sequence, where $a_i \in \{A, U, G, C\}$ for i, $1 \leq i \leq n$.
- We move from the "inside" of the sequence towards the "outside".
- Suppose we have already figured out *every* optimal "substructure" strictly between a_i and a_j.
- So, for example, we already have figured out the optimal structure of the sequence $a_{i+1} \cdots a_j$, and $a_{i+2} \cdots a_j$, and also $a_i \cdots a_{j-1}$ amongst many others.

11

- There are four ways to extend that information to a best structure of $a_i \cdots a_j$.
  1. add an unpaired base i to the best structure for the subsequence $a_{i+1} \cdots a_j$,
  2. add an unpaired base j to the best structure for the subsequence $a_i \cdots a_{j-1}$,
  3. add paired bases i, j to the best structure for the subsequence $a_{i+1} \cdots a_{j-1}$,
  4. combine two optimal substructures $a_i \cdots a_k$ and $a_{k+1} \cdots a_j$, for some k.

12



- Cases 1 and 2 (orange and yellow outline) should not result in an increase in score.
- Case 3 (blue outline) should result in an increase of score.
- Case 4 (teal outline) does not increase the score, however, it requires we try every possible value of k to see which one gives you the structure with the best score.

picture from http://compbio.pbworks.com/w/page/16252918/RNA%20Folding

13

- A recursive procedure would not be a very good way to program the problem.
- Indeed, by starting with the full string and then calculating its optimal structure by calculating smaller ones recursively inwards, we are calculating the optimal structure of many of the sequences many times.

14

- We only want to calculate each one once.
- This is exactly what dynamic programming can do.
- Start with small instances, calculate the optimal structure once and then use those to build larger instances.

15

We start by building an $n \times n$ matrix, and plot w against itself.

Example 1. For this example, let $w = UGAU$.

$$\begin{array}{c|c|c|c|c|} & U & G & A & U \\ \hline U & & & & \\ \hline G & & & & \\ \hline A & & & & \\ \hline U & & & & \\ \hline \end{array}$$

16

Initialization

- We need to initialize it somehow, much as we did with sequence alignment.
- Instead of going from the beginning of the sequences towards the end, as with sequence alignment, we are trying to go from "small subsequences" outwards towards "bigger subsequences".

  note

  We know that the situation in aligning everything between a_i and a_i is boring. There are no basepairs. So, we should initialize all the entries on the diagonal to 0.

17

Initialization

- Also, notice that we are mainly concerned with the situation where $i \leq j$.
- Thus, we can ignore much of the table.
- It turns out that it is a good idea to initialize the main diagonal to be zeros, but also the diagonal just below as well.

Example 2. For our example, this gives us:

	U	G	A	U
U	0			
G	0	0		
A	x	0	0	
U	x	x	0	0

18

Initialization Pseudocode

we initialize the 0's as follows:

```
for $i \leftarrow 0$ to $n-1$
    $M(i,i) \leftarrow 0$
for $i \leftarrow 1$ to $n-1$
    $M(i,i-1) \leftarrow 0$
```

19

Recurrence

Recall the matrix:

	U	G	A	U
U	0			
G	0	0		
A	x	0	0	
U	x	x	0	0

We then fill in the rest of the matrix with the following recurrence:

$$\begin{aligned} M(i,j) = \max\{ & M(i+1,j), \\ & M(i,j-1), \\ & M(i+1,j-1) + d(i,j), \\ & \max\{M(i,k) + M(k+1,j) \mid \text{ for } i \leq k < j\}, \\ & \}, \end{aligned}$$

where $d(i,j) = \begin{cases} 1 & \text{if } w(i) \text{ and } w(j) \text{ are complementary,} \\ 0 & \text{otherwise} \end{cases}$

20

Finding the Score

Example 3. We obtain the following matrix:
$$\begin{array}{c|c|c|c|c|}
 & \text{U} & \text{G} & \text{A} & \text{U} \\ \hline
\text{U} & 0 & 0 & 1 & 1 \\ \hline
\text{G} & 0 & 0 & 0 & 1 \\ \hline
\text{A} & \text{x} & 0 & 0 & 1 \\ \hline
\text{U} & \text{x} & \text{x} & 0 & 0 \\ \hline
\end{array}$$

- Entry $M(0, n-1)$ is the optimal score.
- Of course, now we'd actually like to build one such optimal structure.

21

5 Stacks

Create the Structure

- As with sequence alignment, we trace back from the optimal score.
- Problem: the traceback is complicated by the bifurcation defined by the fourth term in the recurrence relation.

22

6 Stacks

Stacks — a Tangent

- A stack is a computational data structure, similar to an array.
- Intuitively, it is similar to a stack of books.
- We can look at the top book on the stack, we can "pop" a book from the top, or we can "push" a new book onto the top of the stack.

23

- When we push a new book onto the top, everything below is buried, but they are still there.
- We can get at them, but only if we keep on popping books until we get to the one we want.

Example 4.
$$\begin{array}{|c|}
\hline \mathbf{2} \\ \hline
6 \\ \hline
\vdots \\ \hline
5 \\ \hline
11 \\ \hline
\end{array}$$

- We can have a stack of numbers, or anything else for that matter.
- We can also have a stack of ordered pairs of integers (like our next example). So each cell will contain (i, j) for some i, j.

24

7 Finding an Optimal Secondary Structure

Create the Structure

the algorithm

```
PUSH($(0, n-1)$)   //adds the ordered pair $(1, n)$ to the stack

while READ()$\neq$ II //until we hit the bottom of the stack, do the following
    $(i, j) \leftarrow$ POP()

    if $i \geq j$          //if $i$ is greater than or equal to $j$, do nothing
    else if $M(i+1, j) == M(i, j)$ PUSH ($(i+1, j)$)
    else if $M(i, j-1) == M(i, j)$ PUSH ($(i, j-1)$)
    else if $M(i, j) == M(i+1, j-1)$+d$(i, j)$
        output "i pairs with j" //this outputs to the screen
        PUSH($(i+1, j-1)$)
    else
        for $k \leftarrow i+1$ to $j-1$
            if $M(i, k) + M(k+1, j) == M(i, j)$
                PUSH($(k+1, j)$)
                PUSH($(i, k)$)
                break //exits the for loop
```

25

8 RNA Secondary Structure Prediction Methods

RNA Secondary Structure Prediction

- Nussinov's algorithm is from 1978.
- Most algorithms now try to minimize free energy instead of counting basepairs.

26

- The most prominent program is the mfold package of Michael Zucker.

  website

  http://mfold.rna.albany.edu/?q=mfold
- mfold does not take pseudoknots into account.
- This was merged with DINAMelt to create a newer package called UNAFold.

  website

  http://unafold.rna.albany.edu/?q=DINAMelt

27

- There are also algorithms which take pseudoknots into account.
- They typically sacrifice accuracy for speed.
- Indeed, when pseudoknots are included, using minimization of free energy as criteria, the problem has been shown to be NP-complete (see chapter 1).
- Thus, exponential algorithms are likely the best we will be able to do, if we want to be completely accurate.

28