12 RNA Secondary Structure

Sources for this lecture:

- R. Durbin, S. Eddy, A. Krogh und G. Mitchison, Biological sequence analysis, Cambridge, 1998
- J. Setubal & J. Meidanis, Introduction to computational molecular biology, 1997
- D.W. Mount. Bioinformatics: Sequences and Genome analysis, 2001
- M. Zuker, Algorithms in Computational Molecular Biology. Lectures, 2002
- Sean R. Eddy: How do RNA folding algorithms work? Nature Biotechnology, Volume 22 Number 11, pages 1457-1458, 2004
- Rune Lyngsø, Lecture notes on RNA secondary structure prediction, 2005

12.1 RNA

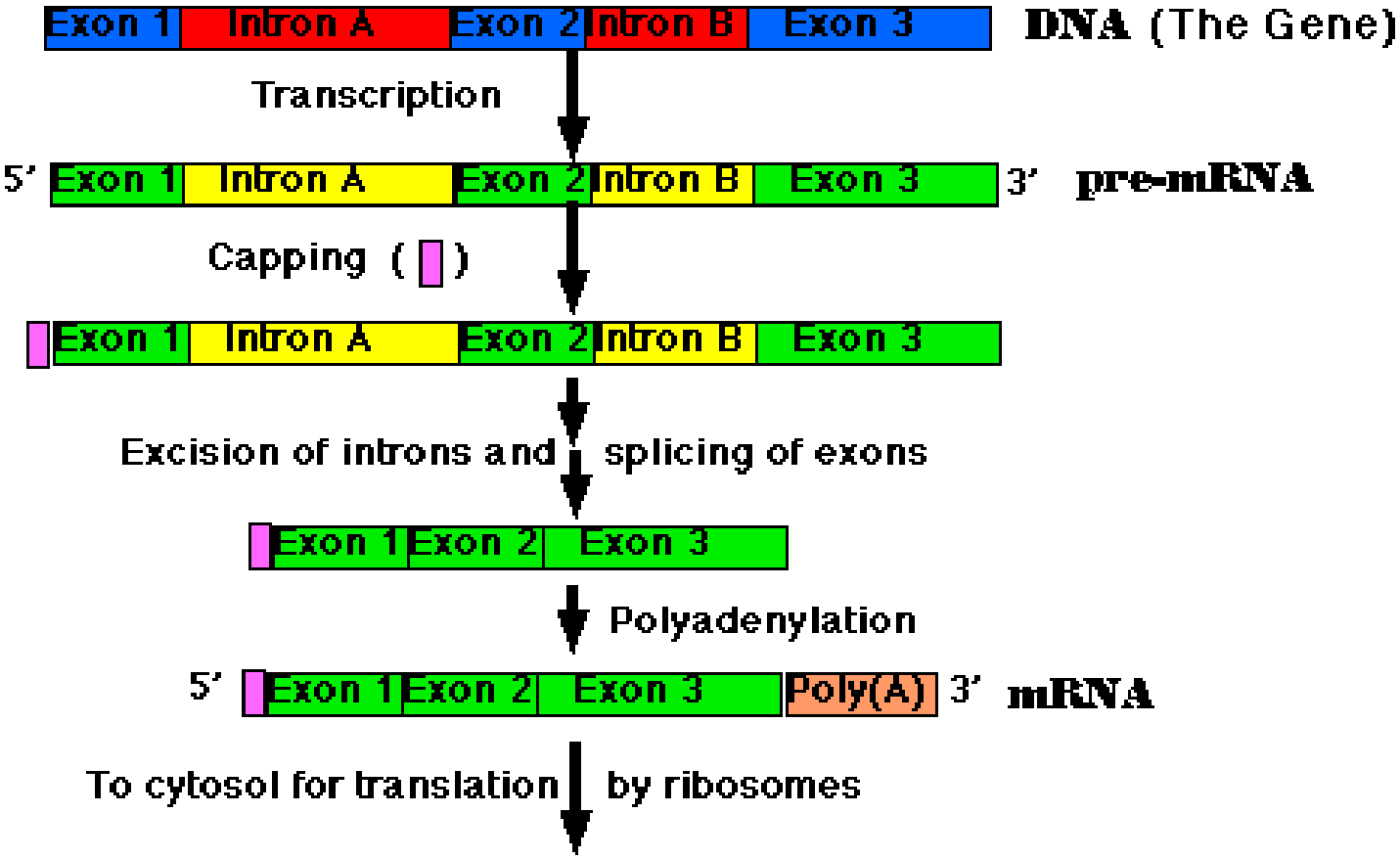
RNA, *DNA* and *proteins* are the basic molecules of life on Earth. Recall that:

- DNA is used to store and replicate genetic information,
- proteins are the basic building blocks and active players in the cell, and
- RNA plays a number of different important roles in the production of proteins from instructions encoded in the DNA.

In eukaryotes, DNA is transcribed into pre-mRNA, from which introns are spliced to produce mature mRNA, which is then translated by ribosomes to produce proteins with the help of tRNAs. A substantial amount of a ribosome consists of RNA.

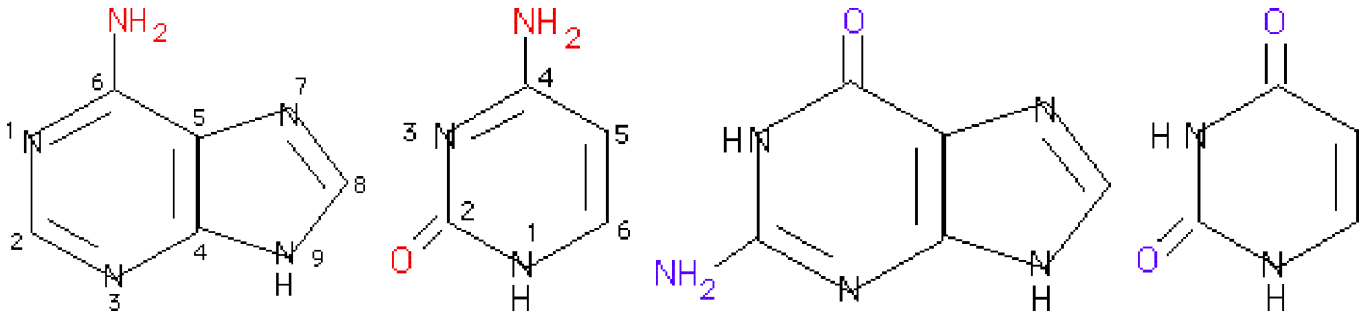
The *RNA-world* hypothesis suggests that originally, life was based on RNA and over time RNA delegated the data storage problem to DNA and the problem of providing structure and catalytic functionality to proteins.

Below you see the process of transcription.



(source: http://users.rcn.com/jkimball.ma.ultranet/BiologyPages)

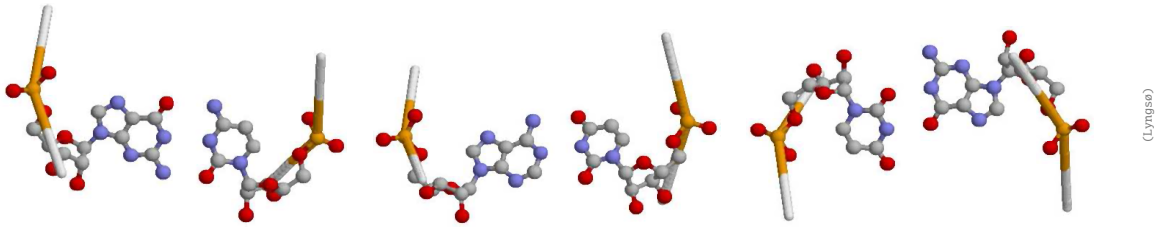
An RNA molecule is a polymer composed of four types of (ribo)nucleotides, each specified by one of four bases:



(Zuker)

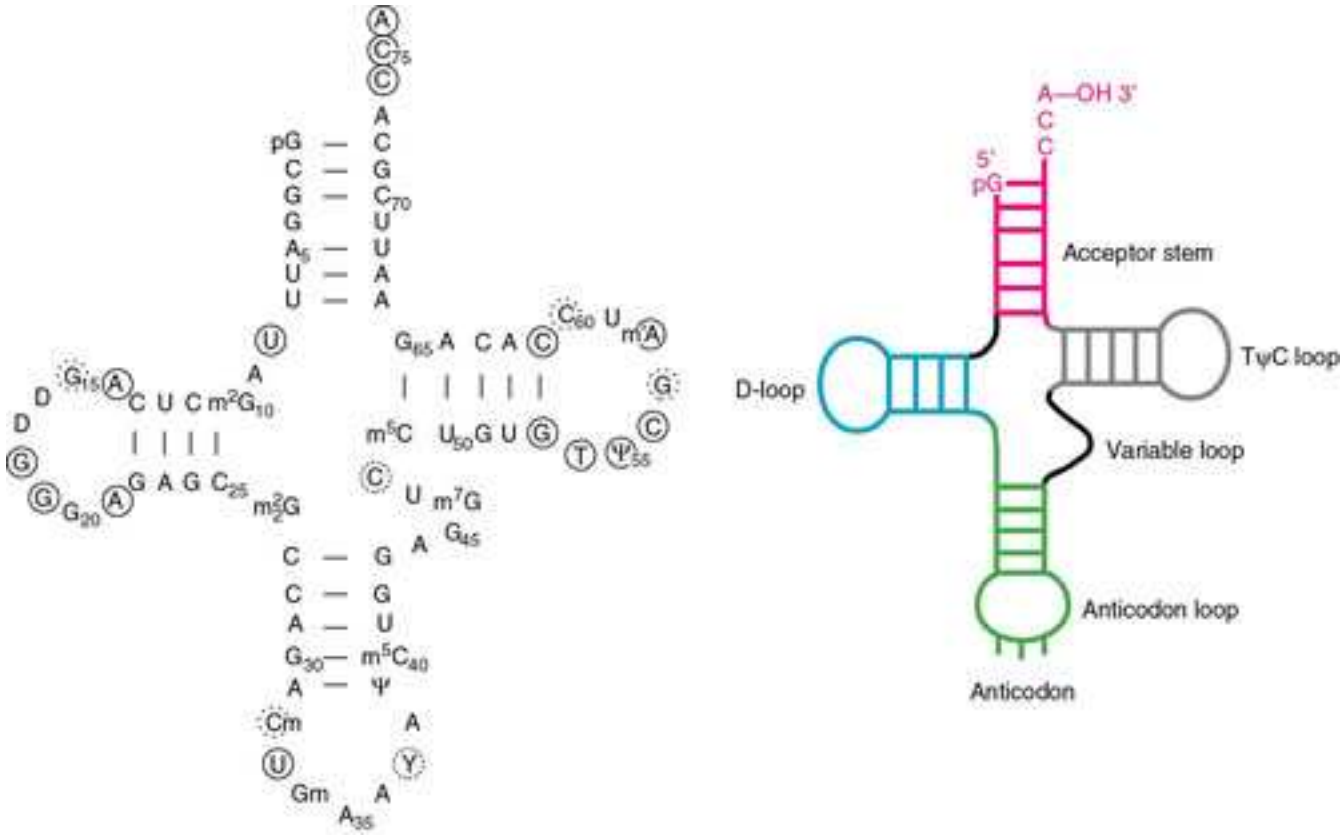
12.2 RNA secondary structure

Unlike DNA, RNA is single stranded. However, complementary bases `C` – `G` and `A` – `U` form stable *base pairs* with each other using hydrogen bonds. These are called *Watson-Crick* pairs. Also important are the weaker `U` – `G` *wobble pairs*. Together they are all called *canonical base pairs*.



Guanine-Cytosine Adenine-Uracil Uracil-Guanine

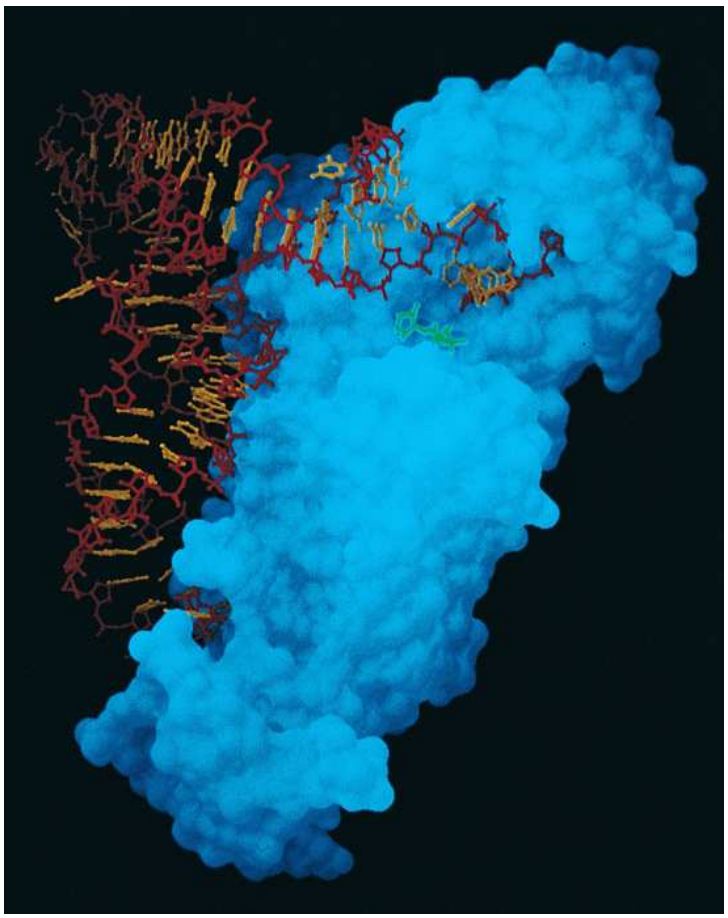
When base pairs are formed between different parts of a RNA molecule, then these pairs are said to define the *secondary structure* of the RNA molecule. Here is the secondary structure of a tRNA:



(www.blc.arizona.edu/marty/411/Modules/ribtRNA.html)

This particular tRNA is from yeast and is for the amino acid phenylalanine.

The three dimensional structure is much less packed than that of a protein:



tRNA-synthetase complex in three dimensions.
(source:http://anx12.bio.uci.edu/~hudel/bs99a/lecture21/trna_synth_3d.gif)

12.3 Definition of RNA secondary structure

The *true secondary structure* of a real RNA molecule is the set of base pairs that occur in its three-dimensional structure.

Definition For our purposes, a *RNA molecule* is simply a string

$$x = (x_1, x_2, \ldots, x_L),$$

with $x_i \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{U}\}$ for all i.

Definition. A *secondary structure* for x is a set P of ordered *base pairs*, written (i, j), with $1 \leq i < j \leq L$, satisfying:

1. $j - i > 3$, i.e. the bases are not too close to each other (although we will sometimes ignore this condition below), and
2. $\{i, j\} \cap \{i', j'\} = \emptyset$, i.e. the base pairs don't conflict.

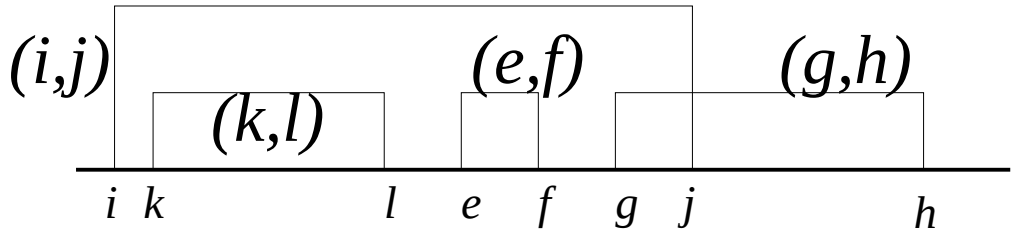
Definition A secondary structure is called *nested*, if for any two base pairs (i, j) and (i', j'), w.l.o.g. $i < i'$, we have either

1. $i < j < i' < j'$, i.e. (i, j) precedes (i', j'), or

2. $i < i' < j' < j$, i.e. (i, j) includes (i', j').

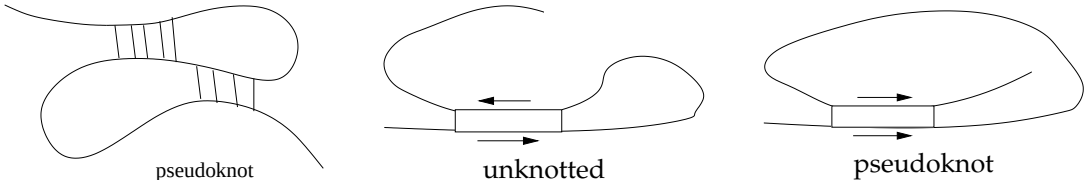
12.4 Nested structures

In the following, we only will consider *nested* secondary structures, as the more complicated non-nested structures are not tractable with the methods we will discuss.

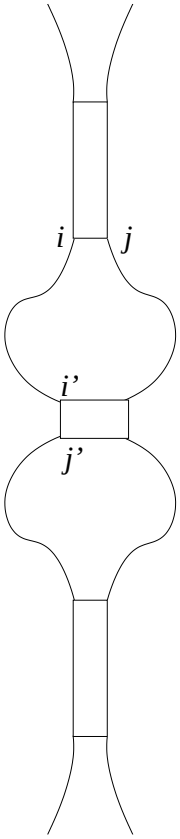


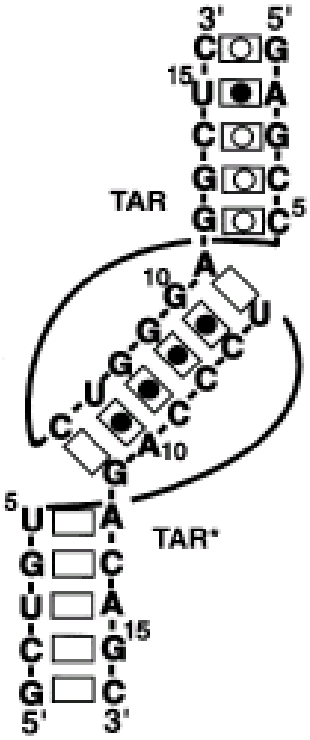
Here, the interactions (i, j) and (g, h) are not nested.

Interactions that are not nested give rise to a *pseudo knot* configuration in which segments of sequence are bonded in the "same direction", or have a three dimensional contact.



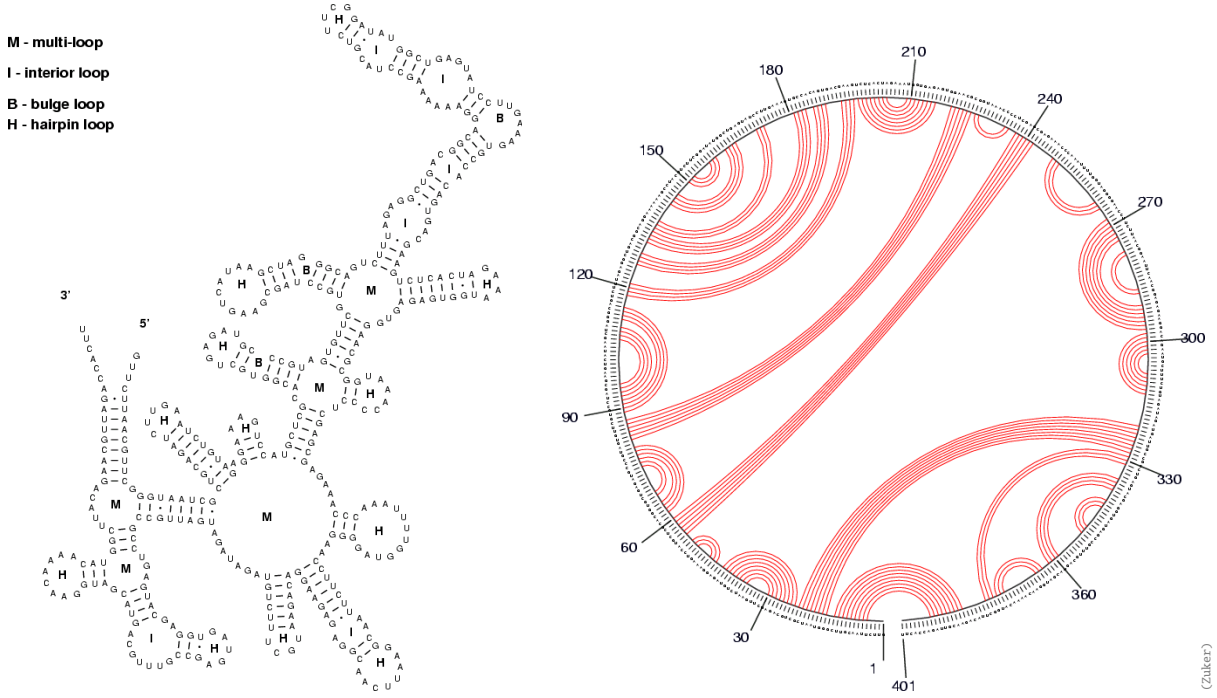
The nested requirement excludes other types of configurations, as well, such as *kissing hairpins*, for example:





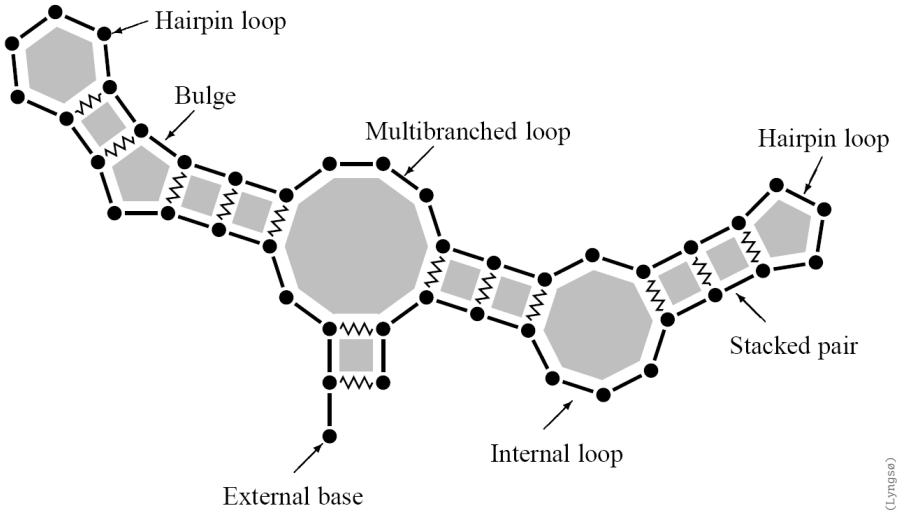
12.5 Example of secondary structure

Predicted structure for Bacillus Subtilis RNAase P RNA:



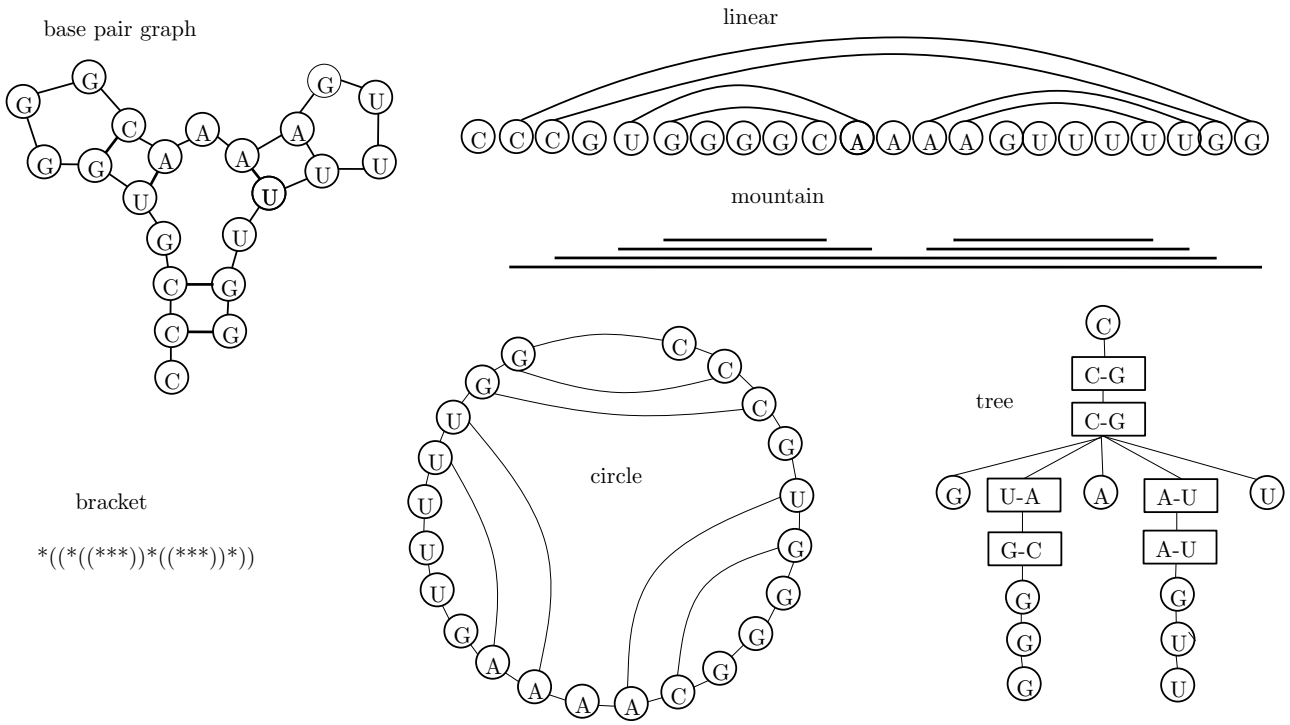
This example shows two different ways to depict RNA secondary structure and as well as most of the different types of single- and double-stranded regions in it:

- single-stranded RNA,
- double-stranded RNA helix of stacked base pairs,
- stem and loop or hairpin loop,
- bulge loop,
- interior loop, and
- junction or multi-loop.



There are various other representations. The most common ones are:

1. the ”base pair” graph representation
2. the linear representation
3. the mountain representation
4. the bracket representation
5. the circle representation
6. the tree representation



12.6 Prediction of RNA secondary structure

The problem of predicting the secondary structure of RNA has some similarities to DNA alignment, except that the sequence folds back on itself and aligns complementary bases rather than similar ones.

The goal of aligning two or more biological sequences is to determine whether they are homologous or just similar. In contrast, a secondary structure for an RNA is a simplification of the complex three-dimensional folding of the RNA molecule.

Problem. Determine the true secondary structure of an RNA.

Variants: Find a secondary structure that:

1. maximizes the number of base pairs, or
2. minimizes the "free energy", or
3. is optimal, given a family of related sequences.

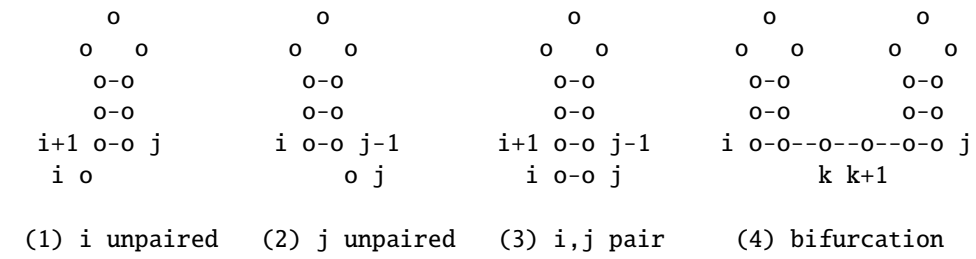
12.7 The Nussinov folding algorithm

The simplest approach to predicting the secondary structure of RNA molecules is to find the configuration with the greatest number of paired bases. Note that the number of possible configurations to be inspected grows exponentially with the length of the sequence.

Fortunately, we can employ dynamic programming to obtain an efficient solution. In 1978 Ruth Nussinov et al. published a method to do just that.

The algorithm is recursive. It calculates the best structure for small subsequences, and works its way outward to larger and larger subsequences. The key idea of the recursive calculation is that there are only four possible ways of the getting the best structure for i, j from the best structures of the smaller subsequences.

Idea: There are four ways to obtain an optimal structure for a sequence i, j from smaller substructures:



1. Add an *unpaired base i* to the best structure for the subsequence $[i + 1, j]$,
2. add an *unpaired base j* to the best structure for the subsequence $[i, j - 1]$,
3. add *paired bases i and j* to the best structure for the subsequence $[i + 1, j - 1]$, or
4. *combine two* optimal substructures $[i, k]$ and $[k + 1, j]$.

We are given a sequence $x = (x_1, \ldots, x_L)$ of length L. Let $\delta(i, j) := 1$, if $x_i - x_j$ is a canonical base pair and 0 else.

The dynamic programing algorithm has two stages:

In the *fill* stage, we will recursively calculate scores $\gamma(i, j)$ which are the maximal number of base pairs that can be formed for subsequences $(x_i, \ldots, x_j)$.

In the *traceback* stage, we traceback through the calculated matrix to obtain one of the maximally base paired structures.

12.8 The fill stage

Algorithm (Nussinov RNA folding, fill stage)
Input: Sequence $x = (x_1, x_2, \ldots, x_L)$
Output: Maximal number $\gamma(i, j)$ of base pairs for $(x_i, \ldots, x_j)$.

Initialization:

$$\gamma(i, i-1) = 0 \qquad \text{for } i = 2 \text{ to } L,$$
$$\gamma(i, i) = 0 \qquad \text{for } i = 1 \text{ to } L;$$

Recursion:
for $n = 2$ **to** L **do** // longer and longer subsequences
  for $j = n$ **to** L **do**
    $i \leftarrow j - n + 1$

$$\gamma(i, j) \leftarrow \max \begin{cases} \gamma(i+1, j), \\ \gamma(i, j-1), \\ \gamma(i+1, j-1) + \delta(i, j), \\ \max_{i<k<j} \big(\gamma(i, k) + \gamma(k+1, j)\big). \end{cases}$$

Consider the sequence $x =$ `GGGAAAUCC`. Here is the matrix γ after initialization ($i :\downarrow$, $j :\rightarrow$):

Nussinov Matrix

	G	G	G	A	A	A	U	C	C
G	0								
G	0	0							
G		0	0						
A			0	0					
A				0	0				
A					0	0			
U						0	0		
C							0	0	
C								0	0

Here is the matrix γ after executing the recursion ($i :\downarrow$, $j :\rightarrow$):

Nussinov Matrix

	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	3
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

Values obtained using $\delta(a,b) = \begin{cases} 1 & \text{if } \{a,b\} = \{\texttt{A},\texttt{U}\} \text{ or } \{\texttt{C},\texttt{G}\}, \\ 0 & \text{else.} \end{cases}$

12.9 The traceback stage

Algorithm traceback(i, j) (Nussinov RNA folding)
Input: Matrix γ and positions i, j.
Output: Secondary structure maximizing the number of base pairs.
Initial call: traceback$(1, L)$.
if $i < j$ **then**
  if $\gamma(i, j) = \gamma(i + 1, j)$ **then** // case (1)

```
    traceback(i + 1, j)
else if γ(i, j) = γ(i, j − 1) then                    // case (2)
    traceback(i, j − 1)
else if γ(i, j) = γ(i + 1, j − 1) + δ(i, j) then      // case (3)
    print base pair (i, j)
    traceback(i + 1, j − 1)
else for k = i + 1 to j − 1 do                        // case (4)
    if γ(i, j) = γ(i, k) + γ(k + 1, j) then
        traceback(i, k)
        traceback(k + 1, j)
```

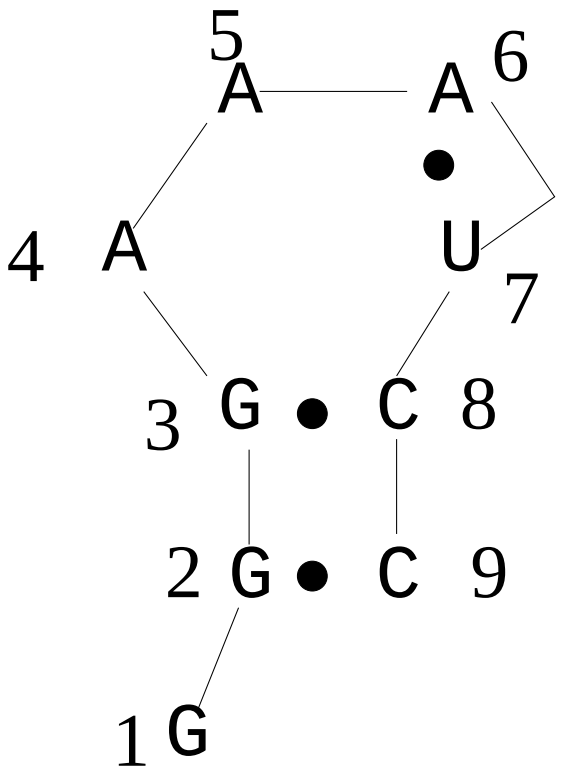
Here is the traceback through γ (i :↓, j :→):

Nussinov Matrix

	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	3
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

(There is a slight error in the traceback shown in Durbin et al. page 271)

The resulting secondary structure is:



12.10 Simple energy minimization

Maximizing the number of base pairs as described above does not lead to good structure predictions. Better predictions can be obtained by minimizing the following *energy function*

$$E(x, P) = \sum_{(i,j)\in P} e(x_i, x_j),$$

where $e(x_i, x_j)$ is the amount of *free energy* associated with the base pair (x_i, x_j).

Reasonable values for e at 37^oC are -3, -2 and -1 kcal/mole for base pairs `C` $-$ `G`, `A` $-$ `U` and `G` $-$ `U`, respectively.

Obviously, a few simple changes to the Nussinov algorithm will produce a new algorithm that can solve this energy minimization problem.

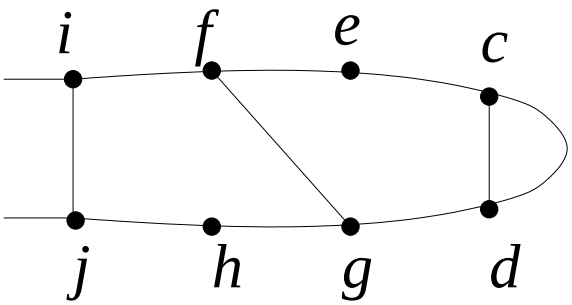
Unfortunately, this approach does not produce very good structure predictions because it does not take into account that helical stacks of base pairs have a stabilizing effect whereas loops have a destabilizing effect on the structure. A more sophisticated approach is required.

The most sophisticated algorithm for folding single RNAs is the *Zuker* algorithm, an energy minimization algorithm which assumes that the correct structure is the one with the lowest *equilibrium free energy* ΔG.

The ΔG of an RNA secondary structure is approximated as the sum of individual contributions from loops, base pairs and other secondary structure elements. As we will see, an important difference to the Nussinov calculation is that the energies are computed from loops rather than from base pairs. This provides a better fit to experimentally observed data.

12.11 The k-loop decomposition

If (i, j) is a base pair in P and $i < h < j$, then we say that *h is accessible from* (i, j) if there is no base pair $(i', j') \in P$ such that $i < i' < h < j' < j$. Similarly, we say that (f, g) *is accessible from* (i, j), if both f and g are.



The set s of all $k-1$ base pairs and k' unpaired bases that are accessible from (i, j) is called the *k-loop closed by* (i, j).

The *null* k-loop consists of all *free* base pairs and unpaired bases that are accessible from no base pair.

The following is a consequence of nestedness:

Fact. The number of non-null k-loops equals the number of base pairs.

We can now give a formal definition of the names introduced earlier:

1. A 1-loop is called a *hairpin* loop.
2. Assume that there is precisely one base pair (i', j') accessible from (i, j). Then this 2-loop is called
   (a) a *stacked pair*, if $i' - i = 1$ and $j - j' = 1$,
   (b) a *bulge loop*, if $i' - i > 1$ or $j - j' > 1$, but not both, and
   (c) an *interior loop*, if both $i' - i > 1$ and $i - j' > 1$.
3. A k-loop with $k \geq 3$ is called a *multi-loop*.

Fact. Given $x = (x_1, x_2, \ldots, x_L)$. Any secondary structure P on x partitions the set $\{1, 2, \ldots, L\}$ into k-loops $s_0, s_1, \ldots, s_m$, where $m > 0$ iff $P \neq \emptyset$.

The *size* of a k-loop is the number k' of unpaired bases that it contains. Each k-loop is assigned an energy $e(s_i)$ and the energy of a structure P is given by:

$$E(p) = \sum_{i=0}^{m} e(s_i).$$

It is important to notice that this sum will use terms based on the neighboring base pairs which allows a good approximation of the Gibbs free energy.

So now the energy is a function of k-loops instead of a function of base pairs.

12.12 Zuker's algorithm for folding RNA

We will now develop a more involved dynamic program that uses loop-dependent rules. It is due to M. Zuker (Zuker & Stiegler 1981, Zuker 1989). We will use two matrices, W and V.

For $i < j$, let $W(i, j)$ denote the minimum folding energy of all non-empty foldings of the subsequence $(x_i, \ldots, x_j)$.

Additionally, let $V(i, j)$ denote the minimum folding energy of all non-empty foldings of the subsequence $(x_i, \ldots, x_j)$, *containing the base pair* (i, j). The following fact is obvious but crucial:

$$W(i, j) \leq V(i, j) \text{ for all } i, j.$$

These matrices are initialized as follows:

$$W(i, j) = V(i, j) = \infty \text{ for all } i, j \text{ with } j - 4 < i < j.$$

(Note that we are now going to enforce that two paired bases are at least 3 positions away from each other).

12.13 Loop-dependent energies

We define different energy functions for the different types of loops:

- Let $eh(i, j)$ be the energy of the *hairpin loop* closed by the base pair (i, j),
- let $es(i, j)$ be the energy of the *stacked pair* (i, j) and $(i + 1, j - 1)$,
- let $ebi(i, j, i', j')$ be the energy of the *bulge* or *interior loop* that is closed by (i, j), with (i', j') accessible from (i, j), and
- let a denote a constant energy term associated with a *multi-loop* (a more general function for this case will be discussed later).

Predicted free-energy values (kcal/mol at 37^oC) for base pair stacking:

	A/U	C/G	G/C	U/A	G/U	U/G
A/U	-0.9	-1.8	-2.3	-1.1	-1.1	-0.8
C/G	-1.7	-2.9	-3.4	-2.3	-2.1	-1.4
G/C	-2.1	-2.0	-2.9	-1.8	-1.9	-1.2
U/A	-0.9	-1.7	-2.1	-0.9	-1.0	-0.5
G/U	-0.5	-1.2	-1.4	-0.8	-0.4	-0.2
U/G	-1.0	-1.9	-2.1	-1.1	-1.5	-0.4

Predicted free-energy values (kcal/mol at 37^oC) for features of predicted RNA secondary structures, by size of loop:

size	internal loop	bulge	hairpin
1	.	3.9	.
2	4.1	3.1	.
3	5.1	3.5	4.1
4	4.9	4.2	4.9
5	5.3	4.8	4.4
10	6.3	5.5	5.3
15	6.7	6.0	5.8
20	7.0	6.3	6.1
25	7.2	6.5	6.3
30	7.4	6.7	6.5

12.14 The main recursion

The main recursion of the Zuker algorithm is as follows: For all i, j with $1 \leq i < j \leq L$:

$$W(i, j) = \min \begin{cases} W(i + 1, j) \\ W(i, j - 1) \\ V(i, j) \\ \min_{i<k<j}\{W(i, k) + W(k + 1, j)\}, \end{cases} \tag{12.1}$$

and

$$V(i, j) = \min \begin{cases} eh(i, j) \\ es(i, j) + V(i + 1, j - 1) \\ VBI(i, j), \\ VM(i, j), \end{cases} \tag{12.2}$$

where

$$VBI(i,j) = \min_{\substack{i<i'<j'<j \\ i'-i+j-j'>2}} \{ebi(i,j,i',j') + V(i',j')\}, \tag{12.3}$$

and

$$VM(i,j) = \min_{i<k<j-1} \{W(i+1,k) + W(k+1,j-1)\} + a. \tag{12.4}$$

Equation 11.1 considers the four cases in which (a) i is unpaired, (b) j is unpaired, (c) i and j are paired to each other and (d) i and j are paired, but not to each other. In case (c) we reference the auxiliary matrix V.

Equation 11.2 considers the different situations that arise when bases i and j are paired, closing (a) a hairpin loop, (b) a stacked pair, (c) a bulge or interior loop or (d) a multi-loop. The two latter cases are more complicated and are obtained from equations 11.3 and 11.4.

Equation 11.3 takes into account all possible ways to define a bulge or interior loop that involves a base pair (i', j') and is closed by (i, j). In each situation, we have a contribution from the bulge or interior loop and a contribution from the structure that is on the opposite side of (i', j').

Equation 11.4 considers the different ways to obtain a multi-loop from two smaller structures and adds a constant contribution of a to close the loop.

12.15 Time analysis

The minimum folding energy E_{min} is given by $W(1, L)$.

There are $O(L^2)$ pairs (i, j) satisfying $1 \leq i < j \leq L$.

The computation of

1. W takes $O(L^3)$ steps,
2. V takes $O(L^2)$ steps,
3. VBI takes $O(L^4)$ steps, and
4. VM takes $O(L^3)$ steps,

and so the total run time is $O(L^4)$.

The most practical way to reduce the run time to $O(L^3)$ is to limit the size of a bulge or interior loop to some fixed number d, usually about 30. This is achieved by limiting the search in Equation 11.3 to $2 < i'-i+j-j'-2 \leq d$.

12.16 Modification of multi-loop energy

In Equation 11.4 we used a constant energy function for multi-loops. More generally, we can use the following function

$$e(\text{multi-loop}) = a + b \cdot k' + c \cdot k,$$

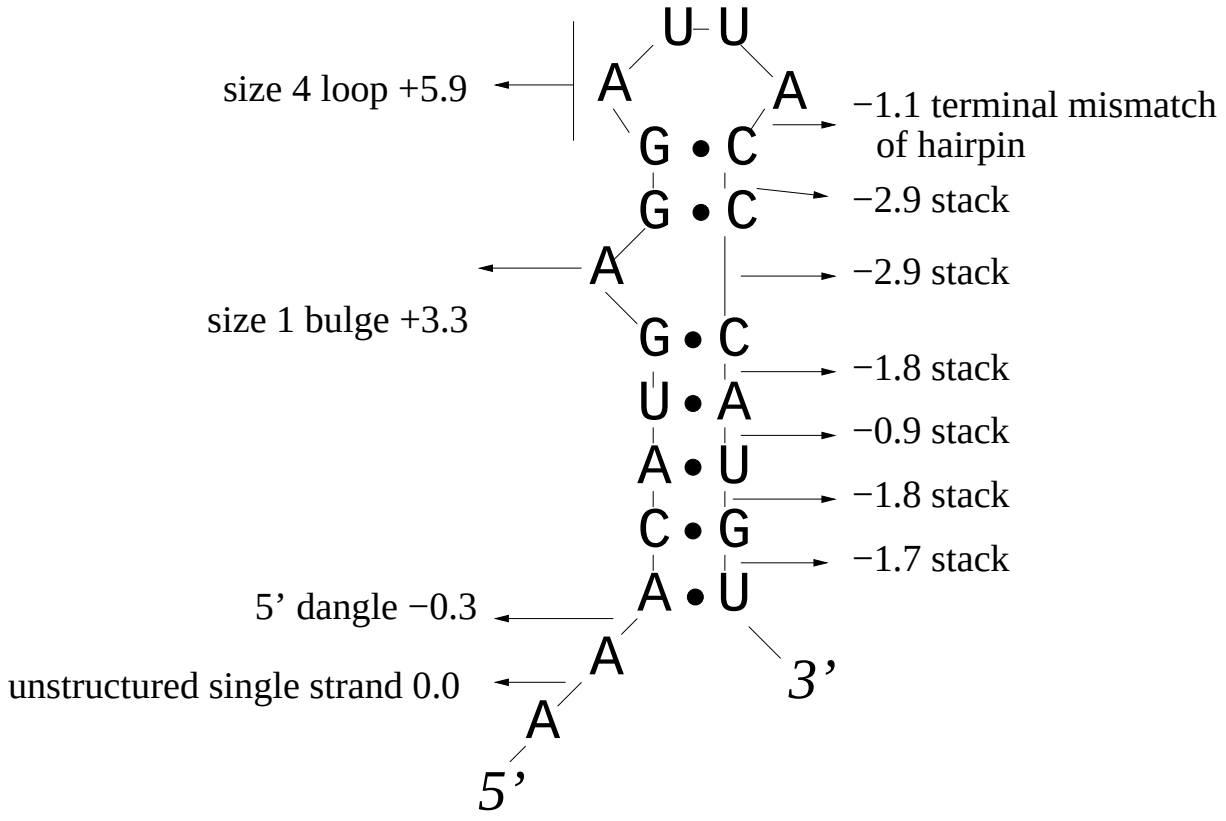
where a, b and c are constants and k' is the number of unpaired bases in the multi-loop.

This is a convenient function to use because, similar to the introduction of affine gap penalties in sequence alignment, a cubic order algorithm remains possible.

A number of additional modifications to the algorithm can be made to handle the stacking of single bases. These modifications lead to better predictions, but are beyond the scope of our lecture.

12.17 Example of energy calculation

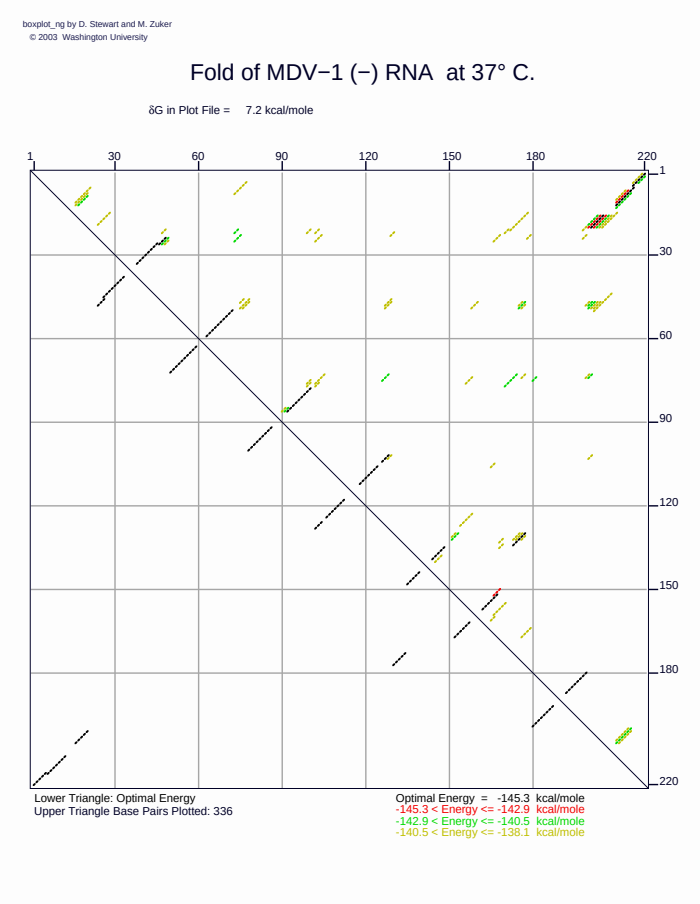
Here is any example of the full energy calculation for an RNA stem loop (the wild type *R*17 coat protein binding site):



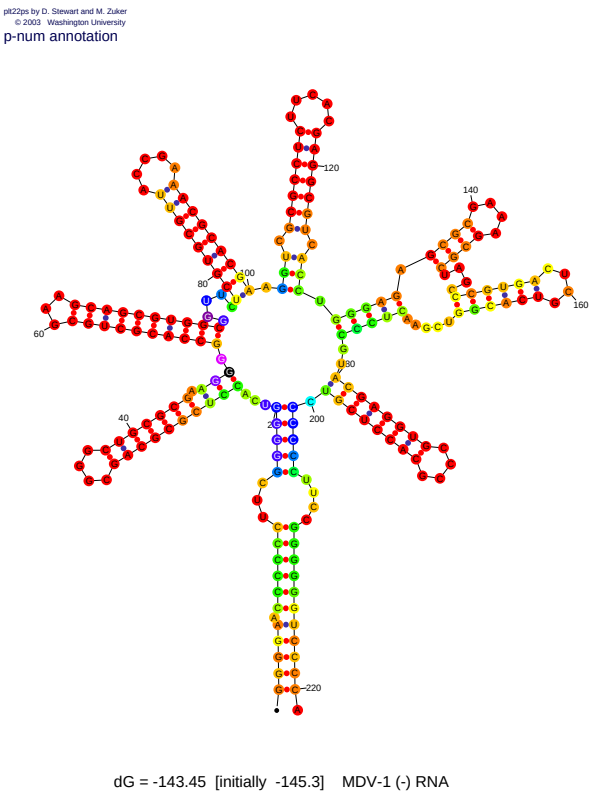
Overall energy value: −4.2 kcal/mol

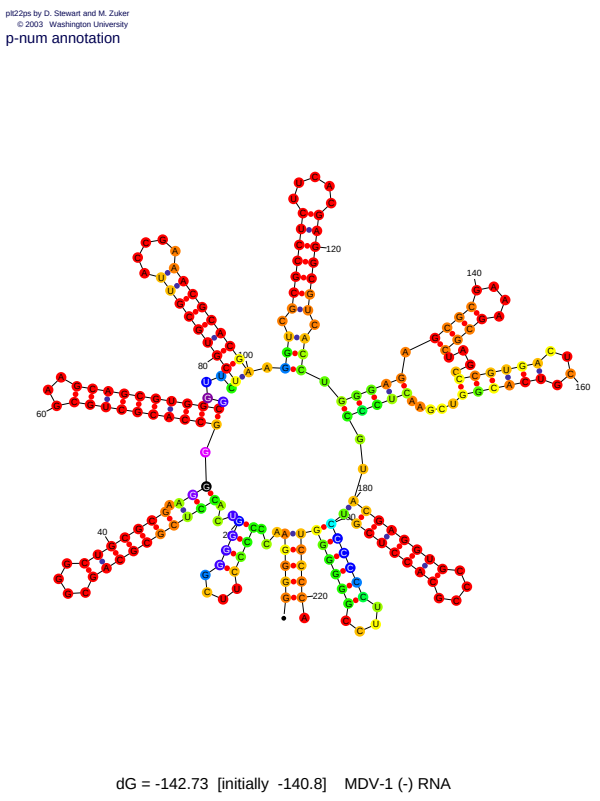
12.18 Suboptimal solutions

Mfold produces as output also a dotplot containing suboptimal solutions, that means it contains all base pairs that can take part in a folding within W energy of the optimal one.



The solutions can differ quite a bit, although their energies are not very different. The first structure has the minimum free energy of −143.45 kcal/mol, whereas the second structure has a free energy of −142.73 kcal/mol.





However, the suboptimal foldings output is somewhat arbitrary as follows:

1. Compute $\Delta G(i, j)$ as the minimum free energy that contains the base pair (i, j). Choose the top pair of the current list and compute the best folding the pair is contained in.
2. Delete all base pairs in the computed folding as well as all within a distance of W.
3. The computed folding is retained if it contains at least W base pairs that were not found in previous foldings.

Quoting from [Eddy04]:

> Dynamic programming algorithms for RNA folding are guaranteed to give the mathematically optimal structure. Any lack of prediction accuracy is more the scoring system's problem than the algorithm's problem. The fundamental trouble seems to be that the thermodynamic model is only accurate to within maybe 5-10%, and a surprising number of alternative RNA structures lie within 5-10% of the predicted global energy minimum. It's therefore hard for a single sequence folding algorithm to resolve which of the plausible lowest-energy structures is correct. Much current research focuses on adding more biological information to the scoring model to further constrain RNA structure predictions.