Helices indices of RNAs

Jiabin Huang
Schattenwald

Genetics & Experimental Bioinformatics
University Freiburg
Institute of Biology III

March 16, 2011

Outline

- an overview of RNA secondary structure elements and a classical RNA secondary structure prediction algorithm
- introducing concept of abstract shapes
- basic ideas about the helices indices
- outlook

Secondary structure elements of RNA

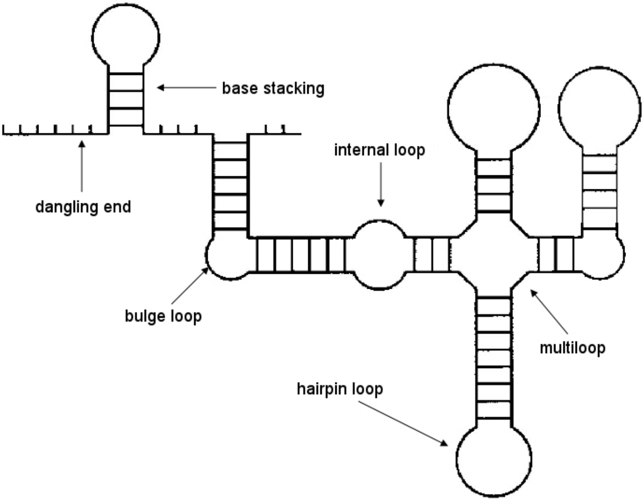


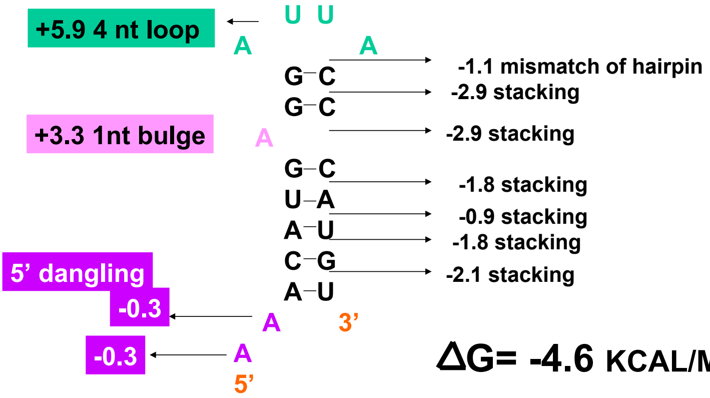
Figure: Secondary structure elements of RNA, all double stranded regions are also called as "helices"

Classical secondary structure algorithms (Zuker 1981)

facts about Zuker algorithm

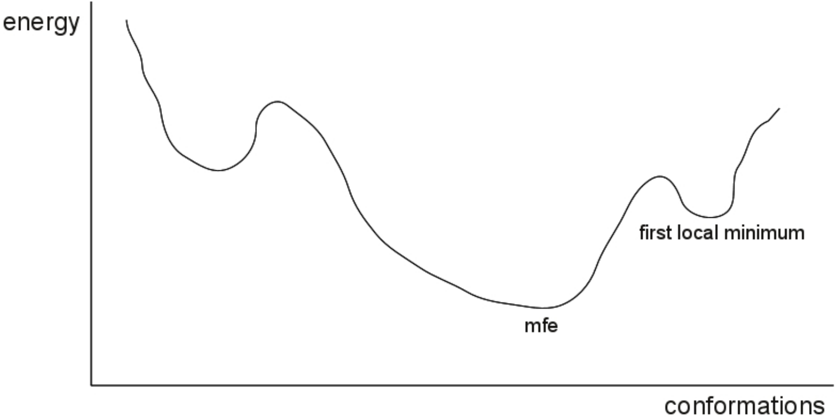
- first described by Zuker and Stiegler in 1981
- basic idea:
  1. a RNA sequence can be folded into many different secondary structure
  2. for every secondary structure, we can calculate a free energy
  3. after that, the algorithm choose the structure with the minimum free energy
- the runtime is $O(n^3)$
- can get only one solution

Free energy computation example



ΔG= -4.6 KCAL/MOL

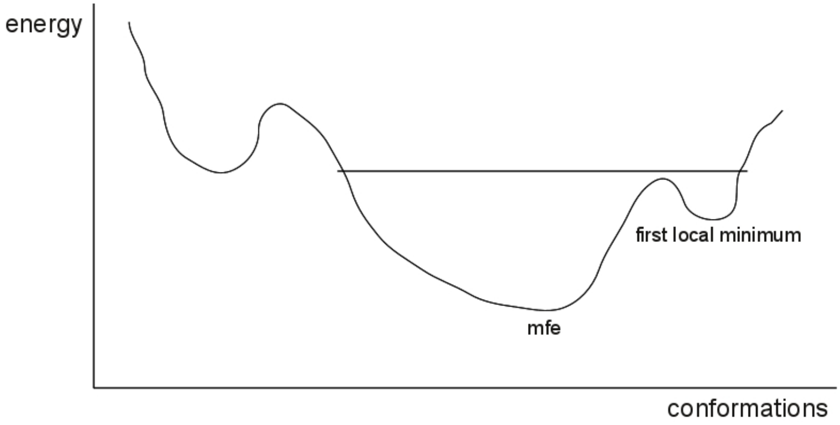
energy landscape



Suboptimal structures

- the native structure is not always the one with the lowest predicted free energy.
- but it must not be far away from the mfe point and it is normally a local minimum
- how to find the native structure: enumerate all suboptimal structures within a given energy range

energy landscape setting a range



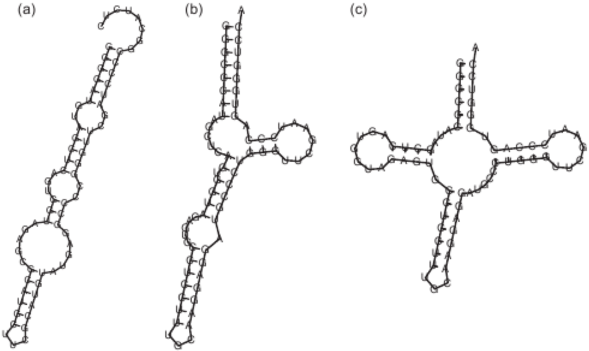
but the number of suboptimal structures grows exponentially with the energy range considered.

Introducing abstract shape

Solution: further classify secondary structures space within the energy range with different approaches.

- abstract shape is one approach in this direction
- developed by Voss and Giegerich
- intial idea:
  1. the user is usually only interested in structures that show fundamental differences
  2. small changes, such as additional base pairs or changing bulge loops are of minor significance
- central to this approach: do not care about all details of the structures and abstract from some types secondary structure elements and length of them
- each shape has a representative structure called shrep (with minimum free energy within the shape class)

Abstract shapes, energy range: 5 kcal/mol



(d)

```
Shape     GGGCCCAUAGCUCAGUGGUAGAGUGCCUCCUUUGCAAGGAGGAUGCCCUGGGUUCGAAUCCCAGUGGGUCCA
[]        (((((((((((((((.((((.....(((((((...))))))).))))))))))).........)))))))).  -35.9 kcal/mol
[[][]]    ((((((((.....((.((((.....(((((((...))))))).))))))(((.......))).)))))))).  -32.2 kcal/mol
[[][][]]  ((((((...((((.......)))).(((((((...))))))).....(((((.......))))).)))))).  -31.7 kcal/mol
```

Drawback of abstract shape

Drawback of abstract shape

- abstract shape is position independent

Consequence of the drawback

make the current implementation of shape abstraction unsuitable for the analysis of folding landscapes in a detailed fashion

Example

```
        AACUAAAACAAUUUUUGAAGAACAGUUUCUGUACUUCAUUGGUAUGUAGAGACUUC
-9.00   .......................((((((((((((.....)))))..)))))))..   []
-10.70  ..((...((((((..(((((.((((...)))).)))))..))).)))..)).....   []
```

develop a new structure abstraction: helices indices

The straightforward idea to overcome the drawback is to develop a new structure abstraction that includes the information of positions of helices

Which secondary structure element should be recorded?

- hairpin loop
- multiloop
- bulge or internal loops
- any combinations of them

Which position of this element should be recorded?

- i
- j
- i,j
- (i+j)/2

helices positions example

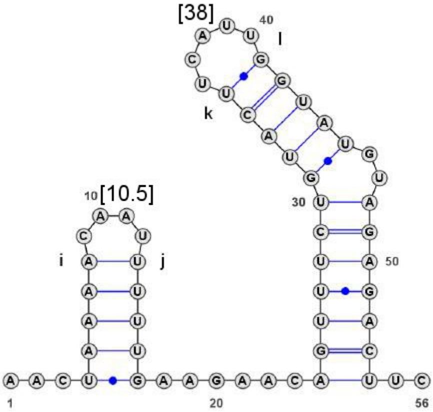


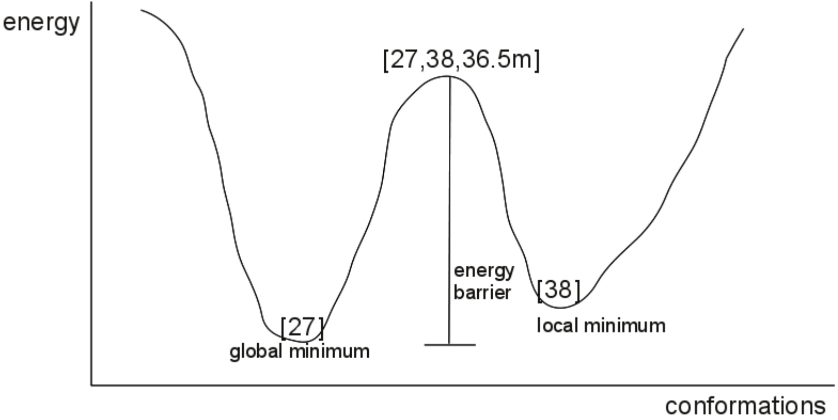
Figure: The structure is composed of two helices which are closed by hairpin loops (i,j) and (k,l), respectively. The positions are: i=8, j=13, k=35 and l=41. Thus, this structure would be abstracted to [10.5,38]

Output from the first version of helices indices

Helices indices, energy range: 10 kcal/mol

```
      AACUAAAACAAUUUUUGAAGAACAGUUUCUGUACUUCAUUGGUAUGUAGAGACUUC
-10.7..((...((((((..(((((.((((...)))).)))))..))).)))..))..... [27] *
-9.0 ......................((((((((((((.....)))))..)))))))..  [38] *
-7.7 ...(((((....))))).....((((((((((((.....)))))..)))))))..  [10.5,38]
-7.4 ........(((...))).....((((((((((((.....)))))..)))))))..  [13,38]
-7.1 ..........(((..(((((.((((...)))).)))))..)))..((....))...  [27,49.5]
-6.7 .......((((((..(((((.((((...)))).)))))..))).))).((....))  [27,52.5]
-6.7 ..((..((......))..))...((((((((((((.....)))))..)))))))..  [11.5,38]
-6.6 ..((.(((.....)))..))...((((((((((((.....)))))..)))))))..  [11,38]
-6.5 ...............(((((.((((...)))).)))))..(((........)))..  [27,47.5]
                               ...
-2.4 .....((...(((..(((((.((((...)))).)))))..)))..((....)))).  [27,49.5,30.5m]
-2.4 ..(((((....))).(((((.((((...)))).))))).............)).....  [9.5,27,27m]
-2.4 .......................((((((((((......)).....))))))))..  [36.5]
-2.4 ...............(((((.((((...)))).)))))((..........))....  [27,45.5]
-2.4 .......................((((((((...((....))....))))))))..  [38.5]
-2.3 .((((.((....)).(((((.((((...)))).))))).))))..((....))...  [10.5,27,22.5m,49.5]
-2.3 ....((....(((..(((((.((((...)))).)))))..)))..((....)))).  [27,49.5,30m]
-2.3 ..((.(((.....)))((((.((((...)))).))))............)).....  [11,27,27m]
-2.3 ................((((..(((...)))((((.....))))........))))  [27,38,36.5m] *
                               ...
-0.7 ..((...........(((((.((((...)))).))))).((...))...)).....  [27,43,27m]
-0.7 ....(((....))).........(((((((..((...........)))))))))..  [9.5,40]
-0.7 .........((.(((....)))..(((((((((((.....)))))..)))))))).  [17.5,38,32.5m]
-0.7 .((((.((....)).(((((.((((...)))).))))).)))).....((....))  [10.5,27,22.5m,52.5]
-0.7 ....((((...))))........(((((((..((...........)))))))))..  [10,40]
-0.7 ((.........))..(((((.((((...)))).)))))..(((........)))..  [7,27,47.5]
-0.7 ................((((.....)))).((.((..((......))..))))...  [23,42.5]
-0.7 ........((((......((((....)))).......))))....((....))...  [24.5,49.5]
-0.7 ..((....(((...))).............(((((.....)))))....)).....  [13,38,27m]
-0.7 ..(((.......(((....)))........(((((.....))))).))).......  [17.5,38,26m]
-0.7 ....((.........((.....))(((((((((((.....)))))..)))))))).  [20,38,30m]
```

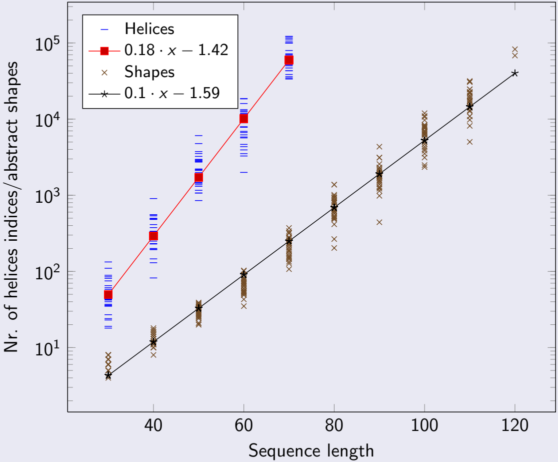
energy barrier



Problem

The helices indices space grows a lot faster than the abstract shapes space

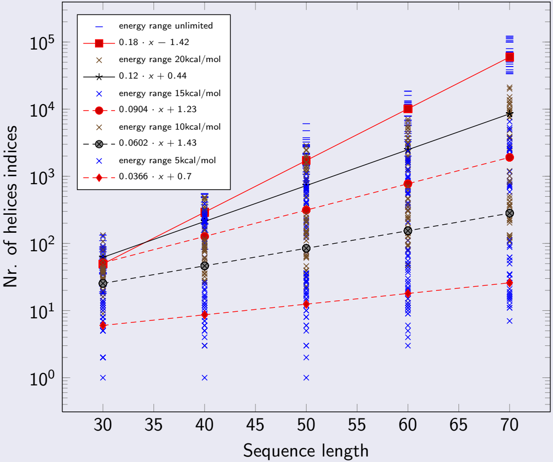
Comparison of helices indices space with abstract shapes space



Solution of the problem

One of the solution: we can limit the helices space by setting an energy range on it

Comparison of helices indices space setting different energy ranges



Outlook

- develop a new structure abstraction (helices indices)
- implement a software based on the idea
- the software will be evaluated by benchmark program
- design a RNA class predictor

End

- Thanks a lot for your attention !
- Questions ?