A-040

On Computation of Minimum Free Energy and Partition Function of Multiple Nucleic Acid Sequences

Keiichiro Takahashi† Masami Hagiya†

Abstract. Given the base sequence of a DNA/RNA molecule, the folding problem is to predict the minimum free energy structure of the molecule. To capture the thermodynamic behavior of the molecule, its partition function is also required. These two functions, i.e., the fold function and the partition function, are implemented by dynamic programming in the Vienna RNA Package, which is widely used to analyze structures of DNA/RNA. However, the Vienna Package can only deal with a single sequence. In order to estimate interactions among multiple DNA/RNA molecules, we introduced virtual bases to concatenate multiple sequences into a single one, and extended the above two functions to cope with sequences containing virtual bases. The extensions have been successfully implemented as modifications to the Vienna RNA Package.

1. Introduction

Several programs for predicting the secondary structure of a given nucleic acid sequence (i.e, RNA or DNA) have been developed [2, 1]. They are all based on the Nearest Neighbour Thermodynamics Model (NNTM). This energy model assumes that the free energy of each component loop depends only on the loop type, the loop length, the base pairs on the loop and the free bases immediately adjacent to these bases, and the free energy of a secondary structure is approximated by the sum of the free energies of its component loops. Note that any secondary structure can be uniquely decomposed into elementary component loops such as stacked pair, interior loop, bulge loop, hairpin loop and multi-loop, and external bases which do not belong to any loop. Therefore, the free energy of structure S can be written as

$$F(S) = \sum_{L \in S} E(L) \qquad (1)$$

in the energy model, where L is any component loop and $E(L)$ is its energy.

The distribution of secondary structures formed by a DNA/RNA molecule at the equilibrium state depends on the partition function of the sequence and the free energy of each structure. The partition function Q is defined as the summation of the Boltzmann factors of all secondary structures. And the function can be converted to the next form [3]:

$$Q_{ij} = 1.0 + \sum_{p,q \;\; i<p<q\leq j} Q_{i,p-1} Q^b_{pq}, \qquad (2)$$

where

$$Q^b_{ij} = \sum_{\text{loops } L \text{ closed by } i.j} e^{-E(L)/kT} \prod_{\text{interior pairs } p.q} Q^b_{pq}.$$

Equations (1) and (2) give us elegant algorithms of the fold function and the partition function [3], and these functions are implemented in the Vienna Package through dynamic programming by Walter Fontana and Ivo Hofacker [1].

Using the Vienna Package, our group has constructed some DNA machines consisting of several DNA sequences [4, 5]. For designing such DNA machines, our group has modified the package by introducing virtual bases which concatenate multiple sequences together so that they can be treated as a single one in the fold function and the partition function. However, the processing of multi-loops containing virtual bases was not implemented, because the sequence design of the DNA machines did not require it. Recently, our group aims at more flexible and robust devices which are comprised of multiple DNA sequences and have the possibility of making multi-loops carrying virtual bases in the process of their sequence design. Therefore, in addition to the previous modification, we fully completed the extension of the Vienna Package for virtual bases including the processing of multi-loops.

2. Implementations

Before describing how we extended the Vienna Package, let us define some functions and dynamic programming arrays. `eH(i,j)` and `eSBI(i,j,p,q)` are functions which return the free energy of a 1-loop (i.e., hairpin loop) closed by $i.j$ and a 2-loop (i.e., stacked pair, interior loop or bulge loop) with a closing pair $i.j$ and an interior pair $p.q$, respectively. The energy of a multi-loop is usually approximated by ΔG(multi-loop) = `MLclosing` + `MLintern`×P + `MLbase`×L, where `MLclosing`, `MLintern` and `MLbase` are empirical constants, P is the number of base pairs on the multi-loop and L is the number of free bases on the loop. `V[i,j]` is the free energy array of the minimum free energy structure of a subsequence $N_i N_{i+1} \ldots N_j$, assuming N_i pairs with N_j ($1 \leq i < j \leq n$, n is the whole length of a sequence). `VM[i,j]` is the free energy array which gives ΔG(multi-loop), assuming $i.j$ is a closing pair on a multi-loop. And `W[i,j]` is also the free energy array of the optimal structure of a subsequence $N_i N_{i+1} \ldots N_j$. Here is the pseudocode of the original fold function in the Vienna Package.

```
for(j=2; j<=n; j++)
  for(i=j-1; i>=1; i--){
    V[i,j]  = min{
        eH(i,j),
        min( i<p<q<j : eSBI(i,j,p,q) + V[p,q] ),
        min( i<k<j-1 : VM[i+1,k] + VM[k+1,j-1])
                          + MLintern + MLclosing }
    VM[i,j] = min{
        VM[i+1,j] + MLbase,
        VM[i,j-1] + MLbase,
        V[i,j] + MLintern,
        min( i<k<j-1 : VM[i,k] + VM[k+1,j] ) }
    W[i,j]  = min{
        V[i,j],
        min( i<k<j-1 : W[i,k] + W[k+1,j] ) }
  }
```

†JST CREST and Department of Computer Science, Graduate School of Information Science and Technology, University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, JAPAN.
{keiichi, hagiya}@is.s.u-tokyo.ac.jp

```
return W[1,n]
```

The computation starts with all the arrays set to infinity and ends when it renders `W[1,n]`. Once the above procedure is done, the optimal structure can be extracted by the technique of backtracking as usual in dynamic programming algorithms.

As mentioned above, to analyse the seconary structures formed by multiple sequences through hybridization, our group has introduced virtual bases. The key point of this modification, as pointed out by Kubota and also mentioned by Zuker recently [2], is that the programs must treat loops containing virtual bases as external loops. According to this point, we extended the Vienna Package subsequently to the Uejima and Kubota's modification. Here is the pseudocode of the extended fold function, which can estimate the optimal structure of given multiple nucleic acid sequences as well as a single one at will.

```
for(j=2; j<=n; j++)
  for(i=j-1; i>=1; i--){
    V[i,j]  = min{
      if(virtual bases exist on 1-loop)
        the energy of the external loop,
      else eH(i,j),
      if(virtual bases don't exist on 2-loop)
        min(i<p<q<j : eSBI(i,j,p,q) + V[p,q] ),
      min( i<k<j-1 : VM[i+1,k] + VM[k+1,j-1])
                       + MLintern + MLclosing ),
      ZVM[i+1,j-1]}
    VM[i,j] = min{
      if(Ni is a real base) VM[i+1,j] + MLbase,
      else INF,
      if(Nj is a real base) VM[i,j-1] + MLbase,
      V[i,j] + MLintern,
      min( i<k<j-1 : VM[i,k] + VM[k+1,j] ) }
    ZVM[i,j] = min{
      ZVM[i+1,j],
      ZVM[i,j-1],
      if(Ni is a virtual base) NZVM[i+1,j],
      if(Nj is a virtual base) NZVM[i,j-1],
      min( i<k<j-1 : ZVM[i,k]+ZVM[k+1,j] ),
      min( i<k<j-1 : NZVM[i,k]+ZVM[k+1,j] ),
      min( i<k<j-1 : ZVM[i,k]+NZVM[k+1,j] ) }
    NZVM[i,j] = min{
      if(Ni is a real base) NZVM[i+1,j],
      else INF,
      if(Nj is a real base) NZVM[i,j-1],
      min( i<k<j-1 : NZVM[i,k]+NZVM[k+1,j]) }
    W[i,j]  = min{
      V[i,j],
      min( i<k<j-1 : W[i,k] + W[k+1,j] ) }
  }
return W[1,n]
```

`ZVM[i,j]` and `NZVM[i,j]` are the free energy arrays which are newly introduced in order to handle multiloops and 2-loops carrying virtual bases. `ZVM[i,j]` is the optimal energy of $N_iN_{i+1}...N_j$ given that the pair $i.j$ closes the external loop which contains virtual bases in $N_{i+1}...N_{j-1}$. And `NZVM[i,j]` is also the optimal energy of $N_iN_{i+1}...N_j$ given that the pair $i.j$ closes the external loop which does not contain virtual bases in $N_{i+1}...N_{j-1}$. The energy contribution of dangling ends and the energy penalty of terminal-ATs have been considered in the practical codes whenever external loops appear.

Calculation of the partition function greatly resembles that of the fold function. More accuracy, the partion function can be computed by replacing minima with sums, sums with products and initializations by infinity with initializations by zero in the fold algorithm, and preparing some auxiliary arrays in order to avoid counting an identical structure twice or more. This means the extension of the partition function is almost the same as the optimal folding case. So let us omit to describe the code here.

3. Simulation

In this section, we show the results of our energy calculations for a simple instance:
5'-GCAATCAGTGAZZZZCATCGACZZZZACGATG
TACATCGTAGCCATACGGCZZZZGTACGATGTC
CTGCTAZZZZAGGGCATACCATGCAGGZZZZCA
CTGATTGC-3'
This instance consists of six sequences linked with virtual bases. Our fold function returned -42.02 [kcal/mol] with ((((((((((.....(((((.......)))))..((((((((((....))).....))))))).(((((.......((.....))..)))))....)))))))))) as its optimal structure. This minimum free energy structure contains virtual bases in its 1-loop, 2-loop and multi-loop. And our partition function returned -43.78 [kcal/mol] as the ensemble energy, when the multiple sequences and the optimal structure are inputted.

4. Discussion

In this study, we successfully extended the Vienna Package so that it can analyze multiple nucleic acid sequences as well as a single one. This extension will widely contribute to the field of not only bioinformatics but also DNA computing and DNA nanotechnology.

One serious problem in the current algorithm is that it does not cope with pseudoknots, as the Vienna Package does not do so. For multiple sequences, however, a secondary structure may contain pseudoknots with virtual bases, which are not real ones. Furthermore, the existence of such pseudoknots depends on the order of concatenating sequences. This implies that we should try all the orders for concatenating sequences for finding the minimum free energy structure.

So far DNA parameters of the NNTM for the Vienna Package are yet to be gained sufficiently in our laboratory. This brings about a little difference between the outcomes of our energy calculations and those of the Vienna group's calculations. However, the Vienna group is supposed to release the DNA parameters in the near future. It will make our outcomes coincide with the Vienna group's .

References

[1] Hofacker, I.L., Fontana, W., Stadler, P.F, Bonhoeffer, L.S., Tacker, M, Schuster, P.: Fast folding and comparison of RNA secondary structures, Monatsh. Chem. 125, 167-188 (1994)

[2] Michael Zuker: Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.* 31 (13), 2003, pp. 3406-3415.

[3] J.S. McCaskill: The equilibrium partition function and base pair binding probabilities for RNA secondary structure. Biopolymers, 29:1105-19, 1990

[4] Hiroki Uejima and Masami Hagiya: Secondary Structure Design of Multi-state DNA Machines Based on Sequential Structure Transitions, *DNA9, Preliminary Proceedings*, 2003, pp.80-91.

[5] Mitsuhiro Kubota, Kazumasa Ohtake, Ken Komiya, Kensaku Sakamoto and Masami Hagiya: Branching DNA Machines Based on Transitions of Hairpin Structures, *Proceedings of the 2003 Congress on Evolutionary Computation*, 2003, pp.2542-2548.