APPLICATIONS NOTE

2005
Pages 1–2

Local Base Pairing Probabilities in Large RNAs

Stephan Bernhart [a], Ivo L. Hofacker [a,*], Peter F. Stadler [b,a,c]

[a]Institut für Theoretische Chemie, Universität Wien, Währingerstr. 17, A-1090 Wien, Austria
[b]Bioinformatics Group, Department of Computer Science and Interdisciplinary Center for Bioinformatics, University of Leipzig, Härtelstr. 16-18, D-04107 Leipzig, Germany
[c]Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM 87501

ABSTRACT

Summary: The genome-wide search for and analysis of non-coding RNAs requires efficient methods to compute and compare local secondary structures. Since the exact boundaries of such putative transcripts are typically unknown, arbitrary sequence windows have to be used in practice. Here we present a method for robustly computing the probabilities of local base pairs from long RNA sequences independent of the exact positions of the sequence window.

Availability: The program `RNAplfold` is part of the `Vienna RNA Package` and can be downloaded from `http://www.tbi.univie.ac.at/RNA`.

Contact: Ivo Hofacker, Tel: ++43 1 4277 53736,
Fax: ++43 1 4277 52793, `ivo@tbi.univie.ac.at`

1 INTRODUCTION

Computational approaches to detecting and classifying structured RNAs at genomic scales require efficient ways of computing *local* RNA secondary structures for both computational and biological reasons: (i) Long-range base pairs in large transcripts are disfavoured kinetically relative to short-range pairs (Flamm *et al.*, 2000). (ii) Global approaches to RNA folding are limited to sequence length ≤ 20000 on most hardware because of memory consumption. (iii) In general, the exact boundaries of the RNAs are unknown, so that global folds cannot add to the accuracy of the structure prediction relative to folding individual sequence windows.

A recent algorithm for microRNA detection is based upon the idea to consider the stability of secondary structure against changes in the immediate environment (Pfeffer *et al.*, 2005; Sewer *et al.*, 2005). More precisely, this approach considers the frequency with which a certain base pair (i, j) occurs in local minimum energy structures that are computed from sequence windows with a given size L. Here we combine this idea with a recently developed algorithm for local minumum energy structure predictions (Hofacker *et al.*, 2004b). More precisely, we derive recursions for the average equilibrium probability of a base pair (i, j) over all fixed-size sequence windows.

2 ALGORITHM

Denote by Z_{ij} the partition function over all secondary structures on the sequence interval $[i, j]$, write $\widehat{Z}_{ij}$ for the partition function subject to the constraint that i and j pair and let p_{ij} be the probability that i and j are actually paired in thermodynamic equilibrium.

The standard backtracking procedure for the partition function folding algorithm (McCaskill, 1990) can be expressed as

$$p_{ij} = \frac{Z_{1,i-1}\widehat{Z}_{i,j}Z_{j+1,n}}{Z_{1,n}} + \sum_{k<i}\sum_{l>j} p_{kl}\Xi_{ij,kl}\,. \tag{1}$$

Here $\Xi_{ij,kl}$ is the ratio of the two partition functions $\widehat{Z}_{ij,kl}$ with the constraint that both i, j and k, l pair, and $\widehat{Z}_{kl}$. The first term describes the case in which the (i, j) pair is external, i.e., not enclosed by another pair, the second (sum) term considers all possible base pairs (k, l) that could enclose (i, j). In the simplest case, i.e., for energies dependent on individual base pairs only, we have

$$\widehat{Z}_{ij,kl} = Z_{k+1,i-1}\widehat{Z}_{ij}Z_{j+1,l-1}\zeta_{kl} \tag{2}$$

where ζ_{kl} is the Boltzmann factor of the pairing energy for the closing the base pair (k, l). In the standard energy model, described e.g. by Mathews *et al.* (1999), $\widehat{Z}_{ij,kl}$ is a sum over contributions for the different loop types (interior loops, bulges, and multi-branched loops) as detailed by McCaskill (1990); for given i, j, k, l it can be computed in constant time from the tabulated partition functions of subsequences.

Let us now turn to interactions localized within a sequence window. We denote by $Z^{u,L}_{ij}$ the partition function over all secondary structures on the sequence interval $[i, j]$ when the sequence window $[u, u + L]$ is folded. Similarly, $\widehat{Z}^{u,L}_{ij}$ denotes the partition function with the additional constraint that positions i and j are paired. Furthermore, we write $p^{u,L}_{ij}$ for the probability that i and j form a base pair when the sequence window $[u, u + L]$ is folded.

Since the partition functions on a subsequence are independent of the external structures as long as the subsequence is contained in the folded sequence window, we have

$$Z^{u,L}_{ij} = \begin{cases} Z_{ij} & \text{if} \quad [i,j] \subseteq [u, u+L] \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Furthermore, we observe that $p^{u,L}_{ij} = 0$ unless $[i, j] \subseteq [u, u + L]$. We can immediately restrict equ.(1) to a sequence window $[u, u+L]$ since the recursions for Z_{ij} depend only on sub-sequences within the interval $[i, j]$ (McCaskill, 1990). Thus

$$\begin{aligned} p^{u,L}_{ij} &= \frac{Z^{u,L}_{1,i-1}\widehat{Z}^{u,L}_{i,j}Z^{u,L}_{j+1,n}}{Z^{u,L}_{1,n}} + \sum_{k<i}\sum_{l>j} p^{u,L}_{kl}\Xi^{u,L}_{ij,kl} \\ &= \frac{Z_{u,i-1}\widehat{Z}_{i,j}Z_{j+1,u+L}}{Z_{u,u+L}} + \sum_{k<i}\sum_{l>j} p^{u,L}_{kl}\Xi_{ij,kl}\,. \end{aligned} \tag{4}$$

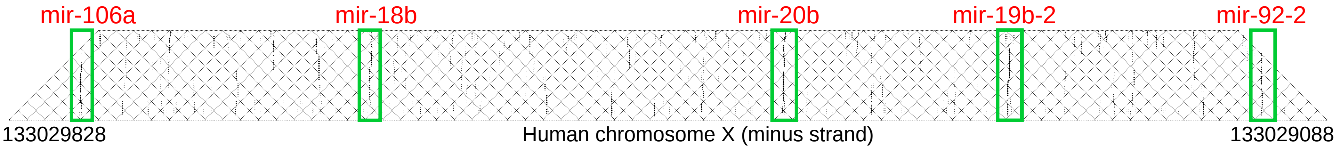

Fig. 1. Local structures in small region of the human X chromosome containing five microRNAs annotated in `miRBase 7.1.` (Griffiths-Jones, 2004).

Next, we define the average probability of an (i,j) pair over all folding windows containing the sequence interval $[i,j]$ as:

$$\pi^L_{ij} = \frac{1}{L-(j-i)+1}\sum_{u=j-L}^{i} p^{u,L}_{ij}\,. \tag{5}$$

For $i+L>n$ and $j-L<1$ the sequence windows are shorter, hence equ.(5) has to be modified accordingly. Substitution of equ.(4) yields in the generic case

$$\begin{aligned}\pi^L_{ij} &= \underbrace{\frac{1}{L-(j-i)+1}\sum_{u=j-L}^{i}\frac{Z^{u,L}_{1,i-1}\widehat{Z}^{u,L}_{i,j}Z^{u,L}_{j+1,n}}{Z^{u,L}_{1,n}}}_{\pi^{*L}_{ij}}\\ &\quad+\frac{1}{L-(j-i)+1}\sum_{u=j-L}^{i}\sum_{k<i}\sum_{l>j}p^{u,L}_{kl}\Xi_{ij,kl} \qquad (6)\\ &=\pi^{*L}_{ij}+\sum_{k=j-L}^{i-1}\sum_{l=j+1}^{i+L}\sum_{u=l-L}^{k}\frac{p^{u,L}_{kl}\Xi_{ij,kl}}{L-(j-i)+1}\\ &=\pi^{*L}_{ij}+\sum_{k=j-L}^{i-1}\sum_{l=j+1}^{i+L}\frac{L-(k-l)+1}{L-(j-i)+1}\pi^L_{kl}\Xi_{ij,kl}\,.\end{aligned}$$

Again, modified expressions apply to the 5' and 3' ends of the sequence.

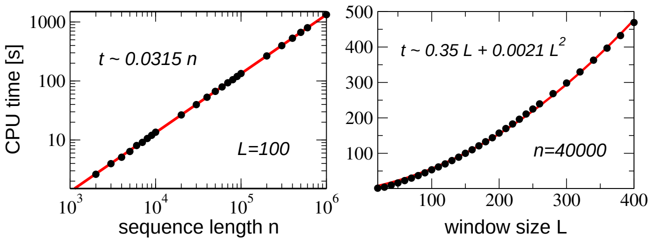

Fig. 2. Performance measurments of our C implementation of `RNAplfold` on a Pentium 4 with 3.2 GHz confirms the theoretical scaling of the CPU time as $O(n \times L^2)$.

3 PERFORMANCE AND APPLICATIONS

Eq.(6) implies that the $n \times L$ matrix π^L_{ij} can be computed in $\mathcal{O}(n \times L^3)$ time and $\mathcal{O}(n \times L)$ memory for any fixed L. This can be improved further by observing that, for fixed i and j, the values of the interior sum can be stored as a function of k in an array of size $\mathcal{O}(L)$, thereby reducing the computational effort to $\mathcal{O}(n \times L^2)$. It is also not necessary to keep the whole matrix π^L_{ij} in memory, which reduces memory requirements to $\mathcal{O}(n + L^2)$.

The recursions (6) are implemented in the ANSI C program `RNAplfold`. Dot plots representing the values of π^L_{ij} are provided in `Postscript` format and can be used to visually inspect the results, Fig. 1. The program is fast enough for genome-wide applications at least when run on a Linux cluster. Fig. 2 shows that about 3Mb of genomic sequence per hour can be scanned on a single CPU.

`RNAplfold` has a number of obvious applications which we are currently exploring. Along the lines of Pfeffer *et al.* (2005) it can be used to efficiently retrieve candidates microRNA and other structured RNA from genomic sequence data. More generally, however, genome-wide tables of π^L_{ij} provide valuable *a priori* structure information that can be exploited by algorithms that search for RNA sequence/structure patterns such as e.g. a local variant of `marna` (Siebert & Backofen, 2005) or `pmmatch` (Hofacker *et al.*, 2004a). As such it provides as starting point for alternative approaches to RNA annotation strategies that are not solely based on comparative genomics as `RNAz` or `qrna`.

Acknowledgment. Financial support by Austrian GEN-AU bioinformatics integration network sponsored by bm:bwk (200.067/3-VI/1/2002) and the German *DFG* Bioinformatics Initiative (BIZ-6/1-2) is gratefully acknowledged.

REFERENCES

Flamm, C., Fontana, W., Hofacker, I. & Schuster, P. (2000). RNA folding kinetics at elementary step resolution. *RNA*, **6**, 325–338.

Griffiths-Jones, S. (2004). The microRNA registry. *Nucl. Acids Res.*, **32**, D109–D111.

Hofacker, I. L., Bernhart, S. H. F. & Stadler, P. F. (2004a). Alignment of RNA base pairing probability matrices. *Bioinformatics*, **20**, 2222–2227.

Hofacker, I. L., Priwitzer, B. & Stadler, P. F. (2004b). Prediction of locally stable rna secondary structures for genome-wide surveys. *Bioinformatics*, **20**, 191–198.

Mathews, D., Sabina, J., Zuker, M. & Turner, H. (1999). Expanded sequence dependence of thermodynamic parameters provides robust prediction of RNA secondary structure. *J. Mol. Biol.*, **288**, 911–940.

McCaskill, J. S. (1990). The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, **29**, 1105–1119.

Pfeffer, S., Sewer, A., Lagos-Quintana, M., Sheridan, R., Sander, C., Grasser, F. A., van Dyk, L. F., Ho, C. K., Shuman, S., Chien, M., Russo, J. J., Ju, J., Randall, G., Lindenbach, B. D., Rice, C. M., Simon, V., Ho, D. D., Zavolan, M. & T., T. (2005). Identification of microRNAs of the herpesvirus family. *Nat. Methods.*, **2**, 269–276.

Sewer, A., Paul, N., Pfeffer, S., Aravin, A., Landgraf, P., Tuschl, T., van Nimwegen, E. & Zavolan, M. (2005). Identification of clustered microRNAs using *ab initio* prediction method. *BMC Bioinformatics*. Under review.

Siebert, S. & Backofen, R. (2005). `MARNA`: multiple alignment and consensus structure prediction of RNAs based on sequence structure comparisons. *Bioinformatics*, **21**, 3352–3359.