Laboratoire IBISC

Biologie Intégrative et Systèmes Complexes

A fast ab initio method for predicting miRNA precursors in genomes

Sébastien Tempel, Fariza Tahi

IBISC University of Evry-Val d'Essonne, Genopole, France

RAPPORT DE RECHERCHE
IBISC-RR-2011-03
mars 2011

A fast ab initio method for predicting miRNA precursors in genomes

Seébastien TEMPEL[1] and Fariza TAHI[1]

Laboratoire IBISC, Université d'Evry-Val d'Essonne/Genopole, 523, Place des Terrasses, 91000 Evry, France
sebastien.tempel@ibisc.univ-evry.fr
fariza.tahi@ibisc.univ-evry.fr

Abstract *miRNAs are small non coding RNA structures which play important roles in biological processes. Finding miRNA precursors in genomes is therefore an important task, where computational methods are required. The goal of these methods is to select potential pre-miRNAs which could be validated or invalidated by experimental methods. Biologists need then computational methods that are able to find true pre-miRNAs without returning a big set of false pre-miRNAs. Besides, with the new generation of sequencing techniques, it is important to have fast algorithms, able to treat whole genomes in acceptable times.*

We developed an algorithm based on an original method where an approximation of miRNA hairpins are first searched, before reconstituting the pre-miRNA structure. The step of approximation allows to decrease substantially the number of possibilities, and then the time searching. miRNAFold was tested on different genomic sequences, and was compared to RNALFold from the RNA Vienna package. miRNAFold gives better or similar sensitivity, and two times better selectivity. miRNAFold is faster than RNALFold: it takes less than one minute for processing a sequence of 1MB of nucleotides, when RNALFold takes more than 5 minutes.

We present here a fast ab initio algorithm for searching for pre-miRNA precursors in genomes, called miRNAFold. miRNAFold is available at http://EvryRNA.ibisc.univ-evry.fr/

Keywords microRNA, ab intio algorithms, miRNA search

1 Introduction

MicroRNAs (miRNAs) are non-coding RNAs with only 21-25 nt in sequence length that are present in all sequenced higher eukaryotes ([3,11]). They are involved as negative regulators of gene expression at the post-transcriptional level by binding to specific mRNA targets whose translations are inhibited or down-regulated ([11]). According to the current understanding of miRNA biogenesis, miRNA are cleaved into a 60-80 nt long precursor of miRNA sequences (pre-miRNAs). The pre-miRNA, structured as an hairpin, is transported into the cytoplasm into the mature miRNA ([3]). In the RISC complex, a miRNA binds with a specific mRNA transcript and leads to the cleavage or the degradation of the mRNA.

Because the detection of miRNAs by experimental techniques is difficult and expensive and requires large amount of time, computational methods represent the first step in miRNA identification. These methods can be divided into three approaches: comparative genomics, homology-based approaches and *ab initio* approaches.

The phylogenetic conservation of some miRNAs in their primary sequence and/or their secondary structure ([3,29]) is used in comparative genomics approaches. These approaches consider multiple alignments of sequences where conserved miRNAs are searched for. Several algorithms based on this approach were developed, for example miRseeker ([21]), MiRFinder ([18]), RNAmicro ([13]), BayesMiRNAfind ([31]) and miRRim ([26]).

The increase of known miRNAs in miRBase (www.mirbase.org) ([9]) permits homology-based approaches to exploit information from both sequence and structure. For example,miRAlign ([28]) uses sequence and structure filters to predict new miRNAs. ERPIN ([22]) uses RNA alignments as weight matrices to look for homologous miRNAs.

Comparative genomics and homology-based approaches cannot detect miRNAs of unknown families and/or miRNAs with no close homologous in genomes. Furthermore, comparative approaches do not work on new

genomes that do not have a closely related sequenced species. Ab-initio methods are needed to predict new miRNAs in genomes.

Almost all existing ab initio algorithms use in a first step known methods, often RNAFold ([14]), for predicting possible secondary structures of a given structure, and then apply filters for predicting miRNAs. We can cite for instance miR-abela ([25]), Triplet-SVM ([30]), miPred ([19]) and MiRPred ([6]), all of them use secondary structures built by RNAFold for selecting pre-miRNA candidates. miR-abela predicts new pre-miRNAs that are close in the sequence to a given known pre-miRNA; it searches for pre-miRNA clusters in human, mouse and rat genomes. Triplet-SVM and miPred are algorithms that classify real and pseudo pre-miRNAs using respectively a support vector machine (SVM) and a random forest prediction model. Finally, MiRPred identifies pre-miRNA structures in the human genome using linear genetic programming.

In our knowledge, there are very few ab initio algorithms that search for pre-miRNA structures in whole genomes and all are specific to one or some genomes. It is the case for instance of miRPred and of CID-miRNA ([27]), which uses a Stochastic Context Free Grammar (SCFG) model for predicting pre-miRNAs. Both are specific to human genome.

Biologists often use RNALFold ([16]), from the Vienna RNA package, which is an optimisation of RNAFold ([14]) for a search on genomes. RNALFold searches in genomic sequences for all possible non-coding RNA secondary structures including hairpins. It consideres a sliding window where structures that have a Minimum Free Energy (MFE) are selected, using the dynamic programming approach.

In this article, we present a new ab initio method, called miRNAFold, for predicting pre-miRNA structures in any genome. We developed an algorithm which searches directly for pre-miRNA hairpins. It targets more precisely pre-miRNA structures by taking into account their characteristics, in order to (i) better select the true pre-miRNAs and (ii) reduce the time searching. The main idea is to search first for an a long stem of the hairpin, that is considered as an anchor allowing to deduce then the hairpin structure (the idea of anchor was initialy used in [7] for RNA secondary structure prediction).

miRNAFold was tested on an artificial sequence and on several real genomic sequences. We show in this article that our algorithm succeeds to predict almost all known pre-miRNAs in genomic sequences of different species. We also show that our algorithm is fast; it takes less than one minute for processing a sequence of 1MB of nucleotides, when RNALFold takes more than 5 minutes.

2 Methods

2.1 Definitions

We define a ”stem” (or ”exact stem”) as a succession of basepairings A-U, C-G and G-U. A ”Watson-Crick stem” is a stem composed only of A-U and C-G pasepairing.

We define a ”symmetrical non-exact stem” a stem composed of several exact stems separated by internal loops such as for each loop *lp* between two stems *s1* and *s2*: (i) *lp* is composed of two single strands of same size (considered as the loop size) and (ii) the size of *lp* is less than the size of *s1* and the size of *s2*.

2.2 Pre-miRNA features

Our first objective was to find features of pre-miRNAs. For this purpose, we downloaded the last version of miRBase database (Release 16, Sept 2010) ([9]), that contains 15172 miRNAs and we studied the pre-miRNAs contained in this database. We then observed several characteristics:

2.2.1 Pre-miRNA hairpins contain long stems We observed that pre-miRNAs are almost always composed of at least one long exact Watson-Crick stem. Fig. 1 Left shows that in most pre-miRNAs in miRBase, the longest succession of basepairings of Watson-Crick type is of length between 5 and 7 nt, and that almost all have a stem of size greater than 3.

2.2.2 Pre-miRNA hairpins are symmetric We also observed that most pre-miRNAs either have very few bulges or bulges of one side almost compensate with bulges of the other side (i.e. there is a similar number of nucleotides on both sides of the hairpin from the terminal loop to the extremities). Fig. 1 Center shows that the number of hairpins decreases when the gap increases. 70% of pre-miRNAs have less than 5 nucleotides in excess on one side. In other words, pre-miRNAs do not form a "curved" hairpin but form an "almost straight" hairpin.

2.2.3 Pre-miRNA hairpins can be approximated by a symmetrical non-exact stem We have observed that in almost all pre-miRNAs of miRBase, there is a symmetrical non-exact Watson-Crick stem that forms an important part of the structure, often around 40%. More than 80% of pre-miRNAs in miRBase have a symmetrical non-exact stem that represents at least 30% of their length (Fig. 1 Right).

2.2.4 Other pre-miRNA features By studying pre-miRNAs of miRBase, we observed several other characteristics. For instance we observed that there is almost never less than 3 different nucleotides in the longest stem of the hairpin, there is almost never long sub-sequences composed of a succession of same nucleotide and there is a balanced number of each of the four nucleotides and a balanced number of each possible basepair.

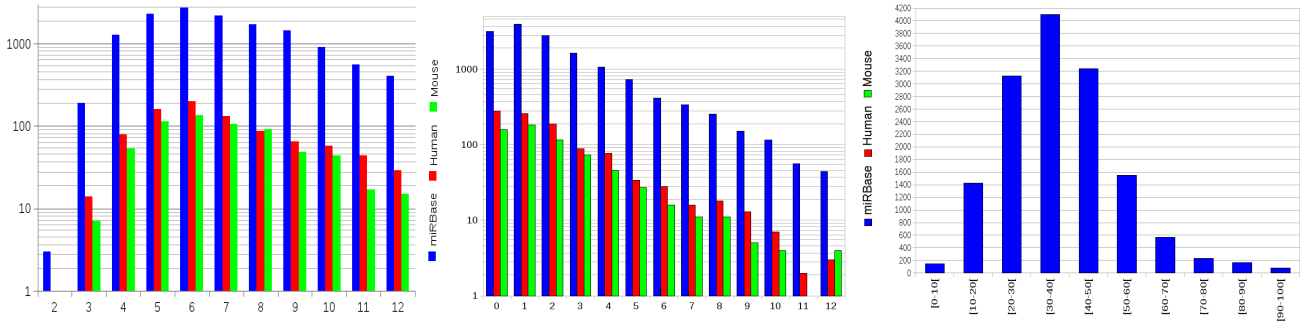

Figure 1. Left: Number of pre-miRNA hairpins (in human genome, mouse genome and in all miRBase) in function of the length of their longest stem with basepairings of type A-U and C-G. **Center:** Number of pre-miRNAs (in human genome, mouse genome and in all miRBase) having a gap of a given size, i.e. having an excess of nucleotides on one side of the hairpin. A gap of zero corresponds to a same number of nucleotides on both sides. **Right:** Number of pre-miRNAs in miRBase in function of the percentage of nucleotides covered by a symmetrical non-exact stem (with basepairings of type A-U and C-G).

2.3 Our approach

Our goal was to develop an algorithm which is able to find efficiently pre-miRNAs in whole genomes in an acceptable time. For this purpose, we adopted the following approach, which was motivated by different observations (presented above) we made on miRBase pre-miRNAs.

We consider a sliding window of a given size L sufficiently long to contain a pre-miRNA, in which we search for pre-miRNA hairpins. In a first step, we search for long exact Watson-Crick stems which verify some criteria, so they are considered as anchors of possible hairpins. Then we extend the selected stem in order to get the longest symmetrical non-exact Watson-Crick stem verifying some criteria. This longest symmetrical non-exact stem can correspond to a large portion of a pre-miRNA. It is then considered as a good approximation of a pre-miRNA hairpin, and gives the hairpin position. Possible pre-miRNA hairpins are then searched for in the subsequence associated to the selected symmetrical non-exact stem, considering the middle position of the symmetrical non-exact stem as the middle position of the hairpin.
Thus, our approach consists of three main steps applied on each window subsequence:

- Search for long stems;
- Extend the selected stems and select the longest symmetrical non-exact stems;
- Predict the secondary structure of the hairpins corresponding to the selected symmetrical non-exact stems.

At each step, several selection criteria are used, corresponding to several features observed on the exact stems, the symmetrical non-exact stems and the hairpins. Because a miRNA hairpin can present some of these features but not all, an exact stem, a symmetrical non-exact stem or a hairpin is selected when a certain percentage of the criteria are verified. This percentage is a parameter which could be set by the user. Among these criteria, we have the classical thermodynamic free energy. We calculate the MFE (minimal free energy), the MFE adjusted (i.e ratio between MFE and the length) and the MFE index (i.e ratio between MFE and C+G contents). These criteria are also used in ([12]).

2.4 The algorithm

For each sequence corresponding to a position of the sliding window, the algorithm performs the two following main steps:

2.4.1 Longest symmetrical non-exact stem searching A triangular Watson-Crick base pairing matrix *M* is built such as:

$$M(i,j) = \begin{cases} M(i-1,j-1)+1 & if\ M(i)\ and\ M(j)\ form\ a\ basepair \\ 0 & otherwise \end{cases}$$

Stems of length greater than a mininal size *lmin* (value equal to 4 by default or set by the user) are searched for in the matrix. For example in Fig. 2 Left, two stems (surrounded by red) are selected.

The thermodynamic free energy $\triangle$G of the selected stems is then calculated. The ten stems with the best MFE (minimum free energy) are kept if they verify also a certain percentage of criteria.

When an exact stem is selected, it is used as an "anchor" for finding a symmetrical non-exact stem. The extension of the exact stem is done by considering only the diagonal containing the exact stem. The diagonal is browsed in Left and in Right of the "anchor", as long as the size of the internal loop is less than the size of the two stems delimiting it. For example, in Fig. 2 Left, only one exact stem has been extended.

Once the longest symmetrical non-exact stem is determined, several parameters are calculated: the total length, the number of exact stems composing it, the MFE, the length of the terminal loop, etc. The stem is selected if it checks a certain percentage of these parameters.

2.4.2 Hairpin Formation The hairpins are predicted from selected symmetrical non-exact stems. We consider that the selected symmetrical non-exact stem approximates well the pre-miRNA hairpin structure. The Center position of the stem (i.e. the middle position between the start and the end position) is then considered as the Center position of the possible hairpins. In order to find the structure of the hairpins, a non-symmetrical basepairing matrix is built from the two subsequences in Left and in Right of the Center position (Fig. 2 Right). Here, we consider also the Wooble basepairing, i.e. G-U.

The "anchor" of the considered symmetrical non-exact stem is positioned in the matrix, and then it is extended in Left and in Right, by browsing the diagonal containing the anchor, but also other close diagonals, in order to allow bulges and non-symmetrical internal loops (Fig. 2 Right).

The hairpins determined from the matrix must verify a set of criteria in order to be selected: length, free energy, size of the terminal loop, etc. Only hairpins where the percentage of verified criteria is higher than a certain percentage are selected.

2.4.3 Complexity of the algorithm The algorithm uses a sliding window of a given size L that slides of 10 nt over the considered sequence. The search of hairpins in the window is of time and memory complexity of $O(L^2)$. Therefore, the total time complexity of the algorithm is $O(L^2.N)$, where N is the sequence length. The memory complexity is of $O(L^2 + N)$.

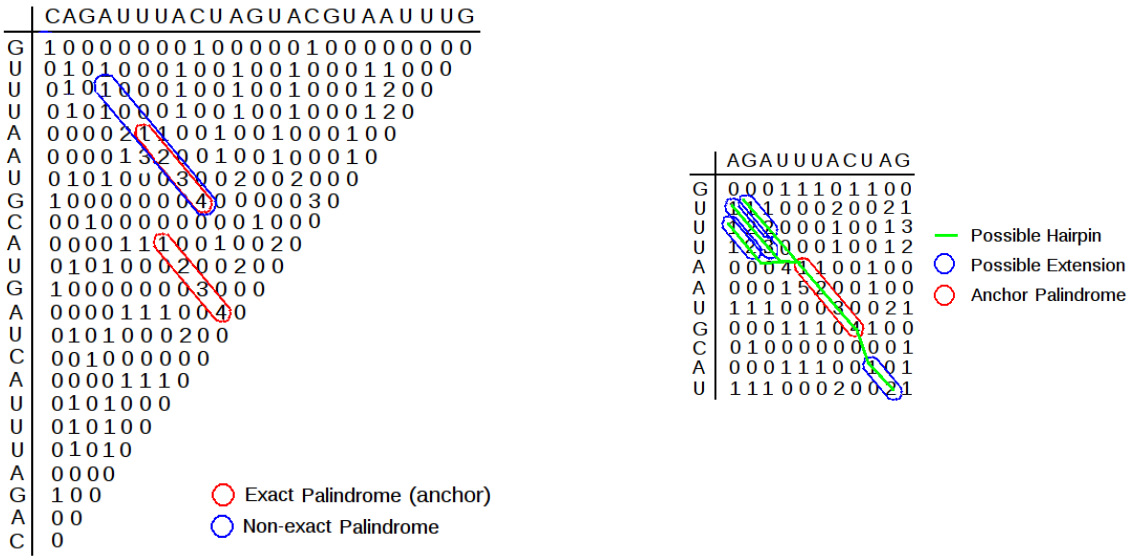

Figure 2. Left: An example of a symmetrical matrix for searching for exact and symmetrical non-exact stems. Two stems are selected with a threshold of minimum length equal to 4 (surrounded by a red circle). One of the two stems has been extended to a symmetrical non-exact stem (surrounded by a blue circle). **Right:** A non-symmetrical matrix for searching for hairpins, built from the longest symmetrical non-exact stem selected from the matrix of Right. The Center position of the possible hairpins is considered as the Center position of the symmetrical non-exact stem. The anchor of the symmetrical non-exact stem is positioned in the matrix (surrounded with red circle) and it is extended in Left and in Right on different diagonals, in order to allow bulges and internal loops. In this example, there are three possible extensions in Left and one in Right (surrounded with blue circle). Thus, there are three possible hairpins (green line).

3 Results

3.1 The data

To test our method, we considered two types of sequences: (i) an artificial sequence obtained by concatenating known pre-miRNAs with mRNA sequences, and (ii) real genomic sequences where a great number of pre-miRNAs are known.

3.1.1 Artificial sequence The artificial sequence was created by the concatenation of human mRNAs and the insertion of 100 human pre-miRNAs. The mRNA sequences came from the Human genome (build 37.1) of the NCBI Website (www.ncbi.nlm.nih.gov) and the pre-miRNAs came from miRBase database (release 16) (www.mirbase.org/). Pre-miRNA lengths are from 63 to 110 nt and the start position of pre-miRNAs begins every 300 nt, the first pre-miRNA starting at position 300. The obtained sequence in given in Supplementary Data 1.

3.1.2 Real genomic sequences We considered for our tests four genomes: human genome, mouse genome, zebrafish genome and sea squirt genome. We chose these genomes because they present a large cluster of known miRNAs:

- The human chromosome 19 (strand '+') has a cluster of 50 pre-miRNAs, the first pre-miRNA starting at position 54.169.933 and the last one ending at position 54.485.651.
- The mouse chromosome 2 (strand '+') has a cluster of 71 pre-miRNAs, the first one starting at position 10.388.290 and the last one ending at position 10.439.906.
- The zebrafish chromosome 4 (strand '-') has a cluster of 50 pre-miRNAs, the first one starting at position 34.353.975 and the last one ending at position 34.481.435.
- The sea squirt chromosome 7q (strand '-') has a cluster of 46 pre-miRNAs, the first one starting at position 5.400.066 and the last one ending at position 6.168.570.

For each of these genomes, we extracted from NCBI Website the sub-sequence that includes the considered pre-miRNA cluster.

3.2 Results and discussion

In order to evaluate our algorithm miRNAFold, we wanted to compare it with existing algorithms of same category, i.e. with ab-initio algorithms searching for pre-miRNA structures in genomes (whatever the species).

We considered RNALFold ([16]) which searches in genomic sequences for all possible non-coding RNA secondary structures including hairpins. We thus compared the hairpins predicted by RNALFold with the ones predicted by our algorithm miRNAFold. RNALFold uses a sliding window where structures that have a Minimum Free Energy (MFE) are selected, using the dynamic programming approach. Its time complexity is of $O(L^2.N)$ and its size complexity is of $O(L^2 + N)$.

We used RNALFold software in version 1.8.4. downloaded from the Vienna RNA Package (http://www.tbi.univie.ac.at/RN
and it was run with its default parameters.

A predicted hairpin does not always correspond exactly to the functional hairpin in cell. Therefore, we considered that a known pre-miRNA is correctly predicted if the returned position is correct. The position of an hairpin is considered as its Center and we assume that a predicted pre-miRNA corresponds to a known pre-miRNA if the distance between the known and the predicted Center is lower that 10% of the hairpin size.

For the two ab initio software RNALFold and miRNAFold, we used a sliding window of 150 nt.

3.2.1 Statistical measures In order evaluate and compare the two software, we used the measures of sensitivity and selectivity (specificity). The sensitivity measures the capability of the software to find known pre-miRNAs. The selectivity represents the probability that a predicted hairpin corresponds to a pre-miRNA. The sensitivity and the selectivity are given by the following equations:

$$Sensitivity = 100 \cdot \frac{TP}{TP + FN}$$

$$Selectivity = 100 \cdot \frac{TP}{TP + FP}$$

where TP (True Positives) is the number of known pre-miRNAs predicted, FN (False Negatives) is the number of known pre-miRNAs not predicted, and FP (False Positives) is the number of wrong pre-miRNAs predicted.

3.2.2 Results on the artificial sequence miRNAFold was run first on the artificial sequence, and its results were compared to those we obtained with RNALFold. An important parameter of miRNAFold is the minimal percentage of criteria that must be verified at each step of the algorithm (see section 2.3). We then tested miRNAFold with several percentage values: 30 %, 40%, 50%, 60%, 70% and 80%. The results of sensitivity and selectivity obtained by miRNAFold and RNALFold are given in Table 1.

	RNALFold	miRNAFold$_{30}$	miRNAFold$_{40}$	miRNAFold$_{50}$	miRNAFold$_{60}$	miRNAFold$_{70}$	miRNAFold$_{80}$
Sensitivity	97	**98**	**98**	**98**	97	97	94
Selectivity	4.53	8.21	8.21	8.95	11.3	11.6	**17**

Table 1. Comparison of prediction results obtained on an artificial sequence by miRNAFold and RNALFold. miRNAFold was run with different values for the parameter of minimal percentage of verified criteria: 30%, 40%, 50%, 60%, 70% and 80 %.

Because the artificial sequence contains 100 pre-miRNAs, the sensitivity values shown in Table 1 correspond to the number of pre-miRNAs correctly predicted (true positives).

RNALFold, miRNAFold$_{30}$, miRNAFold$_{40}$, miRNAFold$_{50}$, miRNAFold$_{60}$, miRNAFold$_{70}$ and miRNAFold$_{80}$ predicted a total of 2142, 1193, 1193, 1095, 858, 837 and 553 hairpins respectively (true positives and false positives).

miRNAFold$_{30}$, miRNAFold$_{40}$ and miRNAFold$_{50}$ predict one more true pre-miRNA than RNALFold, while they find two times less false pre-miRNAs. miRNAFold$_{60}$ and miRNAFold$_{70}$ find as many true pre-miRNAs as RNALFold, while they find in average three times less false pre-miRNAs. Finally, miRNAFold$_{80}$ finds less true pre-miRNAs than RNALFold but it finds almost four times less false pre-miRNAs.

As expected, the more the percentage parameter value considered in miRNAFold is low, the more the sensitivity is high. The more the percentage is high, the more the selectivity is high. When we increase this percentage from 30% to 80%, miRNAFold misses only 4 pre-miRNAs and removes more than 500 false pre-miRNAs. The thresholds allow the user, in a simple way, to choose between the discovery of a maximum number of pre-miRNAs with numerous false positives or the discovery of some pre-miRNAs with a lower number of false positives.

For the following, we decided to set to 70% the default value of this parameter, because it represents a good compromise between the sensitivity results and the selectivity results. This threshold value is thus set as the default.

3.2.3 Results on the real genomic sequences We tested miRNAFold and RNALFold on the four genomic sequences described above (human, mouse, zebrafish and sea squirt) which contain each a cluster of several known miRNAs. miRNAFold was run with a threshold of 70% for its parameter of minimum percentage of verified criteria. Table 2 shows the sensitivity and selectivity results obtained with miRNAFold and RNALFold in each of the considered sequences.

For the human sequence, miRNAFold and RNALFold succeeds both to predict all known pre-miRNAs. For the mouse sequence, our algorithm has a same prediction rate than RNALFold: 98.6%, which means that miRNAFold and RNALfold missed one known miRNA. For the zebrafish sequence, miRNAFold predicts all known pre-miRNAs, when RNALfold missed one known pre-miRNA. For the sea squirt sequence, miRNAFold and RNALFold give the same sensitivity rate (93,5%). Both have missed 3 known pre-miRNAs. Finally, we can say that miRNAFold give better or worse the same sensitivity results than RNALFold.

miRNAFold has in average a selectivity value two times better than the selectivity of RNALFold. This means that miRNAFold findstwo times less pre-miRNAs than RNALFold. For example, for the sea squirt sequence, RNALFold predicted 1832 hairpins when miRNAFold predicted only 738.

	Sensitivity				Selectivity			
	Human	Mouse	Zebrafish	Sea sq.	Human	Mouse	Zebrafish	Sea sq.
RNALFold	**100**	**98.59**	98.25	**93.48**	0.24	2.04	0.79	2.33
miRNAFold	**100**	**98.59**	**100**	**93.48**	**0.45**	**2.60**	**1.06**	**5.51**

Table 2. Sensitivity of RNALFold and miRNAFold in four human, mouse, zebrafish and sea squirt genomic sequences.

To summarize, miRNAFold has better sensitivity and selectivity results than RNALFold on the human, mouse, zebrafish and sea quirt genomic sequences.

These results confirm the results obtained on the human artificial data, where miRNAFold$_{70}$ has same sensitivity results than RNALFold but a better selectivity (2.5 times better). It is nevertheless important to mention that in case of the artificial sequence, the selectivity measure expresses exactly the capability of the algorithm to avoid false positive pre-miRNAs, since any predicted pre-miRNA which is not a known pre-miRNA is obligatory false. This is not the case of the selectivity calculated on the real sequences. A predicted not known pre-miRNA is indeed not obligatory false. It can be a real pre-miRNA which is not yet known.

3.3 Running time

With the increase of sequencing of large genomes, the running time is an important evaluation parameter of miRNA searching algorithms.

To compare the run time of miRNAFold and RNALFold, we considered the sub-sequences of 1 million of nucleotides beginning at positions 54.000.000, 10.000.000, 34.000.000 and 5.400.000 from the Human, Mouse, Zebrafish and Sea squirt genomes respectively, containing the clusters considered above.

Experiments were performed on a Linux machine equipped with Intel Core Duo 2 T6600 of 2.2 GHz and 4GB of RAM. The execution time of the two software on the four sequences is given in Table 3.

	Human	Mouse	Zebrafish	Sea squirt	Average
RNALFold	5m42s	5m45s	5m48s	5m48s	5m46s
miRNAFold	**1m3s**	**0m59s**	**0m55s**	**0m49s**	**57s**

Table 3. Run time of the algorithms RNALFold and miRNAFold for predicting pre-miRNAs in four genomic sequences of 1 million of nucleotides each. The values were rounded to the second. The last column shows the average execution time for a sequence of 1 million of nucleotides.

miRNAFold is the fastest algorithm. Our average time execution is 57 seconds for a sequence of 1 million of nucleotides, when RNALFold, the second fastest algorithm, has an average time execution of 5 minutes and 46 seconds. miRNAFold is then almost 6 times faster than RNALFold.

It should be noted that the times presented above in Table 3 do not take into account the time spent on RNALFold to remove the structures that are not hairpins.

4 Conclusion

We have presented here an original method called miRNAFold which allows a fast search for miRNA precursors in genomes. This method searches in a first step for the position of pre-miRNAs by approximating their structure, before deducing the final structure. The interest of the first step is the run time, since it searches for long exact stems that are then extended into long symmetrical non-exact stems (without bulges and non-symmetrical internal loops). The position of a selected symmetrical non-exact stem represents position of a possible pre-miRNA which structure is then predicted in a fast way. miRNAFold uses a sliding window, where all possible pre-miRNAs are searched for. The algorithm has a complexity of $O(L^2.N)$, where L in the length of the window, and N the size of the sequence.

miRNAFold was tested on several genomic sequences and on an artificial sequence, and was compared to RNALFold. miRNAFold gives better or similar sensitivity, and better selectivity. And most importantly, it is fast. On the tested sequences, it takes less than 1 minute for a sequence of 1 million length, when RNALFold takes more than 5 minutes.

The different criteria thresholds were defined from observations we done on miRBase hairpins. One of our further work is to develop automatic learning methods in order to define automatically these thresholds. Another further work is to optimise and adapt our code for using it on HPC solutions, and more precisely on GPU solutions, in order to make it much faster for whole genomes.Finally, we are working with biologists in order to use miRNAFold for finding new pre-miRNAs in genomes, and more precisely on the *Xenopus tropicalis* genome.

Acknowledgements

This work was funded by the Council of Essonne Region (Pôle System@tic, OpenGPU project). We would like to thank Mikael Trellet and Mederich Besnard, for developing the Webserver for miRNAFold.

References

[1] S. Agarwal and C. Vaz and A. Bhattacharya and A. Srinivasan, Prediction of novel precursor miRNAs using a context-sensitive hidden Markov model (CSHMM). *BMC Bioinformatics*, 11:S29, 2010.

[2] S.F. Altschul and R. Bundschuh and R. Olsen and T. Hwa, The estimation of statistical parameters for local alignment score distributions. *Nucleic Acids Res.*, 29:351-361, 2001.

[3] D. Bartel, MicroRNAs: genomics, biogenesis, mechanism and function. *Cell*, 116:281-197, 2004.

[4] S.H. Bernhart and I.L. Hofacker and P.F. Stadler, Local Base Pairing Probabilities in Large RNAs. *Bioinformatics*, 22:614-615, 2006.

[5] E. Bonnet and J. Wuyts and P. Rouze and Y. Van de Peer, Detection of 91 potential conserved plant microRNAs in Arabidopsis thaliana and Oryza sativa identifes important target genes. *P.N.A.S.*, 101:11511-11516, 2004.

[6] M. Brameier and C. Wiuf, Ab initio identification of human microRNAs based on structure motifs. *BMC Bioinformatics*, 8:478, 2007.

[7] S. Engelen and F. Tahi, Tfold: efficient in silico prediction of non-coding RNA secondary structures. *Nucleic Acids Res.*, 38:2453-2466, 2010.

[8] P.P. Gardner and R. Giegerich, A comprehensive comparison of comparative RNA structure prediction approaches. *Bioinformatics*, 5:140, 2004.

[9] S. Griffiths-Jones and H.K. Saini and S. van Dongen and A.J. EnRight, miRBase: tools for microRNA genomics. *Nucleic Acids Res.*, 36:D154-D158, 2008.

[10] N.R. Hansen, Statistical models for local occurrences of RNA structures. *J. Comput Biol*, 16:845-58, 2009.

[11] L. He and G. Hannon, microRNAs: small RNAs with a big role in gene regulation. *Nat. Rev. Genet.*, 5:522-531, 2004.

[12] S.A. Helvik and O.J. Snove and P. Saetrom, Reliable prediction of Drosha processing sites improves microRNA gene prediction. *Bioinformatics*, 23:142-149, 2007.

[13] J. Hertel and P.F. Stadler, Hairpins in a Haystack: recognizing microRNA precursors in comparative genomics data. *Bioinformatics*, 22:e197-e202, 2006.

[14] I.L. Hofacker and W. Fontana and P.F. Stadler and S. Bonhoeffer and M. Tacker and P. Schuster, Fast Folding and Comparison of RNA Secondary Structures.

[15] , . *Monatshefte f. Chemie*, 125:167-188, 1994.

[16] I.L. Hofacker and B. Priwitzer and P.F. Stadler, Prediction of Locally Stable RNA Secondary Structures for Genome-Wide Surveys. *Bioinformatics*, 20:186-190, 2004.

[17] C.H. Hsieh and D.T.H. Chang and C.H. Hsueh and C.Y. Wu and Y.J. Oyang, Predicting microRNA precursors with a generalized Gaussian components based density estimation algorithm. *BMC Bioinformatics*, 11:S52, 2010.

[18] T.H. Huang and B. Fan and M.F. Rothschild and Z.L. Hu and K. Li and S.H. Zhao, MiRFinder: an improved approach and software implementation for genome-wide fast microRNA precursor scans. *BMC Bioinformatics*, 8:341, 2007.

[19] P. Jiang and H. Wu and W. Wang and W. Ma and X. Sun and Z. Lu, MiPred: classification of real and pseudo microRNA precursors using random forest prediction model with combined features. *Nucleic Acids Res.*, 35:W339-344, 2007.

[20] H. Kiryu and T. Kin and K. Asai, Rfold: an exact algorithm for computing local base pairing probabilities. *Bioinformatics*, 24:367-73, 2008.

[21] E.C. Lai and P. Tomancak and R.W. Williams and G.M. Rubin, Computational identification of Drosophila microRNA genes. *Genome Biol.*, 4:R42, 2003.

[22] M. Legendre and A. Lambert and D. Gautheret, Profile-based detection of microRNA precursors in animal genomes. *Bioinformatics*, 21:841-845, 2005.

[23] L.P. Lim and N.C. Lau and E.G. Weinstein and A. Abdelhakim and S. Yekta and M.W. Rhoades and C.B. Burge and D.P. Bartel, The microRNAs of Caenorhabditis elegans. *Genes and Dev.*, 17:991, 2003.

[24] W. Ritchie and F.X. Théodule and D. Gautheret, Mireval: a web tool for simple microRNA prediction in genome sequences. *Bioinformatics*, 24:1394-6, 2008.

[25] A. Sewer and N. Paul and P. Landgraf and A. Aravin and S. Pfeffer and M.J. Brownstein and T. Tuschl and E. van Nimwegen and M. Zavolan, Identification of clustered microRNAs using an ab initio prediction method. *BMC Bioinformatics*, 6:267, 2005.

[26] G. Terai and T. Komori and K. Asai and T. Kin, miRRim: a novel system to find conserved miRNAs with high sensitivity and specificity. *RNA*, 13:2081-2090, 2007.

[27] S. Tyagi and C. Vaz and V. Gupta and R. Bhatia and S. Maheshwari and A. Srinivasan and A. Bhattacharya, CID-miRNA: A web server for prediction of novel miRNA precursors in human genome. *Biochemical and Biophysical Research Communications*, 372:831-834, 2008.

[28] X. Wang and J. Zhang and F. Li and J. Gu and T. He and X. Zhang and Y. Li, microRNA identification based on sequence and structure alignment. *Bioinformatics*, 21:3610-3614, 2005.

[29] Y. Wang and H.M. Stricker and D. Gou and L. Liu, microRNA: past and present. *Front Biosci.*, 12:2316-2329, 2007.

[30] C. Xue and F. Li and T. He and G-P. Liu and Y. Li and X. Zhang, Classification of real and pseudo microRNA precursors using local structure-sequence features and support vector machine. *BMC Bioinformatics*, 6:310, 2005.

[31] M. Yousef and M. Nebozhyn and H. Shatkay and S. Kanterakis and L.C. Showe and M.K. Showe, Combining multi-species genomic data for microRNA identification using a Naïve Bayes classifier. *Bioinformatics*, 22:1325-1334, 2006.