BIOINFORMATION
Discovery at the interface of physical and biological sciences

BIOMEDICAL INFORMATICS

Open access
www.bioinformation.net
Volume 12(1)
Prediction model

# A method for clustering of miRNA sequences using fragmented programming

**Anatoly Ivashchenko*, Anna Pyrkova & Raigul Niyazova**

Computer Science Laboratory, Al-Farabi Kazakh National University, Almaty - 050038, Kazakhstan, Anatoly Ivashchenko – Email: a_ivashchenko@mail.ru, *Corresponding author

Received January 05, 2016; Revised January 07 2016; Accepted January 07, 2016; Published January 31, 2016

**Abstract:**
Clustering of miRNA sequences is an important problem in molecular genetics associated cellular biology. Thousands of such sequences are known today through advancement in sophisticated molecular tools, sequencing techniques, computational resources and rule based mathematical models. Analysis of such large-scale miRNA sequences for inferring patterns towards deducing cellular function is a great challenge in modern molecular biology. Therefore, it is of interest to develop mathematical models specific for miRNA sequences. The process is to group (cluster) such miRNA sequences using well-defined known features. We describe a method for clustering of miRNA sequences using fragmented programming. Subsequently, we illustrated the utility of the model using a dendrogram (a tree diagram) for publically known *A. thaliana* miRNA nucleotide sequences towards the inference of observed conserved patterns.

**Background:**
The human genome is known to contain thousands of miRNAs. More than 3000 new miRNAs with sequences have been recently identified **[1-3]**. Increasing numbers of such new miRNAs will be identified leading to a problem for affiliating these with known families for finding new families. The division of miRNAs into families does not adequately reflect the degree of nucleotide sequence similarity, and the categorization of miRNAs into families requires quantitative criteria defining the differences between families. The genomes of different organisms have orthologous miRNAs that should be distributed into families. Hence, it is necessary to establish the degree of similarity for orthologous miRNAs and their belonging to different families. Several authors propose different functional clustering methods for this purpose **[4-6].** Therefore, it is of interest to describe a method for clustering miRNAs sequences using fragmented programming.

## Model Illustration

Let $\{u_l\}$ is a set of nucleotide sequences necessary for comparison and clustering, where, $l = \overline{1, N}$ , then

$\langle u_l, u_k, S\rangle_{l,k} = \overline{1, N}\ , l \neq k$ } This is clustering of sequences. } where,

$S = \{u_l, u_k, \%relation\},\ k = 1, N$ } This is the distance between sequences [$u_l$, $u_k$]. } The clustering weight is defined as a function of nucleotide comparison scales.

**Figure 1:** Illustration of a mathematical model for miRNA clustering.

**Methodology:**

*Model for clustering miRNA nucleotide sequences*

Clustering nucleotide sequences is a process of sequence comparison with the definition of maximum number of nucleotide coincidences. This is useful for constructing a graphical structure like a tree defining relationship between sequences. The formulated model for a sequence based clustering problem is illustrated in **Figure 1.**

*Algorithm:*

*Clustering of miRNA sequences for phylogenetic tree*

The main issue with large range nucleotide sequences is lack of sufficient computing power. We describe a fragmented algorithm **(Figure 2)** for clustering miRNA nucleotide sequences in 5 steps using a flowchart (**Figure 3**).

*Dataset for model testing*

A dataset of known miRNA nucleotide sequences from *A. thaliana* (**Table 1)** was tested using this method.

**Table 1:** Splitting miRNA sequences (the number of the processed sequences represents 3700 miRNA) into families for defining the degree of their relationship (in %).

| miRNA name | miRNA sequence |
|---|---|
| leU-7f | UGAGGUAGUAGAUUGUAUAGUU |
| leU-7f-1* | CUAUACAAUCUAUUGCCUUCCC |
| leU-7f-2* | CUAUACAGUCUACUGUCUUUCC |
| leU-7g | UGAGGUAGUAGUUUGUACAGUU |
| leU-7g* | CUGUACAGGCCACUGCCUUGC |
| miR-101 | UACAGUACUGUGAUAACUGAA |
| miR-101* | CAGUUAUCACAGUGCUGAUGCU |
| miR-103 | AGCAGCAUUGUACAGGGCUAUGA |
| miR-103-2* | AGCUUCUUUACAGUGCUGCCUUG |
| miR-103-as | UCAUAGCCCUGUACAAUGCUGCU |
| miR-105 | UCAAAUGCUCAGACUCCUGUGGU |
| miR-105* | ACGGAUGUUUGAGCAUGUGCUA |
| miR-106b | UAAAGUGCUGACAGUGCAGAU |
| miR-106b* | CCGCACUGUGGGUACUUGCUGC |
| miR-107 | AGCAGCAUUGUACAGGGCUAUCA |
| miR-1180 | UUUCCGGCUCGCGUGGGUGUGU |
| … … … | … … … |

## Model algorithm

## Splitting a set of nucleotide sequences

Step 1

$\{u_l\}, \quad l = \overline{1, N}$ This needs to be compared into $M$ uniform groups.

$u_l, \ldots, u_{l_2-1}\}, \; u_{l_2}, \ldots, u_{l_3-1}, \ldots, \; u_{l_M}, \ldots, u_N$ } N is the number of sequences.

## Processing of sequences

Step 2

Each of the *M* groups of sequences is processed by a corresponding procedure. This leads to total sequences for each group.

$u_{l_k}, \ldots, u_{l_k-1} - U_k, \quad k = \overline{1, M}$ } M is the number of fragments.

## Iteration to first step

Step 3

Each of $M$ groups is sent to the first step for clustering of sequences $u_{l_k}, \ldots, u_{l_k-1}$ and total sequence $\overline{u_k}$

## Clustering of sequence groups

Step 4

Cluster a group of sequences (total) $\{\overline{u_k}\}$ $k = \overline{1, M}$ by the 1st process. } This is compared with already clustered $M$ groups $\{u_1, \ldots, u_{l2-1}\}, u_{lM}, \ldots, u_N$ by N processes. N is the number of sequences.

## Dendrogram (Tree diagram)

Step 5

A dendrogram was drawn with Neighbourhood Joining and UPGMA algorithms for the resultant clustered sequences.

**Figure 2:** Illustration of an algorithm for miRNA clustering.

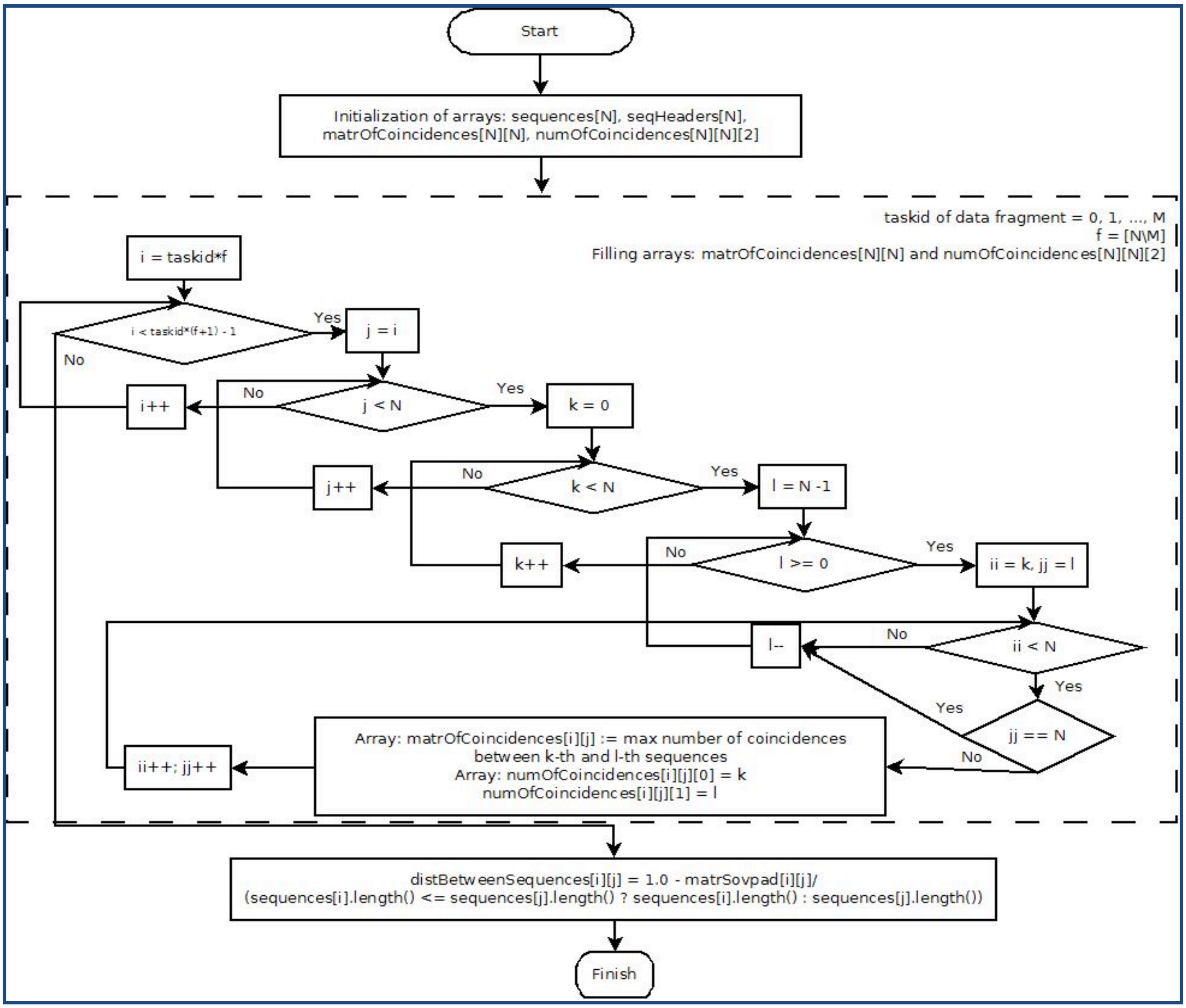
Start
Initialization of arrays: sequences[N], seqHeaders[N], matrOfCoincidences[N][N], numOfCoincidences[N][N][2]
taskid of data fragment = 0, 1, ..., M
f = [N\M]
Filling arrays: matrOfCoincidences[N][N] and numOfCoincidences[N][N][2]
i = taskid*f
i < taskid*(f+1) - 1
Yes
No
j = i
i++
No
j < N
Yes
k = 0
j++
No
k < N
Yes
l = N -1
k++
No
l >= 0
Yes
ii = k, jj = l
l--
No
ii < N
Yes
Yes
jj == N
No
Array: matrOfCoincidences[i][j] := max number of coincidences between k-th and l-th sequences
Array: numOfCoincidences[i][j][0] = k
numOfCoincidences[i][j][1] = l
ii++; jj++
distBetweenSequences[i][j] = 1.0 - matrSovpad[i][j]/ (sequences[i].length() <= sequences[j].length() ? sequences[i].length() : sequences[j].length())
Finish

**Figure 3:** Flowchart for miRNA clustering using fragmented algorithm. This figure shows data processing by each block of the fragmented algorithm. There is filling matrixes matrOfCoincidences and numOfCoincidences where each element of the first matrix is a maximum number of nucleotide coincidences in the corresponding positions between two sequences. Each couple of elements of the second matrix keeps numbers for two compared sequences. In distBetweenSequences matrix the measures of coincidences between each couple of sequences are saved in percentage ratio.

**Results and Discussion:**

A fragmented algorithm for miRNA nucleotide sequence clustering and a program application were developed to define the degree of relationship between sequences according to their clustering. This helps to create phylogenetic trees based on Neighbourhood-Joining (NJ) and UPGMA algorithms in this approach after clustering known miRNA sequences (**Figure 4**).

Many programs are available for searching related sequences in databases. This is useful for creating multiple alignments for generating phylogenetic trees. Tools used in such analysis include BLAST, ClustalW, ClustalX, UGENE and many others. The main issue here is lack of sufficient computing resources for large-scale analysis.

The method described here using fragmented programming optimised the time required for data processing during clustering. This achieved better clustering results by dividing the set of sequences into $M$ independent groups (fragments) processed by $M$ blocks, each of which will undergo fragment clustering irrespective of other fragments. The overall clustering is performed for all sequences in each group.

A clustering process occurs simultaneously in all groups where independent processing is possible for all processed data in fragmented programming. Merging all related sequences in a fragment forms a cluster as clustering is completed in each block for every group. The main block in the algorithm finishes a clustering by merging all received $M$ clusters on the basis of their clustered relationship. Matrixes of related sequences were broken into clusters of related nucleotide sequences and processed by $M$ independent blocks by the fragmented programming algorithm.

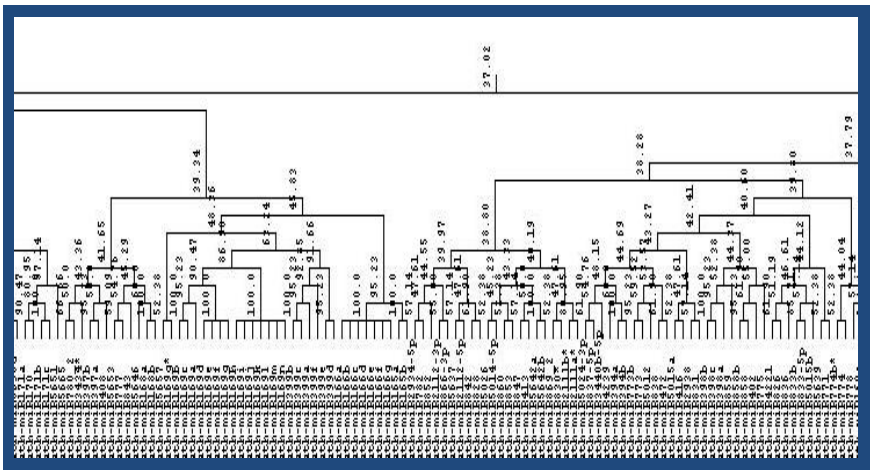

**Figure 4:** Dendrogram of miRNA sequences from *A. thaliana* is shown as a use of the model. This identified a conserved polynucleotide, which serves as the Ath-miR156a-j binding site in paralogous SPL mRNAs. These nucleotides encode the conserved ALSLLS motif and the miR156 and miR157 subfamilies belonging to the same family **[7]**. The HAM1, HAM2, HAM3 paralogous mRNA binding sites for ath-miR171a-c and ath-miR170 are located in the coding DNA sequence and are conserved in the mRNAs of 39 orthologous genes within 13 species. The human miR-1273 family includes miRNAs with different nucleotide sequences; therefore in different miRNAs families **[8]**.

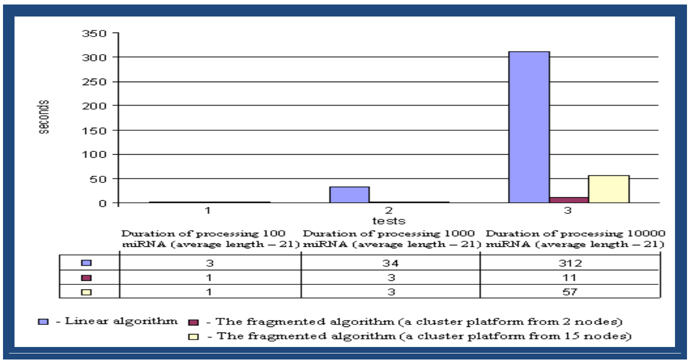

**Figure 5**: Comparative analysis of the linear and fragmented algorithms for clustering miRNA nucleotide sequences while constructing a dendrogram. It should be noted that time is specified in seconds.

The advantages of fragmented programming are the feasibility for automatic (1) parallel computing, (2) dynamic properties, (3) calculation of multiple architectures, and (4) subsequent analysis of parallel computing. The fragmented algorithm requires a minimum management determined by data dependency and is not dependent on the distribution of resources. Thus, it assumes a set of ways for process execution that provides portability. A problem of executive system is to execute display of objects in an algorithm (variables, operations) on resources to a concrete computing system. This automatically provides all necessary dynamic properties for parallel computing. Fragmentation is a processing method for reducing the number of objects in the algorithm. This simplifies a problem of creation for effective distribution of resources and management.

The resultant sub trees were united into one phylogenetic tree **(Figure 4)** by the main block in the algorithm. Its computing complexity makes ***O(nlk)*** operations where *k* is the number of clusters, *n* is the size of a dataset and *l* is the quantity of cycles in the algorithm (**Figure 5**).

## Conclusion:

We describe a method for clustering of miRNA sequences using fragmented programming. The method creates sequence clusters as input to NJ and UPGMA for generating phylogeny related tree diagrams. We used known *A. thaliana* miRNA nucleotide sequences and developed clusters using this method for generating a sample dendrogram to illustrate the utility of the model.

## References:

**[1]** Londina E *et al. Proc Natl Acad Sci* USA. 2015 **112**: E1106 [PMID: 25713380]

**[2]** Wan L *et al. BMC Genomics*. 2012 **13**: S15 [PMID: 23282099]

**[3]** Kaczkowski B *et al. Bioinformatics*. 2009 **25**: 291 [PMID: 19059941]

**[4]** Dib L & Carbone A, *BMC Bioinformatics*. 2012 **13:** 194 [PMID: 23216858]

**[5]** Nerini-Molteni S *et al. Curr Med Chem*. 2012 **19**: 6214 [PMID: 22664252]

**[6]** ElSharawy A *et al. Aging Cell*. 2012 **11:** 607 [PMID: 22533606]

**[7]** Bari A *et al. Biomed Res Int*. 2013 **2013**: 307145 [PMID: 23936788]

**[8]** Ivashchenko A *et al. Biomed Res Int*. 2014 **2014:** 620530 [PMID: 2524316]

**Edited by P. Kangueane**

**Citation**: **Ivashchenko *et al.*** Bioinformation 12(1): 15-18 (2016)

**License statement**: This is an Open Access article which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly credited. This is distributed under the terms of the Creative Commons Attribution License.