BIOINFORMATICS ORIGINAL PAPER

Vol. 28 no. 1 2012, pages 17–24
doi:10.1093/bioinformatics/btr598

Sequence analysis

Advance Access publication November 3, 2011

deepBlockAlign: a tool for aligning RNA-seq profiles of read block patterns

David Langenberger[1,2,†], Sachin Pundhir[3,4,†], Claus T. Ekstrøm[5], Peter F. Stadler[1,2,3,6,7,8,9], Steve Hoffmann[1,2] and Jan Gorodkin[3,4,*]

[1]Bioinformatics Group, Department of Computer Science, Interdisciplinary Center for Bioinformatics, [2]Transcriptome Bioinformatics group, LIFE - Leipzig Research Center for Civilization Diseases, Universität Leipzig, Philipp-Rosenthal-Strasse 27, D-04107 Leipzig, Germany, [3]Center for non-coding RNA in Technology and Health, Department of Basic Animal and Veterinary Sciences, University of Copenhagen, 1870 Frederiksberg C, Denmark, [4]Center for Applied Bioinformatics, Department of Basic Animal and Veterinary Sciences, University of Copenhagen, 1870 Frederiksberg C, Denmark, [5]Center for Applied Bioinformatics, Department of Basic Sciences and Environment, University of Copenhagen, 1871 Frederiksberg C, Denmark, [6]Max Planck Institute for Mathematics in the Sciences, Inselstraße 22, [7]RNomics Group, Fraunhofer Institut for Cell Therapy and Immunology, Perlickstraße 1, D-04103 Leipzig, Germany, [8]Institute for Theoretical Chemistry, University of Vienna, Währingerstraße 17, A-1090 Vienna, Austria and [9]The Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM, USA

Associate Editor: Ivo Hofacker

ABSTRACT

Motivation: High-throughput sequencing methods allow whole transcriptomes to be sequenced fast and cost-effectively. Short RNA sequencing provides not only quantitative expression data but also an opportunity to identify novel coding and non-coding RNAs. Many long transcripts undergo post-transcriptional processing that generates short RNA sequence fragments. Mapped back to a reference genome, they form distinctive patterns that convey information on both the structure of the parent transcript and the modalities of its processing. The miR-miR* pattern from microRNA precursors is the best-known, but by no means singular, example.

Results: `deepBlockAlign` introduces a two-step approach to align RNA-seq read patterns with the aim of quickly identifying RNAs that share similar processing footprints. Overlapping mapped reads are first merged to blocks and then closely spaced blocks are combined to block groups, each representing a locus of expression. In order to compare block groups, the constituent blocks are first compared using a modified sequence alignment algorithm to determine similarity scores for pairs of blocks. In the second stage, block patterns are compared by means of a modified Sankoff algorithm that takes both block similarities and similarities of pattern of distances within the block groups into account. Hierarchical clustering of block groups clearly separates most miRNA and tRNA, and also identifies about a dozen tRNAs clustering together with miRNA. Most of these putative Dicer-processed tRNAs, including eight cases reported to generate products with miRNA-like features in literature, exhibit read blocks distinguished by precise start position of reads.

Availability: The program `deepBlockAlign` is available as source code from http://rth.dk/resources/dba/.

Contact: gorodkin@rth.dk; studla@bioinf.uni-leipzig.de

*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on July 16, 2011; revised on October 21, 2011; accepted on October 25, 2011

1 INTRODUCTION

Recent development in high-throughput sequencing (HTS) technologies have made the demand for efficient algorithms for data processing more urgent than ever. Ironically, while the sequencing costs decrease, the analysis costs increase and consume the bigger part of sequencing projects. Contributing to the demand is the novel possibilities which emerge with these data. Questions that need to be addressed range from expression analysis to the reconstruction of transcript structures and the recognition of particular classes of coding and non-coding transcripts. In most settings, a reference genome is available and analysis protocols start with mapping the sequencing reads to that template genome (Hoffmann *et al.*, 2009; Langmead *et al.*, 2009; Trapnell *et al.*, 2009). Here, we focus in particular on the type of RNA sequencing data that is commonly produced in studies focusing on microRNAs. A series of publications reported that microRNA-sized small RNAs are commonly produced not only from microRNA precursors, but also from most other classes of structured RNAs (Kawaji *et al.*, 2008; Taft *et al.*, 2009). These small RNAs are often, but not always, produced by Dicer (Brameier *et al.*, 2011; Burroughs *et al.*, 2011; Cole *et al.*, 2009; Haussecker *et al.*, 2010; Lee *et al.*, 2009). Several alternative, Dicer-independent pathways that lead to similar small RNAs with microRNA-like functions have been characterized, see Miyoshi *et al.* (2010) for a recent review.

The apparent diversity of processing pathways bears the question to what extent the read patterns in RNA-seq datasets contain information on the processing of particular RNAs. Well-understood examples include the characteristic mutual positioning with a 3′-overhang of miR and miR* products that is characteristic

for Dicer cleavage, see e.g. Gan *et al.* (2008), the anomalous 5′-overhang observed for some microRNAs resulting from a distinct, Dicer-dependent two-step mechanism (Ando *et al.*, 2011), and the Dicer-independent processing of mir-451 (Cifuentes *et al.*, 2010). Therefore, we ask whether it is possible in general to develop 'finger prints' for distinct pathways.

Several recent studies recognized that structured ncRNAs such as tRNAs and snoRNAs give rise to characteristic patterns of read coverage that in many cases are dominated by distinctive clusters of reads with similar start and/or stop position. These clusters are referred to as blocks. In the case of tRNAs, the patterns are influenced in particular by chemical modifications (Findeiß *et al.*, 2011), while in other cases secondary structures play a major role (Langenberger *et al.*, 2010). As a consequence, these patterns convey information about the parent RNAs. Machine learning algorithms have been trained on the combination of relative expression and distances between read blocks to distinguish major ncRNAs classes such as pre-microRNAs, box C/D and box H/ACA snoRNAs, and tRNAs (Langenberger *et al.*, 2010). Similarly, Jung *et al.* (2010) showed that ncRNA classes can also be distinguished by comparing accumulations of reads, i.e. by number of reads and the size of the clusters of overlapping reads. The ALPS scores (Erhard and Zimmer, 2010), which are based on the relative position and the read lengths only, are also capable of discriminating between major types of ncRNAs. Finally, short read patterns in combination with predicted secondary structures and sequence conservation have been used to identify genomic loci with high potential to encode for ncRNAs (Lu *et al.*, 2011). The latter work suggests that even further data, such as high-throughput RNA structure probing experiments (Underwood *et al.*, 2010), could be used together with short read block patterns to complement computational methods for ncRNA gene finding [reviewed by Gorodkin and Hofacker (2011); Gorodkin *et al.* (2010)].

Beyond the primary goal of distinguishing different ncRNAs, it is of particular interest to identify common patterns on different transcripts. Establishing methods for pairwise comparison and subsequent clustering is an important step toward this goal. This allows us to find common patterns for the same class of RNAs, to the detection of putative novel classes of RNAs, and to commonalities among different ncRNAs that share (parts of) processing pathways. The ability to compare read patterns, both at the level of individual read blocks and at the level of block groups independent of sequence and secondary structure data is a necessary prerequisite to disentangle the different influences. Here, we develop the necessary algorithms and provide the `deepBlockAlign` software package that implements these tools for practical use.

2 MATERIALS AND METHODS

The starting point for `deepBlockAlign` is a collection of reads mapped to a (reference) genome. Clusters of overlapping reads are decomposed into blocks of reads with similar start and stop positions using `blockbuster` (Langenberger *et al.*, 2009). Both the length and the coverage profile can vary substantially between blocks. In the following, we introduce an entropy-like measure for the coherence of read blocks. Overlapping and closely spaced blocks of reads form a block group or locus. Our aim is to compare these block groups based on the relative expression of blocks, the distance between blocks and the shapes of the blocks themselves.

`deepBlockAlign` proceeds in two stages. First, an alignment algorithm is employed to compare the coverage profiles of individual blocks, thus computing a similarity score between the blocks. In the second stage, we compare the arrangements of blocks within block groups with each other. Using this procedure, we conduct a clustering to group similar RNAs and to identify if different RNAs share common patterns. This also open up the possibility of discovering entirely new processing patterns. The output will point to cases which need further manual inspection.

Table 1. The HTS dataset used in this study along with possible ID from GEO, the number of reads and number of block groups

Dataset	GEO ID	No. of reads	No. of block groups[a]	
			All	Expression filter
Human_eb[b]	–	7 351 304	1136	455
Human_hesc[c]	–	7 836 912	1386	585
Human_34[d]	GSM450598	7 299 034	1103	377
Human_98[e]	GSM450608	8 371 772	1109	425
Human_14[f]	GSM450605	8 538 940	1614	686
Monkey_9[g]	GSM450615	10 698 419	1738	478

The expression filter requires a block group to have at least two blocks with a minimum of 50 reads. Furthermore, block groups >200 nt or <50 nt are excluded.
[a]Block groups with >1 blocks, >50 nt and <200 nt in length.
[b]Human embryoid cells (Morin *et al.*, 2008).
[c]Human embryonic stem cells (Morin *et al.*, 2008).
[d]Human brain (34 days) (Somel *et al.*, 2010).
[e]Human brain (98 years) (Somel *et al.*, 2010).
[f]Human brain (14 years) (Somel *et al.*, 2010).
[g]Monkey brain (9 years) (Somel *et al.*, 2010).

2.1 Data and their preprocessing

In order to construct a set of benchmark data for `deepBlockAlign`, we downloaded previously published Illumina sequencing datasets shown in Table 1. The human (hg18, Mar. 2006) and rhesus macaque (rheMac2, January 2006) genome assemblies, obtained from the UCSC genome browser (Hinrichs *et al.*, 2006), served as respective references for short read mapping using `segemehl` (Hoffmann *et al.*, 2009) with default parameters. The `segemehl` software detects mismatches and indels and reports multiple hits with optimal score. The read data was normalized by the number of hits for each read. This procedure ensures that the redundancy of multiple (nearly) identical copies (e.g. of tRNAs) is properly taken into account. To account for sequencing errors and ncRNA editing effects (Findeiß *et al.*, 2011), we required a minimum mapping accuracy of 85%. To locate distinct accumulations of reads (putative ncRNAs), we assigned two reads to the same locus, when they were separated by <30 nt. Then, to detect specific expression patterns, we divided consecutive reads within these loci into blocks using `blockbuster` (with parameters: `-distance 30, -minBlockHeight 1, -minClusterHeight 50, -scale 0.5`) (Langenberger *et al.*, 2009). `blockbuster` merges mapped reads into blocks based on their location in the reference genome. Thus, stacks of reads are combined to *read blocks*. This strategy greatly reduces the size of the dataset and allows the application of more costly algorithms while maintaining structural properties such as position, length and approximate read start sites and ends. The obtained loci are then called *block groups*. We obtained 455 block groups from the Human_eb dataset with more than one block, at least 50 reads and the size range between 50 nt and 200 nt. This dataset has been used for benchmarking throughout the study.

These 455 blocks were then compared to known annotation [1049 microRNA loci from miRBase v16, Kozomara and Griffiths-Jones (2011); 513 tRNA loci from gtRNAdb, Chan and Lowe (2009); 402 snoRNA loci as well as 4524 other RNAs from UCSC annotation; Karolchik *et al.* (2004)]. The benchmark set contains 193 microRNAs, 47 snoRNAs, 157 tRNAs, 40 other annotated ncRNAs and 18 unannotated RNAs. In line with previous work (Langenberger *et al.*, 2010), we observe that different ncRNAs give

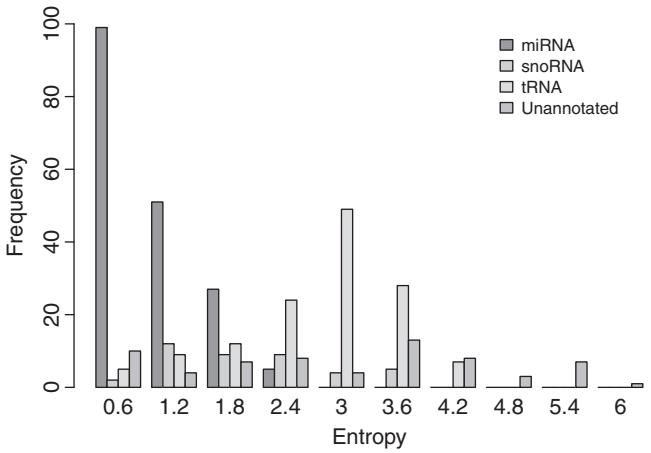

Fig. 1. Entropy of distinct starting positions for different classes of ncRNA of our 455 block groups in Human_eb dataset. The different profiles suggest that the entropy is a distinct measure for each ncRNA type and could be used for separation.

rise to distinct block patterns that are distinguished by characteristic features such as the number of blocks, the lengths of blocks, the distances between consecutive blocks and the relative expression of the blocks.

2.2 Read pattern within a block group

In order to characterize the read distribution within a block group, we measured the entropy of the start positions. Let q_i denote the fraction of reads in a given block group that starts at position i. We consider the entropy

$$I = -\sum_i q_i \log_2 q_i \tag{1}$$

The sum run over all possible positions of read starts within the block group. Small values of I indicate well-defined block patterns, and hence are indicative of specific processing, while large values arise from blurred patterns and suggest random degradation.

All the ncRNA classes, e.g. microRNAs, tRNAs and snoRNAs show varying degrees of diversity (distribution of start positions in the block group), which is reflected in varying entropy distributions as shown in Figure 1. This suggests that the entropy is a characteristic measure for each ncRNA type and indicates to which degree the different families can be separated. It also indicates that this to some extent can be used in the effort to separate different ncRNA classes.

Not surprisingly, we observe a moderate correlation ($r = 0.41$) between entropy and the length of a block group, as the length itself is also an important parameter, when aligning read blocks. For comparison of length and entropy, see Supplementary Figure S1.

2.3 Alignment strategy

The purpose of `deepBlockAlign` is the comparison of the read mapping patterns of two block groups obtained from short RNA-seq experiments. To this end, it employs a two-tiered alignment strategy. In the first step, individual blocks of reads are compared with each other. This is motivated by the observation that start and end patterns, and hence also entropies, may differ substantially between individual blocks of reads. A pairwise alignment algorithm similar to the Needleman–Wunsch algorithm for sequence data (Needleman and Wunsch, 1970) is used to compute an optimal alignment and a similarity score from the normalized frequency of reads covering each position of the two input blocks.

Block groups are then compared using an alignment approach. Here, a similarity measure is used that combines the similarity scores of the individual blocks and differences in the distances between aligned blocks. Algorithmically, a variant of the Sankoff (1985) algorithm is used.

2.4 Alignment of read blocks

Given a deep sequencing experiment, each position i of the reference genome is in essence associated with two measurements: the number of reads covering position i, x_{1i}, and the number of reads starting at position i, x_{2i}. The read profile $\vec{X}$ of a block can thus be thought of as a sequence of pairs $\vec{X}_i = (x_{1i}, x_{2i})$. The differences between the read mapping profiles $\vec{X}$ and $\vec{Y}$ of two blocks can be expressed in terms of a position-wise dissimilarity score $\alpha|x_{1i} - y_{1j}| + \beta|x_{2i} - y_{2j}|$, where α and β set relative weights for the influence of read starts and read coverage. We introduce affine gap cost with C_i (initiation) and C_e (elongation) to minimize the amount of indels, assuming this is reflected as a minimization of the number of different processing events. The optimal alignment of the read blocks $\vec{X}$ and $\vec{Y}$ is obtained with the help of the familiar Needleman–Wunsch algorithm. This simple idea, however, needs a few refinements to become applicable in practise.

First, it appears natural to work with normalized read counts to capture similar shapes at different expression levels. Furthermore, we found it useful to focus on the normalized difference

$$x_i = \left(x_{1i} - x_{2i}\right)/N_X \tag{2}$$

of read coverage and start reads across the block $\vec{X}$, where N_X is the total number of reads in the block group having block X. We have normalized in order to make a meaningful comparison regardless of the absolute expression level (number of reads). A version of the algorithm could be made without normalization. Finally, we disregard differences in similarity whenever two blocks are so dissimilar that they appear entirely unrelated. This leads us to a similarity measure of the form

$$\Psi_\delta^\pm(i,j) = \begin{cases} S_0 \cdot [1 - (\epsilon(i,j) + \eta^\pm(i,j))] & \text{if } |x_i - y_j| < \delta \\ S_1 \cdot [\epsilon(i,j) + \eta^\pm(i,j)] & \text{otherwise} \end{cases}, \tag{3}$$

where δ is the threshold up to which we consider x_i and y_j as related. A $+$ ($-$ respectively) on the r.h.s. on the equation corresponds to a $+$ ($-$ respectively) on the l.h.s. of the equation. The parameters S_0 and S_1 are the weights associated with match and mismatch, respectively. Note that when $\delta = 1$ the 'otherwise' case is never entered. However, for large differences between x_i and y_j the first case can be negative and will in those cases correspond to a 'mismatch' score. The function

$$\epsilon(i,j) = |x_i - y_j| / \max\{x_i, y_j\} \tag{4}$$

penalizes the match score, as the expression difference between two blocks increases. The second term, $\eta^\pm$, measures the relative difference of normalized read count difference at consecutive positions. Provided the previous positions, $i-1$ and $j-1$ have the same read count difference as the present positions, i and j, we set

$$\eta^+(i,j) = \zeta \cdot \frac{\left||x_i - y_j| - |x_{i-1} - y_{j-1}|\right|}{\max\{|x_i - y_j|, |x_{i-1} - y_{j-1}|\}}, \tag{5}$$

otherwise we use $\eta^-(i,j) = 0$. The functions ϵ and η tune the match and mismatch scores according to the difference in expression and shape of the two read blocks, respectively. ζ is a parameter tuning the relative importance of η, and hence of the variation between adjacent positions.

Let $D_{i,j}$ and $E_{i,j}$ denote the optimal score of a subalignment ending in a deletion $(x_i, -)$ and an insertion $(-, y_j)$, respectively, and $M_{i,j}$ denote the optimal score of a subalignment ending in a substitution (x_i, y_j), i.e. a match or mismatch. We furthermore define

$$S_{i,j} = \max\{M_{i,j}, D_{i,j}, E_{i,j}\}. \tag{6}$$

These scores satisfy the recursions

$$M_{i,j} = \max \begin{cases} M_{i-1,j-1} + \Psi_\delta^+(i,j) \\ D_{i-1,j-1} + \Psi_\delta^-(i,j) \\ E_{i-1,j-1} + \Psi_\delta^-(i,j) \end{cases},$$

$$D_{i,j} = \max \begin{cases} S_{i,j-1} + C_i \\ D_{i,j-1} + C_e \end{cases},$$

$$E_{i,j} = \max \begin{cases} S_{i-1,j} + C_i \\ E_{i-1,j} + C_e \end{cases},$$

Note that gap states only implicitly depend on the M states as these only keep track of matches/mismatches from positions $i-1$ and $j-1$. The score of the global alignment, $S = S_{|x|,|y|}$, measures the similarity of the two blocks. The algorithm is easily modified for local alignment of read patterns by including the beginning of a new local alignment (with score 0) in the recursion (6), analogous to the Smith–Waterman sequence alignment algorithm. An alternative implementation would be to let the score depend explicitly on previous positions by using double substitutions (Akbasli, 2007; Crooks *et al.*, 2005). By trial-and-error, we readily found the following parameter values $S_0 = 1, S_1 = -1, C_i = -2, C_e = -1, \delta = 1$ and $\zeta = 1$, which worked well and hence were used in all the subsequent analyses. It should be mentioned that the value of $\delta = 1$ makes the second condition of Equation (3) redundant. Other parameter values (with smaller δ) give comparable results. We tested a range of values for δ and found that values of $\delta \geq 0.05$ largely give the same results (data not shown). An example of aligning the profiles from two blocks is shown in Figure 2a.

2.5 Alignment of block groups

The comparison of block groups is based both on the similarities of individual blocks and on the similarities of distances between pairs of blocks. As for other problems e.g. the Maximum Contact Map Overlap Problem (Caprara *et al.*, 2004), this is in general a hard problem, which could be solved by an ILP approach or using stochastic heuristics. We notice, however, that the emphasis on pairs is reminiscent of the problems of simultaneous computation of an alignment and a secondary structure, which is solvable in polynomial time by the Sankoff algorithm (Sankoff, 1985). The basic idea is that the distances between a collection of blocks on a genome are already determined by a small subset of all distances, so that a collection of nested pairs of blocks already can be expected to contain most of the distance constraints.

Consider two block groups denoted by a sequence of blocks $\mathcal{C} = C_1 \ldots C_n$ and $\mathcal{K} = K_1 \ldots K_m$, ordered by their start position on the reference genome. Using the block alignment algorithm described in the previous section, we readily compute the pairwise similarity scores $S_{i,j} := S(C_i, K_j)$ of two blocks from Equation (6). We furthermore need the differences

$$\Delta_{i,j;k,l} = |\iota(C_j) - \iota C_i)| - |\iota(K_l) - \iota(K_k)| \quad (7)$$

of the distances between the pairs of blocks $C_i, C_j \in \mathcal{C}$ and $K_i, K_j \in \mathcal{K}$, respectively. Here $\iota(B)$ denotes the first position of block B on the reference genome. Since block groups by definition are located on the same contiguous chromosome or (super)contig and share the reading direction, the differences of coordinates are well defined.

In order to devise a Sankoff-style alignment algorithm, we consider the optimal alignment scores $S_{i,j;k,l}$ of the subsequence $\{C_i, C_{i+1}, \ldots, C_{j-1}, C_j\} \subseteq \mathcal{C}$ with the subsequence $\{K_k, K_{k+1}, \ldots, K_{l-1}, K_l\} \subseteq \mathcal{K}$. Furthermore, let $S^M_{i,j,k,l}$ be the best score of a block alignment subject to the constraint that C_i, C_j and K_k, K_l are two pairs of blocks that are included as a paired match into the alignment. The optimal scores then satisfy the recursions

$$S_{i,j;k,l} = \max \begin{cases} S_{i+1,j;k,l} + \gamma & \text{(deletion)} \\ S_{i,j;k+1,l} + \gamma & \text{(insertion)} \\ S_{i+1,j;k+1,l} + S_{i,k} & \text{(single)} \\ \max_{h \leq j, q \leq l}(S^M_{i,h;k,q} + S_{h+1,j;q+1,l}) & \text{(paired)} \end{cases}$$

$$S^M_{i,j;k,l} = S_{i+1,j-1;k+1,l-1} + \tau(S_{i,k}, S_{j,l}, \Delta_{i,j;k,l})$$

with the initialization $S_{i,j;k,l} = |(j-i) - (l-k)|\,\gamma + S_{ik}$. The constant $\gamma < 0$ denotes a gap penalty. The function $\tau(.)$ measures how well two pairs of blocks match in terms of both the similarity of the individual blocks and in terms of their mutual distances:

$$\tau_{i,j;k,l} = \upsilon_{\text{dist}} \cdot (1 - \Delta^2_{i,j;k,l}/\Delta_N) + \upsilon_{\text{block}}(S_{i,k} + S_{j,l}),$$

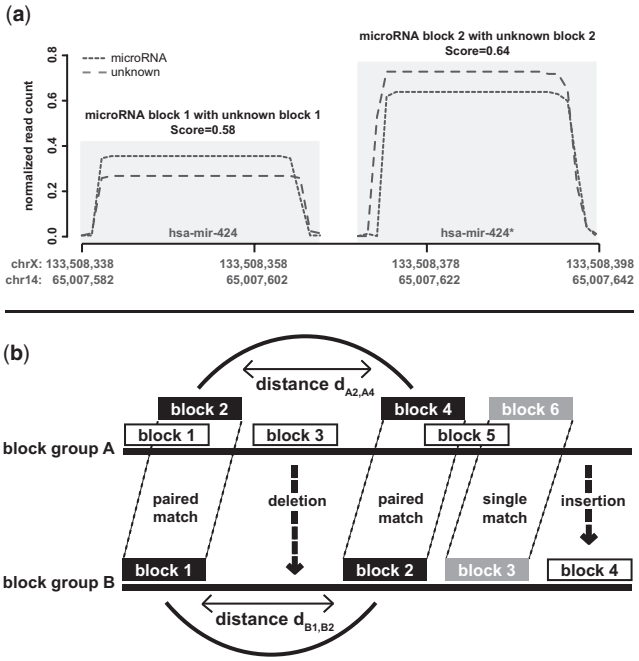

Fig. 2. Visualization of block and block group alignment steps of deepBlockAlign. (**a**) Block alignment computed between similarly placed blocks of a miRNA and an unannotated block group. Both the blocks have similar expression and precise arrangement of reads as also represented in Figure 4c for the same example. (**b**) A representation of alignment computed between two block groups using Sankoff algorithm. The algorithm optimizes the score based on the individual block similarities and pairwise block distances. Pairwise aligned blocks with similar distances are shown in black, single block alignments in gray and inserted or deleted blocks in white.

where $\Delta_N = 40$ is a normalization parameter, and υ_{dist} and υ_{block} are parameters to weight the influence of the distance between the blocks and the block scores, respectively. The default values of the parameters for block group alignment are $\gamma = -1$, $\upsilon_{\text{dist}} = 6$ and $\upsilon_{\text{block}} = 1$. Since, for two block groups to share similar read processing, the relative position of blocks should be same, we have kept a higher distance weight (υ_{dist}) as compared with block score weight (υ_{block}). This has made the block distance slightly more important than block alignment. However, we encounter various examples as included in Supplementary Figure S2, where the importance of block alignment is evident.

Finally, the score is normalized by dividing it with the greater score of the two block groups aligned with themselves. An example of the Sankoff style alignment of block groups is shown in Figure 2b.

2.6 Clustering

To determine an optimal clustering algorithm and the number of clusters that are most appropriate for our benchmark dataset (Human_eb), we used the R-package `clvalid` (Brock *et al.*, 2008). Given a range of clusters, `clvalid` computes the connectivity (Handl *et al.*, 2005), Dunn (Dunn, 1974) and Silhouette (Rousseeuw, 1987) indexes for various clustering algorithms (hierarchical, k-means, SOM and other) and suggests the optimal algorithm and clusters for the dataset. We tested for the presence of two to six clusters using eight clustering algorithms and observed hierarchical clustering with two clusters to be the most suitable for our dataset (Supplementary Fig. S3). Hence, the agglomerative method of average linkage hierarchical clustering as implemented in the R-package `pvclust` (Suzuki and Shimodaira, 2006) was used for subsequent analysis. `pvclust` computes the P-value for each cluster in hierarchical clustering using multiscale bootstrap resampling and

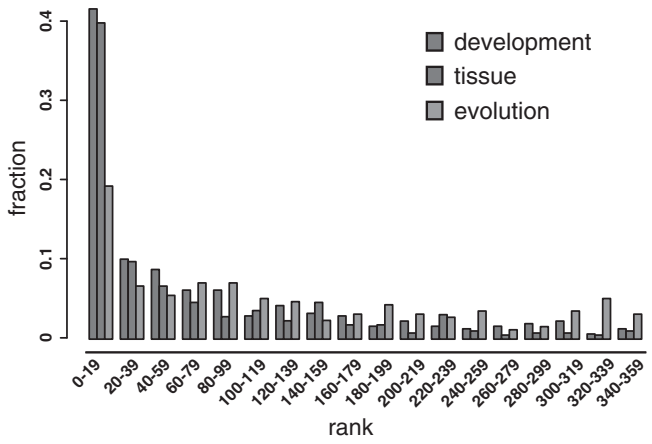

Fig. 3. Retrieval of expressed loci in different specimen solely based on read mapping profiles. The histogram shows for pairs of profiles from different developmental (red: Human_34 and Human_9), tissue (blue: Human_eb and Human_hesc) and evolutionary (green: Human_14 and Monkey_9) samples the best ranks found in the respective mate set, supporting non-random processing.

indicates how strong the cluster is supported by the data. Parameters were set to 10 000 bootstrap replicates, with relative sample sizes set from 0.5 to 1.4, incrementing in steps of 0.1. In this study, we have analyzed all the clusters having a $P < 0.1$.

3 RESULTS

3.1 Conservation of processing patterns

After mapping small RNAs to a reference genome, stacks of reads mapping to similar positions are merged to *read blocks* simplifying the visualization. Closely positioned blocks are joined in block groups.

Previous reports on the degradation of structured RNAs have suggested that, e.g. tRNA processing is largely a random process (Calabrese *et al.*, 2007). In order to assess whether a comparison of block patterns is meaningful at all, we first tested whether block patterns of specific loci are conserved across different experiments sampled from different developmental stages, tissues and species. To this end, we extracted from the datasets in Table 1 all those loci that are expressed in multiple experiments. We then aligned each block group with all block groups from another dataset and ranked the block groups by their `deepBlockAlign` scores. Figure 3 shows the distribution of the ranks of the query locus (or its rhesus ortholog) among all alignments. We find that `deepBlockAlign` ranks corresponding block groups close to the top for nearly half of the queries. Many block patterns are therefore highly non-random and conserved across different tissues, developmental stages and species.

3.2 Clustering of aligned block groups

In order to test whether `deepBlockAlign` can reliably distinguish different classes of structured RNAs, we performed an all-against-all alignment of the 455 block groups from the benchmark dataset. Using average linkage hierarchical clustering, we obtained the tree of significant clusters as shown in Figure 4 and Supplementary Figure S4. Two well-separated clusters were observed, one containing mainly microRNAs (red) and the other composed of tRNAs (blue). Within these two large clusters, 33 distinct subclusters were identified ($P < 0.1$), the largest one containing 90 and the smallest with only 2 block groups.

Within the miRNA cluster, two significant ($P < 0.1$) subclusters (Fig. 4a III and IV) contain most of the microRNAs. Subcluster IV represents miRNAs with an additional block directly upstream or downstream of the mature microRNA. These microRNA-offset RNAs (moRs) have been shown to be a distinct class of small RNAs that arise from pre-miRNA proximal regions in chordates as well as in humans (Langenberger *et al.*, 2009; Shi *et al.*, 2009). The clear separation of these two miRNA classes into different clusters provides a positive control. Some of the microRNAs are clustered rather far away from the majority of its class. Some of those distant miRNAs exhibit four or more blocks such as hsa-mir-103-2. Others lack one of the mature miRNAs resulting in either lower or higher distance between blocks undercutting or exceeding the standard loop distance of 10–20 nt. This is the case e.g. for hsa-mir-320a and hsa-mir-421 where miR and moR are expressed while the miR* is absent. In some cases, the microRNA designation may be a misannotation: the sequence of hsa-mir-1826, for example, is nearly identical to the human 5.8 rRNA.

No well-defined cluster was observed for snoRNAs. There can be several reasons for this: (i) low frequency of snoRNAs as compared with miRNAs or tRNAs in our dataset. (ii) No precise demarcation of entropy for snoRNAs (Fig. 1). While most of the miRNA and tRNA block groups were distinct in their entropy from each other, the entropy distribution for snoRNA, although distinct, overlapped with that of miRNA and tRNA. Consequently, more than half of the snoRNA block groups were clustered together with tRNAs, and 18 snoRNA block groups clustered together with miRNA (Supplementary Table S1). Eleven of these were having an entropy of <1.6. It is to be noted that low entropy does not indicate Dicer processing and further parameters such as similar processing patterns and expressions are necessary to support such a prediction. A more detailed inspection shows that the 18 snoRNA block groups exhibit Dicer-like processing patterns, characterized by (i) precise start position of the reads, (ii) 1–3 read blocks and (iii) 10–20 nt distance between the blocks (miR and miR*), see Figure 4b. Five of these 18 cases (ACA36b, ACA45, U27, U44 and HBI-100) have already been reported in earlier studies to be generating products with miRNA-like functions (Brameier *et al.*, 2011; Burroughs *et al.*, 2011). Since the Dicer processing results in similar patterns, this might be an explanation for snoRNAs clustering together with microRNAs (Fig. 4a II).

The tRNA cluster is more variable compared with the microRNA cluster, as evident from the step-like arrangement of clusters with low distance among each other. In contrast, in the microRNA cluster we see a constant distance to the root of the tree. This might be explained by the observation that the processing patterns for the tRNA class is not as coherent as for microRNAs. Since different tRNA loci seem to have conserved patterns across different experiments (Fig. 1), we assumed that tRNAs sharing the same anticodon would have similar processing patterns. Unfortunately, we were not able to find subclusters supporting this statement, suggesting that there is no specific pattern for different anticodon classes. However, we observed tRNAs having different anticodons (TGG, CGC, GCA, CGG, AGG), but highly similar processing patterns (Fig. 4a I and Supplementary Fig. S5).

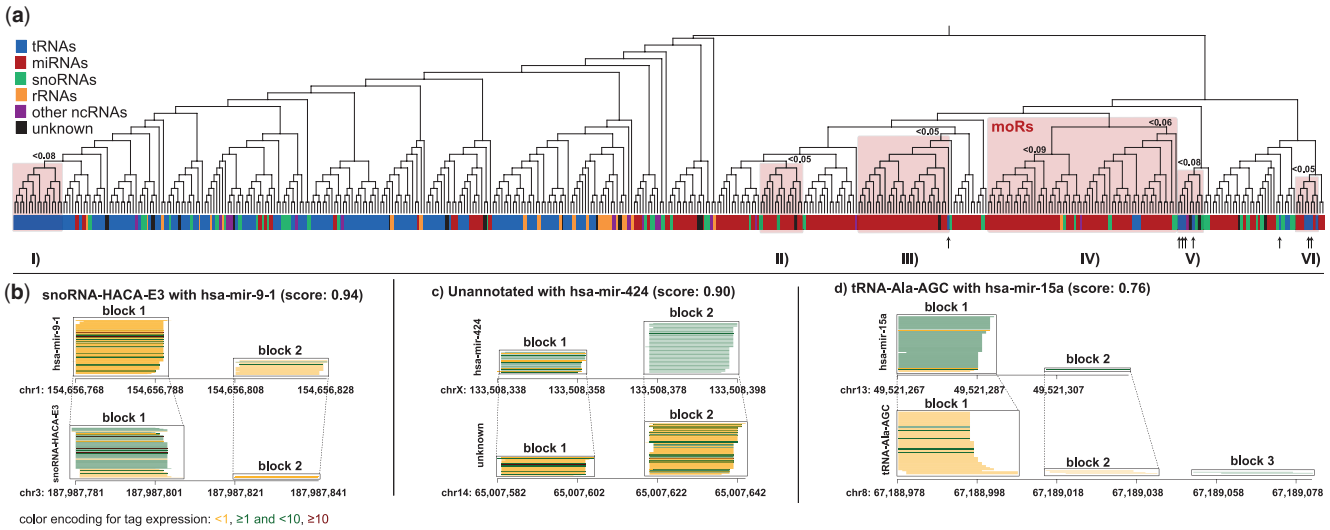

Fig. 4. Hierarchical clustering of 455 block groups based on alignment score from deepBlockAlign. (**a**) A tree visualizing the clustering. microRNA loci (red) are well separated from tRNA genes (blue). Within the microRNA cluster, microRNA-offset RNAs (moRs) can be found in one subcluster (IV), illustrating the different read pattern, caused by the additional blocks flanking the mature microRNA regions. Some significant clusters having tRNAs, snoRNAs or unannotated block groups clustering together with microRNAs (II, III, V and VI). tRNAs that are reported to generate products with miRNA-like features are highlighted with arrows. A cluster having tRNAs with different anti-codons but highly similar expression pattern (I). (**b**) A representation of the deepBlockAlign result for snoRNA-HACA-E3 significantly clustered together with hsa-mir-9-1. The snoRNA candidate shows not only well-placed blocks, like the microRNA, but also precise read arrangements at the 5′ end, suggesting a Dicer processing. (**c**) Alignment of an unknown block group with the hsa-mir-424 microRNA. (**d**) Alignment of the tRNA-Ala-AGC with hsa-mir-15a. The tRNA shows a microRNA-like read arrangement and is similar to the example presented from Cole *et al.* (2009), having most of the reads stacked at the 5′ end of the tRNA.

Interestingly, an earlier study reported a set of individual and characteristic tRNA-derived fragments that are actively derived from mature tRNAs by specific endonucleotic cleavage or exonuclease digestion by a number of enzymes (Lee *et al.*, 2009). Similarly, a Dicer-dependent processing was suggested for a few tRNAs (Babiarz *et al.*, 2008; Cole *et al.*, 2009). In addition, it was shown that Dicer-dependent small tRNA fragments, along with other small RNAs from a number of non-miRNA sources, can potentially bind to Argonaute complexes and thereby unfold *trans*-silencing capacities (Burroughs *et al.*, 2011; Haussecker *et al.*, 2010) Therefore, we examined the 13 tRNAs clustered significantly ($P < 0.1$) within the microRNA cluster (Fig. 4a V, VI and Supplementary Table S1). These 13 block groups align with higher scores to microRNAs than to other tRNAs. By taking a closer look at these candidates, we identified eight (sharing four different anticodons) that have been reported in literature. Lee *et al.* (2009), assume that Dicer might be involved in the 3′ maturation of tRNA_Ala (AGC) and tRNA_Ser (AGA) and Cole *et al.* (2009), suggested dicer processing for tRNA_Lys (TTT) and tRNA_Gln (CTG) with further experimental validation for tRNA_Gln (CTG).

3.3 Novel ncRNA candidates clustering together with known classes

Furthermore, there are 18 block groups without annotation aligning well with known classes, as exemplified in Figure 4c. Six of these fall into the microRNA cluster, while 12 cluster with the tRNAs. Analyzing the candidates on the microRNA side, we observed that two lie in an antisense direction to already annotated microRNAs (hsa-mir-486 and hsa-mir-625). This kind of antisense microRNA reads have been reported before (Stark *et al.*, 2008) and can frequently be observed when analyzing short RNA-seq data. The antisense reads, however, do not necessarily imply the actual transcription of such an RNA, since the complementary stem regions in some cases cannot be distinguished. Upon a detailed inspection, we observed some strand-specific tags for both hsa-mir-486 and hsa-mir-625 (Supplementary Figs S6 and S7). However, considering the perfect complementarity of hairpins in the two miRNAs and low frequency of strand-specific tags especially for hsa-mir-625, it is difficult to assume these two miRNAs as an ideal case of anti-sense miRNA.

Two additional block groups significantly align with microRNAs and show a typical microRNA processing pattern. However, when analyzing the secondary structure of these candidates using `RNAfold` (Hofacker *et al.*, 1994), no hairpin-like structure was observed. However, based on the expression patterns, these examples are clustered correctly. Since `deepBlockAlign` does not take any secondary structure into account, it cannot be expected that all the results will overlap with ncRNA prediction programs. These results thus require further validation. Two candidates clustered together with an snRNA and snoRNA, respectively. Upon a detailed inspection of the respective block groups, none of the two candidates were observed to be having microRNA-like processing pattern.

Six of the 12 candidates in the tRNA cluster overlap known tRNA-derived pseudogenes. Two further loci correspond to two deleted miRBase microRNAs (hsa-mir-1974 and hsa-mir-1978), which had been recognized as mitochondrial tRNA sequences. Three of the remaining four candidates lie within exonic regions and are thus not likely to be ncRNAs. The last one shows two blocks in close

distance (<5 nt) and lies in intergenic region with no annotations. The sequence does not fold into any defined secondary structure and further analysis has to be carried out in order to annotate it.

4 DISCUSSION

We presented an approach, `deepBlockAlign`, and showed that it can be used for a meaningful clustering of ncRNAs based solely on read processing patterns. In particular, we find that the mapping profiles are well conserved between human and macaque. Most microRNAs as well as the majority of the tRNAs fall into well-separated clusters (Fig. 4). Within the microRNA cluster, a subcluster contains the majority of microRNA-offset RNAs, indicating that `deepBlockAlign` is able to precisely distinguish between block groups that share a common core pattern. Consistent with observation that some snoRNAs are processed by Dicer, we find the examples clustered together with microRNAs. Several previously unannotated clusters were identified as potential antisense microRNAs and as tRNA-derived pseudogenes, respectively, showing that `deepBlockAlign` can be used for annotating unknown read mapping patterns through unsupervised clustering. The application of `deepBlockAlign` for annotation of unknown processing patterns on a routine basis, however, will require the development of appropriate measures of statistical significance, such as *P*- or *E*-values. This will require further research as it remains unclear at this point how appropriate background distributions could be constructed. Future updates of the algorithm also includes a more detailed tuning with respect to match versus mismatch scores. We found that this approach is fairly robust against parameter variation. For instance, we tested the robustness of the `deepBlockAlign` algorithm by analyzing the benchmark dataset using various values of the distance weight parameter υ_{dist} observing consistent results (Supplementary Table S2). The clustering approach can in principle be used for constructing multiple alignments. This could in turn be useful in identifying subtle differences in processing patterns and assist the investigation of evolution of processing patterns.

Qualitatively, the read-based clusters closely resemble the results of clustering known and predicted ncRNAs based on their secondary structure (Kaczkowski *et al.*, 2009; Will *et al.*, 2007). We suspect that this is not a coincidence, since small RNAs are preferentially produced from base paired regions (Langenberger *et al.*, 2010). This suggests that read mapping patterns are likely to be influenced, or even determined, by the secondary structure of the parental RNA. This signal appears to be stronger than variations depending on sequencing protocol and GC-content that have previously been reported e.g. by Hansen *et al.* (2010); Li *et al.* (2010).

In the case of tRNAs, chemical modifications are the second major contribution shaping the read mapping patterns (Findeiß *et al.*, 2011). Interestingly, there is a single cluster comprising tRNAs with several different anticodons and isoacceptors that share an almost perfect read processing pattern. This observation requires deeper analysis for further explanation. The read processing patterns of loci with low expression levels may be biased by random fluctuations, thus we have only included patterns with a minimum expression of 50 reads. RNA-seq data with deeper coverage will thus not only improve the clustering results but also increase the number of block groups and thereby facilitate the detection of novel ncRNAs.

Funding: Danish Strategic Research Council (Strategic Growth technologies); Danish Independent Research Council (Technology and Production); Danish Center for Scientific Computation, in part. This publication is supported by LIFE – Leipzig Research Center for Civilization Diseases, Universität Leipzig. This project was funded by means of the European Social Fund and the Free State of Saxony.

Conflict of Interest: none declared.

REFERENCES

Akbasli,E. (2007) *Fast Sequence Alignment in a Managed Programming Language*. Master's Thesis, IT University, Copenhagen, Denmark.

Ando,Y. *et al.* (2011) Two-step cleavage of hairpin RNA with 5′ overhangs by human DICER. *BMC Mol. Biol.*, **12**, 6.

Babiarz,J. *et al.* (2008) Mouse es cells express endogenous shrnas, sirnas, and other microprocessor-independent, dicer-dependent small rnas. *Genes Dev.*, **22**, 2773.

Brameier,M. *et al.* (2011) Human box C/D snornas with miRNA like functions: expanding the range of regulatory RNAs. *Nucleic Acids Res.*, **39**, 675–686.

Brock,G. *et al.* (2008) clValid: an R package for cluster validation. *J. Stat. Softwr.*, **25**, 1–22.

Burroughs,A.M. *et al.* (2011) Deep-sequencing of human argonaute-associated small RNAs provides insight into miRNA sorting and reveals Argonaute association with RNA fragments of diverse origin. *RNA Biol.*, **8**, 158–177.

Calabrese,J.M. *et al.* (2007) RNA sequence analysis defines Dicer's role in mouse embryonic stem cells. *Proc. Natl Acad. Sci. USA*, **104**, 18097–18102.

Caprara,A. *et al.* (2004) 1001 optimal PDB structure alignments: integer programming methods for finding the maximum contact map overlap. *J. Comput. Biol.*, **11**, 27–52.

Chan,P.P. and Lowe,T.M. (2009) GtRNAdb: a database of transfer RNA genes detected in genomic sequence. *Nucleic Acids Res.*, **37**, D93–D97.

Cifuentes,D. *et al.* (2010) A novel miRNA processing pathway independent of Dicer requires Argonaute2 catalytic activity. *Science*, **328**, 1694–1698.

Cole,C. *et al.* (2009) Filtering of deep sequencing data reveals the existence of abundant Dicer-dependent small RNAs derived from tRNAs. *RNA*, **15**, 2147–2160.

Crooks,G.E. *et al.* (2005) Pairwise alignment incorporating dipeptide covariation. *Bioinformatics*, **21**, 3704–3710.

Dunn,J. (1974) Well-separated clusters and optimal fuzzy partitions. *Cybern. Syst.*, **4**, 95–104.

Erhard,F. and Zimmer,R. (2010) Classification of ncRNAs using position and size information in deep sequencing data. *Bioinformatics*, **26**, i426–i432.

Findeiß,S. *et al.* (2011) Traces of post-transcriptional RNA modifications in deep sequencing data. *Biol. Chem.*, **392**, 305–313.

Gan,J. *et al.* (2008) A stepwise model for double-stranded RNA processing by ribonuclease III. *Mol. Microbiol.*, **67**, 143–154.

Gorodkin,J. and Hofacker,I.L. (2011) From structure prediction to genomic screens for novel non-coding RNAs. *PLoS Comput. Biol.*, **7**, e1002100.

Gorodkin,J. *et al.* (2010) De novo prediction of structured RNAs from genomic sequences. *Trends Biotech.*, **28**, 9–19.

Handl,J. *et al.* (2005) Computational cluster validation in post-genomic data analysis. *Bioinformatics*, **21**, 3201.

Hansen,K. *et al.* (2010) Biases in illumina transcriptome sequencing caused by random hexamer priming. *Nucleic Acids Res.*, **38**, e131.

Haussecker,D. *et al.* (2010) Human tRNA-derived small RNAs in the global regulation of RNA silencing. *RNA*, **16**, 673–695.

Hinrichs,A.S. *et al.* (2006) The UCSC genome browser database: update 2006. *Nucleic Acids Res.*, **34**, D590–D598.

Hofacker,I.L. *et al.* (1994) Fast folding and comparison of RNA secondary structures. *Chem. Month.*, **125**, 167–188.

Hoffmann,S. *et al.* (2009) Fast mapping of short sequences with mismatches, insertions and deletions using index structures. *PLoS Comput. Biol.*, **5**, e1000502.

Jung,C.H. *et al.* (2010) Identification of novel non-coding RNAs using profiles of short sequence reads from next generation sequencing data. *BMC Genomics*, **11**, 77.

Kaczkowski,B. *et al.* (2009) Structural profiles of human miRNA families from pairwise clustering. *Bioinformatics*, **25**, 291–294.

Karolchik,D. *et al.* (2004). The UCSC table browser data retrieval tool. *Nucleic Acids Res.*, **32**, D493–D496.

Kawaji,H. *et al.* (2008) Hidden layers of human small RNAs. *BMC Genomics*, **9**, 157.

Kozomara,A. and Griffiths-Jones,S. (2011) miRBase: integrating microRNA annotation and deep-sequencing data. *Nucleic Acids Res.*, **39**, D152–D157.

Langenberger,D. *et al.* (2009) Evidence for human microRNA-offset RNAs in small RNA sequencing data. *Bioinformatics*, **25**, 2298–2301.

Langenberger,D. *et al.* (2010) Identification and classification of small RNAs in transcriptome sequence data. In *Pacific Symposium Biocomputing*. Vol. 15, pp. 80–87.

Langmead,B. *et al.* (2009) Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol.*, **10**, R25.

Lee,Y. *et al.* (2009) A novel class of small RNAs: tRNA-derived RNA fragments (tRFs). *Genes Dev.*, **23**, 2639–2649.

Li,J. *et al.* (2010) Method modeling non-uniformity in short-read rates in rna-seq data. *Genome Biol.*, **11**, R25.

Lu,Z.J. *et al.* (2011) Prediction and characterization of noncoding RNAs in *C. elegans* by integrating conservation, secondary structure, and high-throughput sequencing and array data. *Genome Res.*, **21**, 276–285.

Miyoshi,K. *et al.* (2010) Many ways to generate microRNA-like small RNAs: non-canonical pathways for microRNA production. *Mol. Genet. Genomics*, **284**, 95–103.

Morin,R. *et al.* (2008) Application of massively parallel sequencing to microRNA profiling and discovery in human embryonic stem cells. *Genome Res.*, **18**, 610–621.

Needleman,S.B. and Wunsch,C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, **48**, 443–453.

Rousseeuw,P. (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.*, **20**, 53–65.

Sankoff,D. (1985) Simultaneous solution of the RNA folding, alignment and protosequence problems. *SIAM J. Appl. Math.*, **45**, 810–825.

Shi,W. *et al.* (2009) A distinct class of small RNAs arises from pre-miRNA-proximal regions in a simple chordate. *Nat. Struct. Mol. Biol.*, **16**, 183–189.

Somel,M. *et al.* (2010) MicroRNA, mRNA, and protein expression link development and aging in human and macaque brain. *Genome Res.*, **20**, 1207–1218.

Stark,A. *et al.* (2008) A single Hox locus in Drosophila produces functional microRNAs from opposite DNA strands. *Genes Dev.*, **22**, 8–13.

Suzuki,R. and Shimodaira,H. (2006) Pvclust: an R package for assessing the uncertainty in hierarchical clustering. *Bioinformatics*, **22**, 1540–1542.

Taft,R.J. *et al.* (2009) Small RNAs derived from snoRNAs. *RNA*, **15**, 1233–1240.

Trapnell,C. *et al.* (2009) TopHat: discovering splice junctions with RNA-Seq. *Bioinformatics*, **25**, 1105–1111.

Underwood,J.G. *et al.* (2010) FragSeq: transcriptome-wide RNA structure probing using high-throughput sequencing. *Nat. Methods*, **7**, 995–1001.

Will,S. *et al.* (2007) Inferring noncoding RNA families and classes by means of genome-scale structure-based clustering. *PLoS Comput. Biol.*, **3**, e65.