High-throughput sequencing and small non-coding RNAs

Von der Fakultät für Mathematik und Informatik
der Universität Leipzig
angenommene

Dissertation

zur Erlangung des akademischen Grades

Doctor rerum naturalium
(Dr. rer. nat.)

im Fachgebiet

Informatik

vorgelegt

von Diplom-Ingenieur (FH) David Langenberger
geboren am 28. November 1981 in Forchheim

Die Annahme der Dissertation wurde empfohlen von:
1. Professor Dr. Peter F. Stadler, Institut für Informatik, Universität Leipzig
2. Akad. Oberrätin Dr. Kay Nieselt, Fachbereich Informatik, Universität Tübingen

Die Verleihung des akademischen Grades erfolgt mit Bestehen
der Verteidigung am 22. April 2013 mit dem Gesamtprädikat magna cum laude.
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>DARIO: a ncRNA detection and analysis tool for next-generation sequencing experiments</td>
<td>53</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>54</td>
</tr>
<tr>
<td>4.2</td>
<td>Materials and methods</td>
<td>55</td>
</tr>
<tr>
<td>4.3</td>
<td>Results and discussion</td>
<td>57</td>
</tr>
<tr>
<td>4.4</td>
<td>Conclusion</td>
<td>61</td>
</tr>
<tr>
<td>5</td>
<td>MicroRNA or not microRNA?</td>
<td>63</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>64</td>
</tr>
<tr>
<td>5.2</td>
<td>Materials and methods</td>
<td>66</td>
</tr>
<tr>
<td>5.3</td>
<td>Discussion</td>
<td>69</td>
</tr>
<tr>
<td>6</td>
<td>DeepBlockAlign: Aligning short RNA-seq block patterns</td>
<td>71</td>
</tr>
<tr>
<td>6.1</td>
<td>Introduction</td>
<td>72</td>
</tr>
<tr>
<td>6.2</td>
<td>Material and methods</td>
<td>74</td>
</tr>
<tr>
<td>6.3</td>
<td>Results</td>
<td>81</td>
</tr>
<tr>
<td>6.4</td>
<td>Discussion</td>
<td>85</td>
</tr>
<tr>
<td>7</td>
<td>Dicer-Processed small RNAs: Rules and exceptions</td>
<td>87</td>
</tr>
<tr>
<td>7.1</td>
<td>Introduction</td>
<td>88</td>
</tr>
<tr>
<td>7.2</td>
<td>Materials and methods</td>
<td>89</td>
</tr>
<tr>
<td>7.3</td>
<td>Results</td>
<td>92</td>
</tr>
<tr>
<td>7.4</td>
<td>Discussion</td>
<td>101</td>
</tr>
<tr>
<td>8</td>
<td>Traces of post-Transcriptional RNA modifications in deep sequencing data</td>
<td>103</td>
</tr>
<tr>
<td>8.1</td>
<td>Introduction</td>
<td>104</td>
</tr>
<tr>
<td>8.2</td>
<td>Materials and methods</td>
<td>105</td>
</tr>
<tr>
<td>8.3</td>
<td>Results</td>
<td>107</td>
</tr>
<tr>
<td>9</td>
<td>Conclusion</td>
<td>113</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
<td>117</td>
</tr>
<tr>
<td>Curriculum Scientiae</td>
<td></td>
<td>a</td>
</tr>
<tr>
<td>Publications</td>
<td></td>
<td>c</td>
</tr>
</tbody>
</table>
Abstract

Before the discovery of microRNAs in 2001 it was widely believed that most of the human genome, namely everything not coding for proteins, is just 'junk DNA'. Meanwhile, these tiny RNAs have been shown to be functional by regulating the expression of thousands of genes. RNAs are no longer just carriers of genetic information, but performers of important regulatory tasks within the cell.

A typical microRNA is processed from a primary pol II transcript and cut by the Drosha enzyme, resulting in a characteristic hairpin of length 60-120 nucleotides. This precursor microRNA is then transported by the protein Exportin-5 to the cytoplasm, where the hairpin is processed by the enzyme Dicer into a double stranded RNA about 22nt in length with a 2nt 3’-overhang. The obtained mature microRNA is then incorporated into a protein complex named RISC and one strand is selected (miR), while the other one is degraded (miR*). The microRNA performs the post-transcriptional gene regulation by perfectly or imperfectly binding to cis-regulatory target sites in the 3’ UTR of messenger RNAs. It is predicted that around one third of all human genes are regulated by one or more microRNAs. But microRNAs are not alone: when measuring the amount of RNA molecules within a cell, only 1-5% are protein-coding RNAs, while the rest comes from non-protein-coding RNAs (ncRNAs) of which only <10% are microRNAs. A lately published article of the 'ENCODE project' highlights the functionality of these RNA molecules. They assigned biochemical functions to ~80% of the human genome, while only 1.5% code for proteins.

In this thesis I investigate the processing mechanisms of these short ncRNAs by using data generated by the current method of high-throughput sequencing (HTS). The recently adapted short RNA-seq protocol allows the sequencing of RNA fragments of microRNA-like length (~18-28nt). Thus, after mapping the data back to a reference genome, it is possible to not only measure, but also visualize the expression of all ncRNAs that are processed to fragments of this specific length. For microRNAs a typical pattern of two distinct stacks of reads, representing the miR and miR* sequences, can be observed.
I used short RNA-seq data to show that a highly abundant class of small RNAs, called microRNA-offset-RNAs (moRNAs), which was formerly detected in a basal chordate, is also produced from human microRNA precursors. These additional RNAs are generated from sequences immediately adjacent to mature miR and miR* loci. Like mature miRNAs, they are ∼22nt long, developmentally regulated, and appear to be produced by RNAse III-like processing from the precursor miRNA hairpin. This observation prompted me to specifically search for analogous patterns in human small RNA sequencing libraries. To simplify the search, we developed the blockbuster tool that automatically recognizes blocks of reads to detect specific expression patterns. By using blockbuster, blocks from moRNAs were detected directly next to the miR or miR* blocks and could thus easily be registered in an automated way. Further analysis showed that the expression levels of moRNAs are unrelated to those of the associated microRNAs. We could also show that their microRNA precursors are typically evolutionarily old.

When further investigating the short RNA-seq data I realized that not only microRNAs give rise to short ∼22nt long RNA pieces, but also almost all other classes of ncRNAs, like tRNAs, snoRNAs, snRNAs, rRNAs, Y-RNAs, or vault RNAs. Only for some types, like snoRNAs or microRNAs, it was already known that they undergo specific maturation processes that lead to the production of shorter RNAs. The formed read patterns that arise after mapping these RNAs back to a reference genome seem to reflect the processing of each class and are thus specific for the RNA transcripts of which they are derived from. I explored the potential of this patterns in classification and identification of non-coding RNAs. Using a random forest classifier which was trained on a set of characteristic features of the individual ncRNA classes, it was possible to distinguish three types of ncRNAs, namely microRNAs, tRNAs, and snoRNAs. With Positive Predictive Values (PPV) and recall rates of ∼0.8 for all three classes, the classifier performed well and I used it to predict new ncRNA candidates. Another finding of the performed analysis of this dataset is the direct connection of the read patterns to the predicted secondary structure of the RNAs. The pairing probabilities of bases covered by HTS reads are significantly increased, indicating the necessity of properly paired nucleotides for processing.

To make the classification available to the research community, we developed a free web service that allows to study short read data from small RNA-seq experiments. This web server is called DARIO and it provides a wide range of analysis features, including quality control, read normalization, ncRNA quantification, and prediction of putative ncRNA candidates using the random forest classifier. The web site supports six species: human, rhesus monkey, mouse, fruit fly, worm, and zebrafish. After file upload, a single job typically takes between 5 and 30 minutes and the results are summarized on a single web page containing job details, quality control measures and figures, ncRNA quantification and classification.
The classification has shown that read patterns are specific for different classes of ncRNAs. To make use of this feature, we developed the tool deepBlockAlign. deepBlockAlign introduces a two-step approach to align read patterns with the aim of quickly identifying RNAs that share similar processing footprints. Overlapping mapped reads are first merged to blocks using the earlier developed tool blockbuster 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 patterns 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.

It has already been shown that Dicer is not only involved in microRNA biogenesis. It appears to be also involved in the processing of other small RNA species beyond canonical microRNAs. In order to find possible exceptions to the well-known microRNA maturation by Dicer and to identify additional substrates for Dicer processing I re-evaluated the small RNA sequencing data of a Dicer knockdown experiment in MCF-7 cells. While the prominent non-Dicer mir-451 was not sufficiently expressed in these experiments, there were several additional Dicer-independent microRNAs, among them the important tumor supressor mir-663a. I recovered previously described examples of non-miRNA Dicer substrates such as tRNA-Gln and several snoRNAs. Interestingly, snoRNA-derived RNAs from box C/D snoRNAs are Dicer-independent, while those from box H/ACA snoRNAs are often Dicer dependent. Several pol-III transcripts, in particular the vault RNAs and the great ape specific snaRs are processed by Dicer, while the small RNAs originating from Y RNAs seemed to be Dicer independent.

It is known that many aspects of the RNA maturation leave traces in RNA sequencing data in the form of mismatches from the reference genome. I was able to recover many well-known modified sites in tRNAs, providing evidence that modified nucleotides are a pervasive phenomenon in these data sets. Furthermore, I checked if non-encoded CCA tails, which are post-transcriptionally added to tRNAs, can be seen in short RNA-seq data. Surprisingly, they can be found in a diverse collection of transcripts, including sub-populations of mature microRNAs.
Acknowledgments

First of all, I thank my supervisors Peter F. Stadler and Steve Hoffmann for their continuous support and extremely helpful discussions.

Furthermore, I thank all current and previous members of the bioinformatics group at the University of Leipzig for their scientific advice and support. Special thanks go to Anke Busch, Stephanie Keller-Schmidt, Mario Fasold, Lydia Hopp, Alexander Donath, Jörg Lehmann Markus Riester, Dominic Rose, and Gero Doose.

Additionally, I would like to say thank you to all these that mostly remain unmentioned in sciences: family and friends that indirectly contributed to this work by, for example, reading and reviewing manuscripts, discussing different topics related to my work or putting their trust in me. Here I want to emphasize Elisabeth, who supported me in every step and believed in me.

This PhD thesis was conducted within the framework of the Leipzig Research Centre for Civilization Diseases (LIFE). In this context, I would like to thank the European Social Fund (ESF) of the European Union and the Free State of Saxony for funding three full years of my PhD studies. Further, I want to thank my former university 'Fachhochschule Weihenstephan' (University of Applied Sciences Weihenstephan) and especially Prof. Dr. Frank Leßke for supporting me and my goal of doing the PhD.
This thesis is based on the following publications:


**Computational prediction of microRNA genes.** RNA sequence, structure and function: computational and bioinformatic methods, in press.


**Recurrent mutation of the ID3 gene in Burkitt lymphoma identified by integrated genome, exome and transcriptome sequencing.** Nature genetics 11;44(12):1316-1320. * authors contributed equally


**deepBlockAlign: A tool for aligning RNA-seq profiles of read block patterns.** Bioinformatics 1;28(1):17-24. * authors contributed equally


**DARIO: A ncRNA detection and analysis tool for next-generation sequencing experiments.** Nucleic Acids Res. 39(Web Server issue):W112-7. * authors contributed equally


# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>About this work</td>
<td>3</td>
</tr>
<tr>
<td>1.2</td>
<td>Genome, transcriptome and proteome</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>Non-coding RNAs</td>
<td>7</td>
</tr>
<tr>
<td>1.3.1</td>
<td>Different types of non-coding RNAs</td>
<td>7</td>
</tr>
<tr>
<td>1.3.2</td>
<td>RNA interference and the microRNA pathway</td>
<td>8</td>
</tr>
<tr>
<td>1.4</td>
<td>Sequencing methods</td>
<td>12</td>
</tr>
<tr>
<td>1.4.1</td>
<td>A short history about sequencing</td>
<td>13</td>
</tr>
<tr>
<td>1.4.2</td>
<td>454 pyrosequencing</td>
<td>14</td>
</tr>
<tr>
<td>1.4.3</td>
<td>Illumina / Solexa</td>
<td>16</td>
</tr>
<tr>
<td>1.4.4</td>
<td>Short RNA-seq</td>
<td>17</td>
</tr>
<tr>
<td>1.5</td>
<td>Sequencing data</td>
<td>18</td>
</tr>
<tr>
<td>1.5.1</td>
<td>Data format</td>
<td>19</td>
</tr>
<tr>
<td>1.5.2</td>
<td>Short read mapping</td>
<td>20</td>
</tr>
<tr>
<td>1.5.3</td>
<td>Error sources</td>
<td>23</td>
</tr>
<tr>
<td>1.6</td>
<td>Short RNA-seq and microRNAs</td>
<td>24</td>
</tr>
<tr>
<td>1.6.1</td>
<td>The microRNA pattern</td>
<td>25</td>
</tr>
<tr>
<td>1.6.2</td>
<td>MicroRNA gene prediction using structure and read patterns</td>
<td>25</td>
</tr>
<tr>
<td>1.6.3</td>
<td>MicroRNA-like processing products from other ncRNAs</td>
<td>27</td>
</tr>
<tr>
<td>1.7</td>
<td>Computational methods</td>
<td>28</td>
</tr>
<tr>
<td>1.7.1</td>
<td>Short read mapping: segemehl</td>
<td>29</td>
</tr>
<tr>
<td>1.7.2</td>
<td>Machine learning technique: random forest classifier</td>
<td>30</td>
</tr>
<tr>
<td>Introduction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.7.3</td>
<td>Hierarchical clustering: pvclust</td>
<td>31</td>
</tr>
</tbody>
</table>

---

---

---

---

---

---
For almost one decade now, non-protein-coding RNAs (ncRNAs) have been well known to not only act as important adapter molecules within the cell, like the amino acid transporting transfer-RNAs (tRNAs), but also directly interact with the protein construction (expression) apparatus in order to regulate the production of a great amount of proteins. Thus, these regulatory interactions add a new layer of complexity. It was quickly realized, that errors in this new network can lead to major mis-regulation and thus disease. The race for finding unknown ncRNAs in the human genome started and the number of new predictions of known classes and even new ncRNA classes increased monthly. Nowadays, the new technique of high-throughput sequencing (HTS) allows the measurement of huge amounts of RNA and provides a possibility to validate and quantify known and newly predicted ncRNAs. HTS has also successfully been used to predict microRNA genes by developing a new sequencing protocol, specifically measuring molecules of microRNA-like length (short RNA-seq).

1.1 About this work

In this work, I take a deeper look into data from short RNA-seq experiments. These datasets only contain RNAs that were processed within the cell to smaller pieces, which are thought to be functional. I realized, that these short RNAs form specific patterns, when being mapped back to a reference genome and that these patterns can be used to classify different types of ncRNAs. I designed algorithms to cluster these piles of mapped molecules, assign them to ncRNA classes, align them and classify unknown ones. The classification algorithm was also made available to the research community by an easy to use web server. Furthermore, I took a deeper look into a widely used database of known microRNAs and discovered, that there are several false annotations. Continuative studies showed that it is hard to distinguish some ncRNA classes, since they seem to be processed by the same mechanism, ending up in similar patterns. This well known behavior was then double-checked using another analysis in which this mechanism was switched off in-vivo. Overall, in this work, I used the new method of HTS to study and understand the processing of ncRNAs and I used this data to predict new ncRNA candidates.

1.2 Genome, transcriptome and proteome

The human body is built up of around $10^{13}$ cells. Every single cell contains a genome storing the identical genetic information, which is specific for each human being. This information is stored using DNA (deoxyribonucleic acid). It is encoded by chaining up monomeric molecules called nucleotides to polymeric macromolecules. There are four different characters in the alphabet of the DNA, adenine (A), guanine (G), thymine (T) and cytosine (C). By putting
the human genome and the production of proteins. a) The human genome is situated in each cell of a human being. The nuclear genome is fractionated in 24 chromosomes, while the mitochondrial genome is stored in a circularized manner. b) The central dogma of molecular biology describes the way of how genetic information is expected to be processed. DNA is transcribed to RNA molecules, which are translated to proteins. c) After transcription of DNA to pre-mRNA the spliceosome cuts out the non-protein-coding introns. Pictures redrawn from (Brown, 2006).

these four nucleotides together, like characters in a book, nature found a way to write down the blueprints for all tools, which are needed by the cells to survive. When thinking of the genome as an encyclopedia, every single entry stores the information of how one specific tool has to be built up. These entries are called *genes* and the tools are molecules with important functions like enzymes, the workhorses within the cells.

The human genome is divided into two distinct parts, the nuclear genome and the mitochondrial genome. The nuclear genome consists of 3,137,144,693 nucleotides (GRCh37), separated into 24 chromosomes, which hold information for 20,110 protein-coding genes (Gencode V12 May 2012 freeze). The mitochondrial genome is a circularized molecule, 15,000-17,000 nucleotides in length and consists of just 13 protein-coding genes. Overall, the stored genetic code between two human beings is highly similar (∼99.9% identical, Clinton (2000)) and only very small differences in the genetic code of the proteins lead to the differences, like the color
of the hairs, the eyes, or the skin.

The genome itself is not able to release the stored information to the cell. For the so-called *expression* of genes, several enzymes and proteins are needed.

One of the first main actors is the RNA polymerase enzyme, which precisely finds the entry for a needed tool within the huge genome and generates a copy of only this entry. The copied information consists of RNA (ribonucleic acids) molecules. In contrast to DNA, the alphabet of the RNA has an uracil (U) instead of a thymine (T). The copying process is known as *transcription* and the copy of the protein-coding gene is called messenger RNA (mRNA). In the human genome, the encoded and transcribed genes consist of three main parts, protein-coding exons, non-protein-coding introns and untranslated regions (UTRs) at the 5’ and the 3’ ends (see Figure 1.1c). The UTRs are important, since they contain functional sequences for further processing and regulatory motifs. The introns are not needed for the protein productions and thus they are found and deleted by a complex within the nucleus, the spliceosome (see Figure 1.1c). This process is called *splicing* and results in an intron-free mature mRNA sequence. The sum of all transcribed RNA polymers is known as the transcriptome of a cell.

In a next step, the encoded information of a mRNA has to be used to design proteins. The synthesis of proteins from single RNA molecules is called *translation*. The copies of the genes, the mRNAs, are found by ribosomes, which read the text and build together the proteins, using amino acids, the building blocks of proteins. The collectivity of all proteins is called proteome. The proteins are used to build up the cell and perform important tasks for its function.

The explained process of the expression of a gene with the transcription of DNA to RNA and the translation to proteins is known as the ‘central dogma of molecular biology’ (Crick, 1958; Crick *et al.*, 1970) (see Figure 1.1b).

It is quite obvious that cells from different tissues, like cells from the heart and cells from the skin, have to have an unequal behavior and a different construction. These adapted functions of cells can be reached by adjusting the expression of proteins. Skin cells for example have to be more robust, and thus more proteins for stabilizing the cell walls are created. To regulate the expression of genes and thus end up with a set of needed proteins for a specific function, the transcription of genes can be turned on and off (Latchman, 2005). It was long thought, that this regulation is mainly handled by two different regulatory layers: 1) Epigenetic modifications control the readability of genes by preventing the polymerase from binding the DNA (Khavari *et al.*, 2010) (see Figure 1.3a). 2) Transcription factors, a special kind of proteins, bind next to the start of a gene and activate or inhibit the transcription of this locus (Latchman, 1997; Karin *et al.*, 1990) (see Figure 1.3b).

Even though different compositions of proteins can build up hundreds of different types of cells (Levine and Tjian, 2003; Buchler *et al.*, 2003), only \(\sim1.5\%\) of the human genome code
It was long believed that the rest of the genome is just 'junk DNA' (Ohno, 1972; Comings, 1972) without any functionality.

There are several widely discussed hypotheses, why there should be that amount of useless DNA in our genome, like the protection against mutations (Yunis \textit{et al.}, 1971), or due to evolutionary accumulation of dysfunctional genes (Brosius and Gould, 1992). But there are some striking observations, conflicting with the assumption that these parts are really useless and thus 'junk'. First of all, there are some well known molecules that are functional in their RNA state, like transfer RNAs (tRNAs) or ribosomal RNAs (rRNAs). The genes that code for the named RNAs are transcribed, but not translated to proteins (Ladner \textit{et al.}, 1975; Kim \textit{et al.}, 1973; Yusupov \textit{et al.}, 2001). Furthermore, when measuring the amount of RNA molecules within a cell, only 1-5% are mRNAs and thus come from protein-coding RNAs (Maniatis, 1989). The rest consists of 80-85% rRNAs and 10-15% other small RNAs (tRNAs, microRNAs, etc.) which do not result in proteins. A lately published article of the 'ENCODE project' highlights the functionality of these RNA molecules. The researchers have systematically analyzed transcribed regions and assigned biochemical functions to $\sim$80% of the genome (Khatun, 2012). This discovery of the 'ENCODE project' is a logical consequence of several observations made years before. It was realized, that a huge amount of transcribed RNA pieces show regulatory functionality and a new subgroup of RNAs, the non-(protein)-coding RNAs (ncRNAs), was born. Since then, the transcriptome of a cell is divided into two parts, the protein-coding RNAs and the ncRNAs (see Figure 1.2). While the protein-coding fraction follows the 'central dogma of molecular biology', the ncRNAs are functional in their RNA state. These functional RNA molecules have several essential roles within the cell.

![Figure 1.2: The transcriptome of the human cell can be divided into two fractions, protein-coding RNAs and ncRNAs. Picture redrawn from (Brown, 2006).](image-url)
1.3 Non-coding RNAs

Non-coding RNAs have been known for quite some while. The first known ncRNAs were tRNAs (transfer RNAs) and rRNAs (ribosomal RNAs). These types of RNAs have well-known and important functions within the human cells. At the beginning of this century, nevertheless, several new ncRNAs have been found and analyzed. Short RNA molecules, like miRNAs (microRNAs), piRNAs (PIWI interacting RNAs), siRNAs (short interfering RNAs), snoRNAs (small nucleolar RNAs), or snRNAs (small nuclear RNAs) show regulatory functionality. This army of tiny regulators changed the thitherto picture of the regulation of gene expression by transcription factors and added a new layer of complexity (see Figure 1.3c). Till now, the regulatory pathways are not fully understood and the underlying networks of reciprocative influence seem to be almost unpredictable.

1.3.1 Different types of non-coding RNAs

Different classes of non-coding RNAs are distinguished by their functions, which directly depend on molecular similarities, like the length of the molecule, the composition of their sequences, as well as their secondary structures. ncRNAs tend to fold into completely different secondary structures. Partly, because they need this structure to be functional, partly, because of downstream processing to shorter RNA pieces. The probably best known example of the class that needs structure to be functional are the transfer RNAs (tRNAs). tRNAs fold into their typical cloverleaf structure, which extrudes and thus presents the anti-codon, which is needed to bind to the correct position on the mRNA. Another task of the secondary structure is the processing of longer RNA molecules to shorter, functional RNAs. Here, the class of microRNAs is the probably most famous. Their typical hairpin structure is found by enzymes,
which cut and mature the microRNAs. In the last years, the family of short and long non-coding RNAs was growing fast and researchers are still working on fully understanding their exact functionality. Some of the most important ncRNAs are the following (as described in (Brown, 2006)):

**Ribosomal RNAs (rRNAs)** are the RNA components of ribosomes. The ribosomes construct proteins, using the mRNA as template. rRNAs directly interact with tRNAs during the translation process. rRNAs are the most abundant ncRNAs in a cell. Around 80% of a cell’s RNA consists of them.

**Transfer RNAs (tRNAs)** are involved in protein synthesis. They carry the amino acids to the ribosomes, which assemble them to polymeric molecules, the proteins.

**Small nuclear RNAs (snRNAs)** are small RNAs found in the nucleus of a cell. They are involved in the splicing process, where the introns of a primary transcript are deleted, resulting in the mRNA.

**Small nucleolar RNAs (snoRNAs)** direct enzymes that perform modifications to specific nucleotides of other RNAs. There are two classes of snoRNAs. C/D box snoRNAs, which are linked to methylation and H/ACA box snoRNAs, which are linked to pseudouridylation. This class of small RNAs is also known as guide RNAs, since it guides other enzymes to specific locations.

**MicroRNAs (miRNAs)** are tiny RNAs that regulate gene expression by binding to mRNAs and repressing their translation. microRNAs attract a great deal of attention, since they fine-tune the expression of thousands of genes. Furthermore, by using their endogenous pathway, researchers found a way to turn off (knock-down) specific genes in living organisms. In 2006, for the discovery of this method, called *RNA interference*, the nobel price was awarded.

### 1.3.2 RNA interference and the microRNA pathway

RNA interference (RNAi) was first observed in 1990, when Jorgensen and his group tried to enrich flower pigmentation by overexpressing chalcone synthase and ended up with reduced pigmentation (Napoli et al., 1990; Liu and Paroo, 2010). They poorly understood it and did not know that antisense RNA was the cause. Years later, the groups of Andrew Fire and Craig Mello systematically highlighted the involvement of double-stranded RNA (dsRNA) (Fire et al., 1998). By injecting short dsRNA, which was homologous to the mRNA of a gene called unc-22, they were able to significantly repress its expression resulting in a change of the genes phenotype. The repression using the dsRNA was much better than just using the sense or antisense molecules alone. They named this dsRNA induced silencing method...
RNA interference (RNAi) and the exogenous dsRNAs small interfering RNAs (siRNAs). Since then, this methodology was constantly improved and refined. Nowadays, it is used to perform high-throughput RNAi screenings, to assign phenotypes to specific genes by 'knocking them down' and it also found its way to therapeutic applications. Finally, in 2006, Fire and Mello got the noble price for their work in RNAi.

Until Fire and Mello highlighted the potential of these short RNAs in 1998, only long mRNA molecules were in the focus. Researchers used gel electrophoresis to filter out these longer transcripts, resulting in a complete oversight of the smaller RNA fragments. Knowing about the functionality of the short RNA molecules, the run after the short, expression regulating RNAs started and the recent years resulted in a profound change in our understanding of the regulation of gene expression. Small non-coding RNA especially came into focus as it became clear that they are key players in many cellular processes by post-transcriptionally regulating gene expression via either degradation, translational repression, or both (Kim and Nam, 2006; Lagos-Quintana et al., 2001).

The most prominent candidate of the small ncRNAs are the microRNAs (miRNAs). They are endogenously encoded in many animal and plant genomes (Bartel, 2004; Griffiths-Jones, 2006) and are now recognized to be one of the major regulatory gene families in eukaryotic cells. They are believed to regulate the expression of around one third of all genes in the human genome (Lewis et al., 2005), involved in many fundamental processes like metabolism, development and regulation of the nervous and immune systems (Ouellet et al., 2006; Bagasra and Prilliman, 2004). Furthermore, it has been reported that some microRNAs are actively involved in the development of pathologies like cancer (Lu et al., 2005).

The microRNA pathway (see Figure 1.4) is probably one of the newest and best understood processing pathways. 1993, microRNAs were firstly discovered by the groups of Ruvkun (Wightman et al., 1993) and Ambros (Lee et al., 1993). They found a small RNA (lin-14) that, when being expressed, negatively regulated the production of the LIN-14 protein in C. elegans. LIN-14 encodes a protein whose activity is required for specifying the division timing of specific cells during postembryonic development (Ruvkun and Giusto, 1989). Since lin-4 is only produced in the first larval stage, it temporally decreases the production of LIN-14 and thus controls the developmental-stage timing of this worm. No homologs of this first microRNA were found in other species in further studies. Only in the year 2000 another microRNA was observed. The discovery of let-7 (Reinhart et al., 2000) changed the picture of microRNAs, since it is highly conserved between species. Even in human several homologs were found, showing the immense importance of this small piece of RNA. A new class of small RNAs was born, regulating protein production by complementary RNA-RNA binding to mRNA molecules. First these short RNAs were named small temporal RNAs (stRNAs) (Pasquinelli et al., 2000), but after finding several other candidates with similar functions, they were grouped together and named microRNAs (Lagos-Quintana et al., 2001). Nowadays,
Introduction

several thousands of these small regulatory microRNAs have been identified, building up a huge regulatory network, controlling not only developmental-timing, but influence almost all cellular processes.

microRNAs can be encoded in the genome as independent units, being transcribed by RNA-Polymerase II, or they can occur in introns, being transcribed together with their host genes and then spliced out by the spliceosome (see Figure 1.1c). The latter are called mirtrons and it is thought that ∼40% of all known microRNAs lie in the introns of protein- or non-protein-coding genes (Rodriguez et al., 2004). The transcribed RNA sequence is called primary microRNAs (pri-miRNAs) and directly folds into a stem loop (hairpin) structure, which is typical for microRNAs.

There are also microRNA clusters in the genome, containing up to six microRNA genes, which are regulated and transcribed together, using a common promoter (Altuvia et al., 2005; Lee et al., 2004; Cullen, 2004). Pri-miRNAs encoding microRNA clusters can be several hundred nucleotides long and fold in several stem loop structures with each microRNA hairpin being flanked by a region long enough for efficient downstream processing.

The secondary stem loop structure of each microRNA gene is then found by a protein named ‘DiGeorge Syndrome Critical Region 8’ (DGCR8). DGCR8 is bound to Drosha, a RNase III enzyme that cuts RNA, forming the ‘Microprocessor complex’ (Gregory et al., 2006). Drosha cuts out the hairpins, ending up with precursor microRNAs (pre-miRNAs). One exception here are the mirtrons, which bypass the processing by Drosha, since the spliced out intron automatically folds into a valid pre-miRNA. The pre-miRNAs are around 70nt in length and have a two-nucleotide overhang at their 3’ end. The pre-miRNAs are then exported to the cytoplasm by a protein called Exportin-5, using the two-nucleotide overhang left by Drosha as a docking station (Murchison and Hannon, 2004).

In the cytoplasm, the hairpins are further processed by a RNase III enzyme named Dicer, interacting with the 3’ end of the hairpin and cutting of the loop structure (Lund and Dahlberg, 2006). Dicer cuts, like Drosha, with a two-nucleotide 3’ overhang, resulting in an imperfect double stranded RNA molecule of around 22-24nt in length.

This double stranded RNA is then found by the RNA induced silencing complex (RISC) (McManus et al., 2002), which takes one strand and incorporates it (guide microRNA, or miR), while the other strand is degraded (passenger microRNA, or miR*). The loaded RISC complex uses the miR sequence to find and bind to complementary regions in the mRNA sequence. In human, argonaut proteins within the RISC complex can then, depending on the perfectness of the binding, either cleave the transcript, or recruit additional proteins to repress its translation (Pratt and MacRae, 2009). If the binding is perfect, the mRNA will be directly cleaved by the argonaut Ago2 and degraded (Kawasaki and Taira, 2004). Imperfect bindings result in prevention of translation (Lim et al., 2005) and occur mostly in the 3’UTR part of the mRNAs. One mRNA can be targeted by several microRNAs at a time and the
Figure 1.4: The RNA interference pathway in human. Endogenous microRNAs are encoded in the genome and transcribed as individual units by polymerase II, or they occur in introns, transcribed together with host genes and spliced out. These primary miRNAs folds into typical hairpin structures, which are recognized by the RNase enzyme Drosha and cut out. The resulting precursor miRNA uses Exportin-5 to be transported to the cytoplasm, where it is found by another RNase (Dicer), which cuts the loop, releasing a double-stranded mature miR-miR* miRNA. The latter is loaded to the RNA induced silencing complex and the miR sequence is used to bind to complementary mRNA regions, while the miR* sequence is degraded. mRNAs with miRNA target sites can be, depending on the binding, cleaved and degraded, or post-transcriptionally regulated. mRNAs targeted by several miRNAs show stronger regulatory effects.
down regulation of the protein production seems to correlate with the number of target sites for microRNAs (Rajewsky, 2006; Krek et al., 2005).

In consideration of the fact, that the sequence composition of these tiny RNAs is of great importance for their binding, small differences (e.g. mutations) in the functional molecule can change the expression of hundreds of targeted genes. Thus, the individual sequences of the mature microRNAs have to be deciphered. Till some years ago, this task was achieved by performing a size fractionation with a downstream Sanger sequencing (see Figure 1.5). Since there are not only microRNAs in this size range, but also degradation products of all kinds of different, longer RNAs, it was like fishing in muddy waters and thus a very expensive challenge. The new developed method of high-throughput sequencing (HTS) provides a new technique allowing the measurement of millions of these RNA snippets in short time and low-price. Morin and colleagues (Morin et al., 2008) have shown that experimentally measured miRNA molecules have variations with respect to their genomic encoded sequences. They called this phenomenon isomers and defined four different types. The 5’ end or the 3’ end of the microRNA is elongated or shortened (5’ trimming and 3’ trimming), there are additional nucleotides at the 3’ end (3’ nucleotide addition) or nucleotides of the precursor are post-transcriptionally changed (nucleotide substitution). In the following I will describe the idea of HTS to show, how it can be used to make such findings. I will explain the two most widely used sequencing technologies, then I will shortly explain a protocol which assures, that only molecules in microRNA-like length are sequenced and finally I will go into more detail and show, how these sequences are used for microRNA prediction.

1.4 Sequencing methods

Determining the order of the nucleotide bases A, C, G, and T is known as sequencing. New methods, known as high-throughput sequencing have made it feasible to contemplate sequencing the genomes of hundreds - if not thousands - of species of agronomic, evolutionary, and ecological importance, as well as biomedical interest (Haussler et al., 2009; Dalloul et al., 2010). The main idea behind this method is to shear long DNA sequences to short pieces and read out the nucleotides in a parallelized manner. Using this trick speeds up the sequencing process and makes it thus feasible.

In the following section, I will summarize two different ideas of sequencing DNA. I will try to turn the readers attention to some important characteristics of the different methods, highlighting error sources needed to be handled in downstream analysis. The high-throughput sequencing methods and preparation protocols are intellectual properties of the respective companies and similar wordings of the explanations are indispensable.
1.4.1 A short history about sequencing

Around one hundred years ago the DNA molecule was discovered, and it was soon realized that it is the molecule of heredity. In 1953 Watson and Crick announced the double helical structure of the DNA and thereby set the stage for almost everything that takes place in biomedical research since then. This structure showed researchers that the DNA molecule is a prerequisite for complex biological life. Right after that a series of studies tried to answer basic questions of how this information is used to create the building blocks of cells, leading to the central dogma of molecular biology: Biological information transfers from DNA to RNA, and then to proteins. We now know that it is much more complicated than that, but back in that time, this was a fundamental new finding.

In the late 1970’s Frederick Sanger came up with the idea of using dideoxy nucleotides to sequence DNA (see Figure 1.5). This revolutionary method made it possible to commercialize DNA sequencing. From the early 80’s to the late 90’s, a lot of improvement was done in this method. Then, in the late 90’s, the ‘Human Genome Project’ was set up and the method improved even more. In the end, sequencing the first human genome took around 13 years and cost around $300 million. Interestingly, only one percent of the genome was sequenced after 5 years, highlighting the progress in optimizing the sequencing methodology. Around the year 2000, several companies invented machines that completely automatically sequence DNA. Several such machines were standing in few institutes around the world and produced large amounts of DNA sequence data, improving the first reference genome. But the introduction of the so-called Next Generation Sequencing machines in 2005 changed the world of sequencing. These machines are able to sequence millions of sequences in parallel. The two primary devices

Figure 1.5: Sanger sequencing with fluorescent markers. A DNA polymerase transcribes a template DNA by adding a mixture of normal nucleotides and fluorescent labeled dideoxy nucleotides to the growing chain. When adding a dideoxy nucleotide, the transcription stops, since there is no 3’-OH, which is needed for further incorporation of nucleotides. This method results in fragments of different length. In a final step, the fragments are separated by their length, using a gel-filled capillary and the different light signals at each position are read out, resulting in the templates sequence. Picture redrawn from (Scott, 2004).
Introduction

here are the 'Roche-454 pyrosequencing machine' and the 'Illumina Genome Analyzer'. Both companies frequently improve their machines leading to more sequenced nucleotides, shorter sequencing times and thus lower costs.

1.4.2 454 pyrosequencing

The 454 pyrosequencing method was the first high-throughput sequencing method ready for the markets (Margulies et al., 2005). All sequencing methods start with the preparation of a library. This step is shared by almost all methods and is highly similar in its design. In the beginning the DNA of interest is randomly fragmented to shorter pieces, specific adapters are ligated to both ends and the double-strand is opened. These fragments are then immobilized to a solid surface and amplified. The amplification step is very important since in the downstream sequencing process, nucleotides are incorporated into a growing strand, emitting light signals. In order to intensify these lights and measure them correctly, hundreds of duplicates are needed.

In the 454 pyrosequencing method (see Figure 1.6), the DNA fragments are bound to solid

![Diagram of the 454 Pyrosequencing process](image)

Figure 1.6: 454 Pyrosequencing. (c) Roche Diagnostics. All rights reserved.
1.4 Sequencing methods

beads, covered with sequences complementary to the adapters on the fragments. By washing significantly less DNA molecules than beads, it is statistically assured that not more than one fragment binds to a single bead. The beads are then dispersed in a water-in-oil emulsion. This way, each bead is covered by an oil bubble, creating a sealed environment for DNA amplification. This method ensures that only clones from a unique fragment will be amplified and attached to the bead. The oil-bubble is filled with all reagents, needed for the polymerase chain reaction (PCR) (Bartlett and Stirling, 2003) cycling steps, ending up with hundreds of identical copies of the original fragment. Some cleaning steps are performed, freeing the beads from the oil. For the sequencing, the beads are brought on a picotiter plate. This glass structure has tiny wholes (wells), just big enough for one single bead. This plate is put into the sequencing machine. The top of the picotiter plate allows to load enzymes to the wells, by just flowing the reagents over it. The bottom of the plate is made out of optically clear glass that sits right on top of a high density CCD camera, recording the flashes of almost a million sequencing reactions, as they occur. The four different nucleotides (A, C, G, and T) are sequentially washed over the plate. A DNA polymerase incorporates matching nucleotides, releasing a pyrophosphate moiety, which goes through a series of downstream reactions, that are catalyzed by the enzymes on the beads and the output is light, recorded by the CCD camera. After each cycle, the used nucleotides are washed away, assuring that the signal of the next cycle is triggered by the correct nucleotide. These steps are repeated hundreds of times.

The first four nucleotides of each fragment on the beads form the string 'TCAG', which is called the 'key-sequence'. The sequencing of this string is important, since it returns the signal of a single nucleotide incorporation and is used for calibration. The occurrence of several nucleotides of the same type in a row, known as homopolymers, is a major problem, since the incorporation does not stop after each nucleotide. Thus, homopolymers result in more pyrophosphate and thus a stronger light signal. The intensity of the light is the only way to get information about the length of the homopolymer and rather complicated signal processing steps, using the information of the key-sequence, are necessary. The main advantage of the pyrosequencing method is the length that can be sequenced. With several hundreds of nucleotides, it is very useful for e.g. whole genome assemblies. The sequenced fragments are called reads and stored in a machine readable manner.

The latest machine using this method is the 454 FLX+ machine. While the read length of around 700nt is advantageous, the main drawback of the 454 machines is the relatively small number of parallelized processes, ending up with a small throughput. With 900 mega bases, several runs are needed to get a sufficient coverage for sequencing a complete human genome.
1.4.3 Illumina / Solexa

Like in the 454 pyrosequencing method, the input DNA for the Illumina method (see Figure 1.7) is fragmented and adapter sequences are ligated to the ends of the fragments. The adapter ligated DNA is size fractionated, filtering out fragments between 150 and 200 bases of length, using a gel. This size fractioning step is important, since the length of the fragments sequenced in parallel is fixed and shorter sequences would decrease the number of sequenced nucleotides. Thus, size fractioning assures an optimal throughput.

The adaptor ligated, size fragmented fraction of DNA sequences is called a library and will be used for sequencing. In a next step, the library is washed over a glass flow cell, which is decorated with adapter sequences. DNA oligonucleotides with their adapters, reverse complementary to the adapter sequences on the flow cell, are in this way immobilized on the surface. A low concentration of fragments in the solution washed over the flow cell assures that the fragments bind scattered all around the flow cell in distance to each other. In a process called bridge amplification, the DNA molecules bend over and encounter a complementary second-end primer on the surface. A DNA polymerase creates multiple copies in one place, which results in a collection of millions of copies of the same fragment, called a cluster. The reverse strands are washed away, ending up with a cluster of all fragments bound at the same end. These clusters explain the needed distance of the initial fragments bound to the surface. If the solution contains too many fragments, the cluster density is getting too high and it
1.4 Sequencing methods

is hard to distinguish the signals and if the density is too small, the throughput of the run decreases. Like in the pyrosequencing method, the amplification step is needed to multiply the fragments in order to get a stronger signal. The sequencing chemistry of Illumina sequencing is fundamentally different to the 454 one. All four nucleotides are supplied at each sequencing step. Each nucleotide has its own and unique fluorophore attached, reporting a specific wavelength, when they are scanned by a laser. This way it is possible to obtain the identity of the nucleotide by the specific color. It is possible to add all four nucleotides at once, because the bases have at their 3’ ends a chemical block in place, where normally there is a hydroxyl available for the next base incorporation.

This block does not allow the incorporation until it goes through the detection and deblocking steps of the sequencing. At the detection step, a laser scans the flow cell, stimulating the fluorophor on the incorporated bases, resulting in the release of light, which is recorded by a sensitive camera. This way, all incorporated nucleotides of all clusters at a specific round are measured and stored. Then the fluorescent group is cleaved off and the chemical block is deleted, getting the flow cell ready for the next round. This process is repeated several times, resulting in the complete sequence of all clusters fixed on the flow cell. This method is called dye-terminators technology (Erlich and Higuchi, 1994) and was patented in 2004.

Just to name some numbers: One Illumina flow cell consists of eight lanes, storing around six billion read clusters and is thus able to sequence 600 billion bases in one run. The whole run can be done in around eleven days. Thus, using this technique, it is possible to sequence six complete human genomes with a 30x coverage.

1.4.4 Short RNA-seq

The machines explained above need DNA as input, which technically restricts the sequencing to the genome. But by using cDNA, which is DNA synthesized by reverse transcription using the input RNA as template, researchers found a way to also sequence RNA. The first application of this method was the sequencing of mRNAs in a cell. But, because of microRNAs being substantial regulators, a special protocol to sequence the mature microRNAs was developed. The ∼24nt long RNA pieces regulate mRNAs by binding to its 3’ UTRs and thus the specific sequence of them is of high interest. Since these short regulators are smaller than the sequenced length, no size fragmentation is needed. In the short RNA sequencing (short RNA-seq) protocol, the RNA of a cell is isolated and size fractionated, using a gel. Only these bands of the gel including short RNAs (18-30 nt) are cut out and used as library for the sequencing (see Figure 1.8). In this way, all precursor microRNAs, tRNAs, rRNAs, etc. are discarded, since they are too long. The short mature microRNA molecules pass this gel filter and are sequenced, using a special Illumina protocol, that sequences pieces up to 35 nt in length, speeding up the sequencing process, lowering the sequencing costs. One important note here is the fact, that the main fraction of the short RNA molecules is shorter than the
Introduction

35 nt, sequenced by the machine. Thus, parts of the ligated adapter sequences at the 3' end of the fragments are also sequenced (see Figure 1.9), necessitating the subsequent clipping of this adapters. This clipping step recovers the sequence of the original molecule and is performed computationally in the downstream bioinformatics analysis of the data.

Figure 1.8: Size fractionation using a gel. By cutting out the respective length region, it is possible to select specific kinds of ncRNA classes of interest. For a short RNA-seq experiment, the region for the mature microRNAs is extracted (~18-30nt in length).

1.5 Sequencing data

This section is based on a book chapter written by Steve Hoffmann which explains the basic output formats of high-throughput sequencing and the different approaches to map the identified molecule sequences back to a reference genome (Hoffmann et al., 2011). I modified and shortened some parts and added new fractions to accentuate the application of short RNA-seq data, instead of the much longer DNA data originally used in the book. Since short RNA fragments add new and different problems to the bioinformatics analysis, it is of high importance to go into detail about the format and the mapping.

The bioinformatics tasks start with the process of converting the electromagnetic signals into the correct nucleotides, named base calling. There are base calling approaches coming together with the sequencing machine, but also several different tools with optimized results are available. Here I will not go into detail about the different base callers, since only the company-provided base callers were used. The customized file formats, the complicated mapping procedure, as well as the different sources for errors, nevertheless, have to be explained.
Figure 1.9: When performing short RNA-seq experiments using the Illumina technique, most of the sequenced reads contain parts of the adapter at their 3' end. After size fractionation and cDNA synthesis, 5' and 3' adapters are ligated to the ~18-30nt long double-stranded short RNA pieces. These sequences are then immobilized on the flow cells surface and the Illumina machine performs 35 sequencing circles. For all sequences that are shorter than 35nt in length, the last sequenced fraction consists of a non-cellular adapter sequence (highlighted in red) which has to be clipped in order to receive the original read length.

1.5.1 Data format

The base calling methods assign a quality value to each nucleotide they determine. These numbers reveal the estimated probability of a base being wrong. The output format of the 454 pyrosequencing machine is different to the format of the Illumina sequencer. The 454 machine returns a binary SFF file (Standard Flowgram Format), which can be exported to two multiple FASTA files. One contains the obtained sequences and the other stores the nucleotide-wise quality values. The data coming out of the Illumina sequencing machines are stored in a FASTQ file (Quality FASTA). This modified FASTA file contains not only the sequence information, but also the quality values. The exact calculation of the quality values is still a business secret of the developers.

In the multiple FASTA format each sequence has two entries. The first one is the header line, which starts with the symbol ”>”, followed by the identifier and further sequence information. All following lines, without the header label at the first position, hold the sequence. In an output file of a 454 run, the header contains four columns, a unique identifier, the read length, the coordinates of the bead on picotiter plate and the date of the run.
Introduction

The header lines of the second file are identical, allowing an identification of associated sequence-quality value pairs. The number of quality values below the header line has to be identical to the length of the sequence in the first file, since every single nucleotide got one quality value assigned.

The FASTQ format used by Illumina is in FASTA style, but consists of four, instead of two, entries for each sequence.

The header of the FASTQ format starts with an "@", followed by several information delimited by colons. The stored information is the unique instrument name (HWI-EAS244), the flow cell lane (6), the tile number within the flow cell (1) and the x/y-coordinate of the cluster (5:927). The following lines hold the sequence itself. The third entry starts with the marker "+", and provides space for further information. In most experiments this entry is empty (no text after the "+", but the marker itself has to be available), or it contains the same information as the header field. In the last entry, the quality values are stored. Note that the values are stored in ASCII code, allowing the usage of one character per nucleotide. Compared to the 454 way of storing these numbers, where two character numbers (e.g. 39) have to be delimited by a space, using the ASCII code a great amount of memory can be saved.

The stored quality values are in Phred format. It was developed during the Human Genome Project and is given by

$$Q = -10 \cdot \log_{10} p$$

where $p$ is the probability that the given nucleotide was called incorrectly. It has to be mentioned that the ranges of the quality values have been changed and are still subject to changes. For the Phred score range from 0-62, the ASCII characters from 64 to 126 are used, while for the range from 0-93, the ASCII characters from 33 to 126 are used.

1.5.2 Short read mapping

The short length of the reads obtained by short RNA-seq experiments complicates the search for their locus of origin. To illustrate the problem, a very simple calculation can be made.
The human genome consists of around $3 \cdot 10^9$ base pairs with an alphabet size of four letters (A,C,G and T), each for one of the nucleotides adenine, cytosine, guanine, and thymine. When assuming that the probability $p$ of each nucleotide to occur at each position is 0.25 and the nucleotides are uniformly and randomly composed (which is of course not correct for the human genome, but for reasons of simplicity, we just set it like that), one can easily calculate the minimum length of a DNA sequence expected to occur just once in the human genome. Using the formula

$$E = p^k \cdot n$$

one can calculate the expected number of occurrence of a sequence of length $k$ within a genome of length $n$. When rewriting the term, to bring the length of the sequence to the left side, one gets the following equation:

$$k = \log_p E/n$$

When setting $p = 0.25$, $n = 3 \cdot 10^{-9}$ and the expectation of occurrence $E = 1$, we obtain a length $k \approx 16$. That means, that a sequence of at least 16 nucleotides is needed to statistically assure, that there will not be a second hit in the human genome, just by chance. But it has to be clarified, that the human genome is not random and the nucleotides are not uniformly distributed. Based on experience, we are able to find sequences of length of as little as 15 nucleotides, but a length of at least $\sim 20$ base pairs (bp) is preferred. Another issue is that the sequencing protocols also introduce errors to the sequences, complicating the discovery of its correct position within the reference genome.

Locating the locus of origin is called *mapping* and, based on the problems explained above, is one of the major challenges when working with high-throughput sequencing data. The large size of the genomes, the huge number of short sequences (millions), and the relatively high rate of errors (see 1.5.3), result in the need of sophisticated mapping algorithms. In a standard short RNA-seq experiment, several million molecules are measured. Assuming that the algorithm would need one second to find the mapping position of one sequence (find a $\sim 20$bp long sequence in a $3 \cdot 10^9$ long reference sequence) and we have 1 Mio of these, we would end up in a running time of 11.5 days, which is very inefficient.

As explained above, the errors in the sequences make the finding of the correct position of origin difficult. There can be three different types of errors in a sequenced read: 1) The sequencing machine called a wrong nucleotide, e.g. it measured an adenine, instead of a guanine. The resulting mapping at this position will be called a *mismatch*, since the nucleotide in the reference does not match the sequenced one. 2) The sequencing machine called one extra nucleotide, e.g. two adenines, instead of one, resulting in an *insertion*. 3) Or the machine reads over one nucleotide, resulting in a missing base and thus a *deletion*. 
Another problem of the mapping procedure is the fact that some reads may map to multiple regions within the genome. Especially in ncRNAs this behavior is well known, since e.g. tRNAs or microRNAs occur in multiple copies in one genome, resulting in equally good alignments at all these loci. Thus, the demand for algorithms which reliably returns all possible sites, is mandatory when working with short RNA-seq data.

There are three different types of modern mapping algorithms, one using hash tables (HT) and the others using enhanced suffix arrays (ESA) or the Burrows-Wheeler transform (BWT). In table 1.1, I summarized the most popular mapping tools.

One of the first mapping algorithms was MAQ (Li et al., 2008a). MAQ starts with creating an indexed hash tables holding only the first 28bp of each read (the seed). Each seed is stored in a way that all reads with up to 2 mismatches can be found. Then, in the mapping procedure, every time when MAQ finds a hit of the seed, it extends the locus within the reference genome to discover and score the complete locus. Using the MAQ algorithm, it is not possible to find hits with insertions and deletions (Indels) and no multiple, equally good hits are reported.

BWA (Li and Durbin, 2009), Bowtie2 (Langmead and Salzberg, 2012), and SOAP2 (Li et al., 2009b) are based on the Burrows-Wheeler transform (Burrows and Wheeler, 1994). In a first indexing step, the reference genome is transformed using the BWT. The transformation permutes the order of the characters in a way, that substrings, occurring multiple times, are stored in the transformed string at several places with single characters, repeated multiple times in a row. This way the sequence can be better compressed. As a simple example the text “AANNA$” can be taken. First, all rotations of the text are sorted in alphabetical order, then the last column is stored, ending up with the transformed text “AANNSA” (for a more detailed description see (Burrows and Wheeler, 1994)). In the mapping step, the backward search algorithm (Ferragina and Manzini, 2000) is used. Using two arrays, it can directly access the compressed BWT and simulate a fast traversal of a prefix for the sequence of interest. Since it does not need to load the complete transformation into the memory, it is fast and has a low memory footprint. Nevertheless, to retrieve inexact matches, a time consuming enumeration of all possible mismatches is needed. Even though, tools using the BWT method are very fast in finding exact matches, the speed decreases significantly when searching for loci with >2 errors, or finding all multiple, equally good hits of one read.

The tool segemehl (Hoffmann et al., 2011) is based on enhanced suffix arrays (Abouelhoda et al., 2004). A suffix array (Manber and Myers, 1993; Frakes and Baeza-Yates, 1992) is a simple data structure to store data from suffix trees by creating a sorted list of all suffixes. A suffix tree can implicitly represent all substrings of a given string. A string $S$ consisting of $n$ characters results in a suffix tree of $\leq n$ (edge labels can get compressed) paths from the root to its leaves. Each leaf holds one suffix. ESAs are able to combine the benefits of suffix trees with suffix arrays. While, just like in the suffix tree, an exact search of a string requires
Table 1.1: Popular mapping tools for NGS data. Abbreviations: AL: alignment, S: seed, Y: default, N: not possible y: possible, but not default, ESA: enhanced suffix array, BWT: Burrows-Wheeler-transform, HT: hash table.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Reference</th>
<th>Multiple loci</th>
<th>Mismatches</th>
<th>Indels</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>segemehl</td>
<td>(Hoffmann et al., 2011)</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>SOAP2</td>
<td>(Li et al., 2009b)</td>
<td>y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Bowtie2</td>
<td>(Langmead and Salzberg, 2012)</td>
<td>y</td>
<td>y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>BWA</td>
<td>(Li and Durbin, 2009)</td>
<td>y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>MAQ</td>
<td>(Li et al., 2008a)</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
</tbody>
</table>

only linear time, it needs less memory. Nevertheless, the required memory is still much larger than using the BWT. After creating the ESA in an indexing step, in the mapping, segemehl searches for exact and inexact, matches of all substrings of a read and the reference genome. The two sets of exact and inexact mapping loci is then evaluated by an accurate semi-global alignment using the Myer’s bit vector algorithm (Myers, 1999). By default, segemehl is able to find mapping loci with mismatches, insertions, and deletions not only in the complete alignment, but also in the seed region. Furthermore, it returns all multiple mapping loci of a read, if the number does not exceed a user definable value (default is 100).

All mapping methods have their benefits. BWT allows a very fast and memory efficient mapping, but gets very inefficient when allowing more than two errors, or multiple mapping loci. An ESA, on the other hand, has a big memory footprint, but it is fast when allowing more errors and multiple, equally good hits. Since I am interested in mapping short RNA-seq reads, which are known to occur in multiple copies, the ESA strategy and thus the tool segemehl is used for all our analyses.

### 1.5.3 Error sources

When performing high-throughput sequencing, there are several different sources of errors influencing the feasibility to find the correct loci of origin for the reads. These sources can be divided in three layers: (1) The library preparation, (2) the sequencing process, and (3) the bioinformatics processing.

**Library preparation** During the library preparation, several errors can be inserted. The used sample can be contaminated by external DNA/RNA, leading to a lower coverage or incorrect mappings of exogenous fragments. During the amplification steps, some molecules are preferred by the PCR, resulting in a bias. One example here is the G/C-bias of the Illumina machines. Molecules with a high G/C content are favored and thus the sequenced reads show a higher G/C content than the input sequences. Furthermore, when generating the
clusters, errors in the beginning are passed to all later clones, resulting in wrong mismatches when mapping the reads back to the reference genome.

Sequencing Every sequencing method has its own specific sources of errors. The 454 pyrosequencing for example has problems calling polymers of a single nucleotide, resulting in a multitude of insertions and deletions when mapping the sequenced molecules back to the genome. The Illumina sequencer, since it uses a chemical block, allowing the measurement of one base at a time, shows less insertions and deletions. The error source of the Illumina method is different and explains why it can only sequence rather small molecules, while the 454 machines can sequence hundreds of nucleotides in a row. In theory, the performed cycles of incorporating nucleotides with a chemical block and a fluorophor label, recording the color, deblocking and cleaving the fluorosphor, seems quite straight forward and exact. In practice, the chemicals can be incorrect. There can be bases without the blocking, resulting in the incorporation of more than one nucleotide, or the dye-terminator removal fails. This leads not only to a change in the measured color, but also to a downstream error. This fragment, from now on, is always one base ahead or behind of all the others in the cluster. This molecule is 'out of phase' and decreases the quality value of all subsequent bases. Nucleotides without a fluorophor, or with a fluorophor that was not cleaved correctly, just affect the quality of one single base. All these errors cumulate resulting in the well known drop of the quality values at the 3' ends, limiting the length of fragments that can be sequenced using the Illumina technique.

Bioinformatics processing The bioinformatics processing of the output data starts with the base calling. Normally, the base caller provided by the companies are used, making it impossible to evaluate the error it introduces. During the read mapping, there are several error sources. First of all, the reference genome can be incorrect, or incomplete. Another problem is the alignment of the reads itself. If one read already has errors from the previous steps, it might map to several positions with the same error rate. If that happens, it is no longer possible to tell, where the read originally came from. The errors of the multiple hits can then lie at completely different positions within the alignment. In downstream analysis, these incorrect mismatches, insertions, or deletions can e.g. lead to the incorrect call of mutations. The same problem occurs, if the previous steps did not include any errors, but the sequenced molecule itself shows a mutation that exists in the genome, but the reference genome shows a different allele of it.

1.6 Short RNA-seq and microRNAs

In this section I will shortly explain one application of short RNA-seq. Main parts of this section refer to a book chapter I wrote together with my colleague Jana Hertel (Hertel et al.,
We summarized and explained two different classes of microRNA gene prediction tools, those which use conservation and those which are based on high-throughput sequencing data. I will shortly recap the algorithm procedures for those using HTS, to highlight the rudiment idea behind them.

The traditional experimental approach to measure the expression levels of microRNAs involves cloning and Sanger sequencing. This is an expensive and time-consuming procedure, and as a consequence, relatively little expression data are currently available (Landgraf et al., 2007). Moreover, the huge range of microRNA expression from tens of thousands to just few molecules per cell complicates the detection of microRNAs expressed at low copy numbers. Hence many undetected microRNA may exist even in well-explored species. Then, microRNA expression profiling panels (microarrays) became available for measuring expression levels by means of hybridization. These panels allow a high-throughput detection of microRNA expression. However, they do not allow the detection of new microRNAs. Next generation sequencing platforms like the 'HiSeq 2000' or the 'Genome SequencerTM FLX' became recently available for the sequencing of small RNA molecules, which allows both the detection of expression levels and new microRNA sequences at high speed and sensitivity and low cost. For the detection of new microRNA genes, so called short RNA-seq read patterns are used.

1.6.1 The microRNA pattern

When performing a short RNA sequencing (short RNA-seq) experiment, only short pieces are sequenced and give access to the actual genomic region the RNA molecules arose from. After mapping the short reads back to a reference genome by taking a deeper look to known microRNA loci, it is possible to observe a read pattern generated by the microRNA processing mechanism (explained above). Two high stacks of reads are positioned directly above the annotated miR and miR* regions, showing a specific distance with almost no reads in between (see Figure 1.10). This gap belongs to the loop region which is not seen in the RNA-seq data. The length of the clipped loop (<15 nt) is shorter than the used size fraction of the protocol (~17-28 nt) and thus not sequenced. Most of the currently available methods use this short RNA reads in combination with secondary structure predictions to identify novel miRNA loci.

1.6.2 MicroRNA gene prediction using structure and read patterns

Given the importance of microRNAs in the regulation of gene expression, in the coming years many deep sequencing experiments will be carried out to detect and measure their expression. Therefore, several distinct approaches for miRNA prediction have been employed.

In 2008, miRDeep (Friedländer et al., 2008) was the first tool using high-throughput sequencing data to predict new microRNA candidates. This stand-alone application includes all steps from mapping the reads to a reference genome, clustering consecutive reads occurring in close genomic distance, elongating the region to fetch the whole precursor, calculating
the secondary structure, and determining a probabilistic score. The score is calculated using informations from relative positions of the reads within a predicted hairpin (miR/miR*), the 3’ 2nt overhang of the assumed miR and miR* sequences, as well as secondary structure information, like the minimum free energy (MFE). Unfortunately, miRdeep does not allow errors in the alignment steps and discards reads mapping to multiple loci, loosing all edited microRNAs, as well as miRNAs that have exact copies in the genome. In 2012, a new version of the software, miRDeep2, was published (Friedländer et al., 2012). It identifies microRNAs with better accuracy and shows an improvement of usability.

In 2009, a web server called miRanalyzer (Hackenberg et al., 2009) was released. To predict new microRNAs with this online tool, only the output of a sequencing machine is needed. In contrast to miRDeep, where the user has to download the reference genomes, install several third-party tools, and use the own machine (which might be rather slow), miRanalyzer offers a ‘one-stop-shop solution’. A recent version of miRanalyzer (Hackenberg et al., 2011) supports 34 species. The simplicity of a web server is a big benefit for researchers with no computational background. miRanalyzer uses a random forest machine learning approach to decide if a cluster of consecutive reads (distance <30 nt) belongs to a microRNA, or not. The used set of features consists not only of the relative position of the mapped reads on the hairpin, but also on a wide range of secondary structure informations, like the MFE, the number of paired nucleotides, the number of bulges, and the length of the loop, to name a few. Up to two mapping errors are allowed in the read alignments and the results of the prediction can be downloaded.

There are some other applications that predict microRNAs by using high-throughput sequencing but these use previously published methods for prediction (see table 1.2).
Table 1.2: Available software tools that can be used for the prediction of miRNAs using HTS. The columns list the name of the program (Tool), its reference publication, the set of species that can be handled, and how the authors have made their program available to other researchers. Abbreviations: s.a.: stand alone, ws.: webserver and db.: database.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Ref.</th>
<th>Species</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>miRanalyzer</td>
<td>Hackenberg et al. (2009, 2011)</td>
<td>animal</td>
<td>ws.</td>
</tr>
<tr>
<td>DIANA-mirExTra</td>
<td>Alexiou et al. (2010)</td>
<td>human &amp; mouse</td>
<td>ws.</td>
</tr>
<tr>
<td>miRTRAP</td>
<td>Hendrix et al. (2010)</td>
<td>animal</td>
<td>s.a.</td>
</tr>
<tr>
<td>incRNA</td>
<td>Lu et al. (2011)</td>
<td>animal</td>
<td>s.a.</td>
</tr>
<tr>
<td>mirTools</td>
<td>Zhu et al. (2010)</td>
<td>animal</td>
<td>ws. using miRDeep</td>
</tr>
<tr>
<td>deepBase</td>
<td>Yang et al. (2010)</td>
<td>plant &amp; animal</td>
<td>db. using miRDeep</td>
</tr>
<tr>
<td>DARIO</td>
<td>Fasold et al. (2011)</td>
<td>animal</td>
<td>ws.</td>
</tr>
</tbody>
</table>

1.6.3 MicroRNA-like processing products from other ncRNAs

With the advent of high-throughput sequencing and the possibility to measure millions of molecules within the wished size range, the chase for new microRNA candidates started and several algorithms were trained to predict these (see last section). When running these tools, one of the first observations one will make is the overlap of new predictions with different kinds of ncRNAs, like tRNAs, snoRNAs, and others. Overlaps with other ncRNAs might indicate a false positive prediction. But when taking a deeper look at these candidates, the classification of them makes sense. Furthermore, some tRNAs show high stacks of reads, which can not be explained by any degradation process. There were several publications that also observed these accumulations of reads having microRNA-like length (Miyoshi et al., 2010). Here, I will shortly present the two most prominent types of ncRNAs that can be processed to microRNA-like RNAs, snoRNAs, and tRNAs. snoRNAs are $\sim$70-200 nucleotides long and guide enzymatic modifications of selected rRNA nucleotides (Matera et al., 2007). By using short RNA-seq experiments, it has been shown that H/ACA box snoRNAs, which have two pre-microRNA-like hairpins linked by a hinge (Ender et al., 2008), require Dicer to be processed to short microRNA-like molecules, but are independent of Drosha. These snoRNAs that give rise to microRNA-like sequences, are fully functional as snoRNAs, resulting in two functions, the guidance of rRNA modifications and the regulation of the expression of genes, using their processing products (Ender et al., 2008). A similar picture can be drawn for tRNAs. The group around Lee sequenced short RNA fragments that could be mapped back to tRNA loci (Lee et al., 2009). They called these molecules tRNA-derived RNA fragments (tRFs). When taking the isoleucine tRNA (tRNA-Ile) gene as an example, it has been shown that the sequence of this tRNA has the potential to form not only the typical cloverleaf
structure, but also a \(~100\text{nt}\) long hairpin structure (Babiarz \textit{et al.}, 2008). When mapping the tRFs to the tRNA-Ile sequence with the underlying secondary structure, they map to the 3’ end of the hairpin, explaining the processing by Dicer. Just as the snoRNAs, the tRNAs that can be processed to tRFs are completely functional. These two examples show the importance of this observation. Some tRNAs and snoRNAs are fully functional in their traditional field of duty, but they can also produce microRNAs and thus regulate gene expression. Till now, for most microRNA experiments, all regions overlapping with already annotated loci were filtered out, missing all the microRNA-like molecules. This procedure resulted in a gap within the regulatory network of microRNAs, which has to be closed. There are hundreds of experiments, building up regulatory networks by using microRNA:mRNA bindings, trying to explain differentially expressed genes. Taking microRNA-like RNAs into account might refine the performance.

### 1.7 Computational methods

When performing high-throughput sequencing experiments, the main problem is the amount of reads, which increases constantly. As already explained before, sophisticated algorithms are needed to map short RNA-seq reads back to a reference genome, allowing not only mismatches, but also insertions, deletions, and multiple mapping loci. The computational effort to solve this problem is very high and sometimes it is even impossible to locate the correct origin of a read. Especially, when the reads are too short (<15nt), originate from repetitive regions, or the sequencing process added too many errors, the mapping algorithms reach their limits. The selection of the best algorithm for the respective experiment is important and affects the results of the analysis. In the following, a more detailed description of the here used tool \textit{segemehl} (Hoffmann \textit{et al.}, 2011) is given.

In bioinformatics, machine learning algorithms are widely used for the classification of a variety of things, like genes, proteins, or regulatory interactions. It is a type of artificial intelligence, which “learns” known data and uses this knowledge to classify unknown data. Since the data that is used for the training of the algorithm directly affects the quality of the classification, a well-considered selection is necessary. Nevertheless, due to finite training data, which might change in the future, no guarantees on the performance of machine learning algorithms can be given. Usually probabilistic bounds are used to overcome this problem. In this thesis, a random forest classification (Breiman, 2001) is used and thus it will be explained in this section.

Another method to understand or classify data is hierarchical clustering. Elements are stored in clusters and by using a measure of dissimilarities, it is decided, if clusters are combined or split, depending on the cluster algorithm. The aim is to find a hierarchy of clusters and thus bring some order in the data. Since the calculation of pairwise distances of
1.7 Computational methods

observations in two clusters defines the next step, the used metric (distance between elements) and linkage criteria (distance between clusters) influence the final shape of the clusters. The function of hierarchical clustering, using the `pvclust` algorithm (Suzuki and Shimodaira, 2006), is described in some detail at the end of this chapter.

1.7.1 Short read mapping: segemehl

The tool `segemehl` (Hoffmann et al., 2011) uses an enhanced suffix array (Abouelhoda et al., 2004) to find the original position of a read in a reference genome. The main idea of the `segemehl` approach is the discovery of inexact seeds of maximal possible length, which allows to handle not only mismatches, but also insertions and deletions.

Let us assume an alignment with only two errors to illustrate the idea. If the errors are close to each other, there might be an error free seed at one ends of the read, long enough to find the original position of the read in the reference genome. Another possibility is that the two errors are far apart, allowing the intermediate range to produce perfectly matching seeds. `segemehl` uses this observation by taking all suffixes of a read and trying to find the longest prefixes, i.e. the longest error free match beginning at the first position of the suffix in the reference genome. If the longest prefix occurs only at a few positions, this locus is elongated and a local alignment is used to check, if the mapping of the complete read is good enough. Even though this method works well for most of the reads, the computation of the longest prefix might fail to relocate the correct position with the optimal scoring local alignment. This might be the case, when mismatches, insertions, or deletions keep the longest prefix from being exact enough, resulting in too many possible loci. To overcome that problem, `segemehl` allows a limited number of errors in the seed by enumerating all possible mismatches and indels during the computation of the longest prefix match.

This is done by using the properties of two consecutive suffixes, starting at position $i$ and $i + 1$. It is obvious, that if the suffix starting at position $i$ has a longest prefix of length $l$, the longest prefix of the suffix starting at position $i + 1$ is $l - 1$. And the following suffix has a longest prefix of $l - 2$, and so on. This way, using an enhanced suffix array, it is possible to determine the longest prefix match of the next suffix without rematching the first $l - 1$ characters. At this moment it is also known that the mismatches and indels are in the remaining characters of the suffix in the model. Using this method, it is possible to obtain a set of exact matches and inexact (alternative) matches, together with their positions in the reference genome. These positions, if they are not discarded because of their frequency, are elongated and a semi-global alignment of the complete read and its potential locus of origin is computed, using a Myers’ bit vector algorithm (Myers, 1999). The genomic position is reported, if the fraction of matches within the alignment does not exceed a user-defined value.
1.7.2 Machine learning technique: random forest classifier

A random forest classifier (Breiman, 2001) is made up of a set of trees. Each tree is grown using a form of randomization. The leaves are labeled with the estimates of the posterior distribution over the classes. The internal nodes hold tests which best split the data to be classified. Every unknown element is then classified by running through every tree of the forest, accumulating the reached leaves, ending up with a total sum for each class. The random forest algorithm then simply takes the classification supported by most trees. The randomness can be introduced in two ways: 1) The training data can be subsampled so that every single tree is trained using a different subset, or 2) the node tests can be selected randomly (Bosch et al., 2007).

Each tree is grown as follows (as described in (Breiman, 2001)):

1. If the number of cases in the training set is \( N \), sample \( N \) cases at random - but with replacement, from the original data. This sample will be the training set for growing the tree.

2. If there are \( M \) input variables, a number \( m \ll M \) is specified such that at each node, \( m \) variables are selected at random out of the \( M \) and the best split on these \( m \) is used to split the node. The value of \( m \) is held constant during the forest growing.

3. Each tree is grown to the largest extent possible. There is no pruning.

Breiman (2001) also showed that the error rate of the forest depends on two things:

- The correlation between any two trees in the forest. Increasing the correlation increases the forest error rate.

- The strength of each individual tree in the forest. A tree with a low error rate is a strong classifier. Increasing the strength of the individual trees decreases the forest error rate.

Since, the correlation and the strength are reduced when reducing \( m \) and both are increased when increasing \( m \), an ideal range of \( m \) is somewhere in between. The so-called \( oob \) (out-of-bag) error rate helps to find a value of \( m \) in this range, rendering the need for cross-validation or a separate test set to get an unbiased estimate of the test sets error.

The \( oob \) error rate is calculated as follows (as described in (Breiman, 2001)):

“Each tree is constructed using a different bootstrap sample from the original data. About one-third of the cases are left out of the bootstrap sample and is not used in the construction of the \( k \)th tree. Put each case left out in the construction of the \( k \)th tree down the \( k \)th tree to get a classification. In this way, a test set classification is obtained for each case in about one-third of the trees. At the end of the run, take \( j \) to be the class that got most of the votes
every time case n was oob. The proportion of times that j is not equal to the true class of n averaged over all cases is the oob error estimate. This has proven to be unbiased in many tests.”

One benefit of the random forest algorithm is that it does not overfit (Breiman, 2001). Thus, it is possible to run as many trees as wanted.

### 1.7.3 Hierarchical clustering: pvclust

The pvclust tool (Suzuki and Shimodaira, 2006) is a software package for the environment R (R Development Core Team, 2008). It performs hierarchical cluster analysis and assigns p-values to each cluster via multi scale bootstrap resampling (Efron et al., 1996; Shimodaira, 2002, 2004), indicating how strong the cluster is supported by the data.

Having a set of N elements to be clustered together with an \( N \times N \) distance matrix, the basic procedure of hierarchical clustering (defined by Johnson (1967)) is as follows:

1. Start by assigning each element to a cluster, so that if you have N elements, you now have N clusters, each containing just one element. Let the distances between the clusters the same as the distances between the items they contain.

2. Find the closest (most similar) pair of clusters and merge them into a single cluster, so that now you have one cluster less.

3. Compute distances between the new cluster and each of the old clusters.

4. Repeat steps 2 and 3 until all items are clustered into a single cluster of size N.

The distances between the clusters (step 3) can be calculated in different ways. The three mostly used ones are single-linkage, complete-linkage, and average-linkage clustering:

- In single-linkage clustering the distance of one cluster to another cluster is equal to the shortest distance from any element of one cluster to any element of the other cluster.

- In complete-linkage clustering the distance between one cluster and another cluster is equal to the greatest distance from any member of one cluster to any member of the other cluster.

- In average-linkage clustering the distance between one cluster and another cluster is equal to the average distance from any member of one cluster to any member of the other cluster.

This kind of hierarchical clustering is called agglomerative because it merges clusters iteratively.
As explained in the last chapter, mapped short RNA-seq reads accumulate in two stacks above annotated microRNA loci (see Figure 1.10). While visualizing and reviewing some short RNA-seq experiments from human, we realized, that right upstream and/or downstream of some microRNA loci, there are one or sometimes even two more stacks of reads (see hsa-mir-425 as an example in Figure 2.2). Since these unexpected blocks do not overlap with these from the mature microRNA, but directly start in front or behind them, it seemed clear that they are processed out of a longer pre-microRNA molecule. After reviewing literature, it was clear, that these reads are so-called microRNA-offset-RNAs (moRNAs), that have already been reported in another species. This chapter describes how we systematically analyzed their occurrence in human and it largely follows the resulted discovery note in Bioinformatics (Langenberger et al., 2009).

MicroRNA-offset-RNAs were detected in 2009 as a highly abundant class of small RNAs in a basal chordate. Using short read sequencing data, we showed that moRNAs are also
microRNA-offset RNAs: A new class of RNAs in human

produced from human microRNA precursors, albeit at quite low expression levels. The expression levels of moRNAs are unrelated to those of the associated microRNAs. Surprisingly, microRNA precursors that also show moRNAs are typically evolutionarily old, comprising more than half of the microRNA families that were present in early Bilateria, while evidence for moRNAs was found only for a relative small fraction of microRNA families of recent origin.

2.1 Introduction

In a recent study, Shi et al. (2009) found that in the tunicate *Ciona intestinalis*, half of the identified microRNA (miRNA) loci encode up to four distinct, stable small RNAs. These additional RNAs, termed miRNA-offset RNAs (moRs), are generated from sequences immediately adjacent to mature miR and miR* loci. Like mature miRNAs, they are ∼20nt long, developmentally regulated, and appear to be produced by RNAse III-like processing from the pre-miRNA hairpin. This observation prompted us to specifically search for analogous pattern in human small RNA sequencing libraries.

2.2 Materials and methods

For our analysis we used a, back then, unpublished short RNA-seq dataset from human brain, provided by Philipp Khaitovitch’s group from Shanghai. By now the dataset is available online and can be downloaded from NCBI’s Gene expression Omnibus (GSE18069) (Somel et al., 2010). In detail, total RNA was isolated from the frozen prefrontal cortex tissue using the TRIzol (Invitrogen, USA) protocol with no modifications. Low molecular weight RNA was isolated, ligated to the adapters, amplified and sequenced following the Small RNA Preparation Protocol (Illumina, USA) with no modifications.

Reads were mapped to the human genome (NCBI36.50 Release of July 2008) using *segemehl* (see section 1.7.1). Of the 355 453 reads, 83 093 (23.4%) mapped to miRBase loci (version 12.0). The mapped reads were then sorted by genomic position. Two reads were assigned to the same putative ncRNA locus if they are separated by <39 nt.

Once ncRNA loci were defined, we faced the problem of dividing consecutive reads into blocks to detect specific expression pattern. Note that this task is different from the segmentation of e.g. tiling array profiles (Huber et al., 2006) since we cannot *a priori* restrict ourselves to non-overlapping blocks. Due to biological variability and sequencing inaccuracies, the read arrangement does not always show exact block boundaries. We have developed the *blockbuster* tool that automatically recognizes blocks of reads. In the first step, a mapped read *u* with start and end positions *a*<sub>*u*</sub> and *b*<sub>*u*</sub> is replaced by a Gaussian density *ρ*<sub>*u*</sub> with mean *µ*<sub>*u*</sub> = (*b*<sub>*u*</sub> + *a*<sub>*u*</sub>)<sub>/2</sub> and variance *σ*<sub>*u*</sub><sup>2</sup>. We set *σ*<sub>*u*</sub> = *s*|(*b*<sub>*u*</sub> − *a*<sub>*u*</sub>)<sub>/2</sub>|, where *s* is a parameter that is used to tune the resolution. For each locus, these Gaussian densities are added up separately for the two reading directions. The resulting curves *f*<sup>+</sup> and *f*<sup>−</sup> exhibit pronounced but
2.2 Materials and methods

Figure 2.1: Decomposition of the cluster of reads at the mir-125b-1 locus (Lee et al., 2005) on chr.11 (bottom panel). The blockbuster algorithm replaces each read by a Gaussian profile centered at the midpoint of the read. The middle panel shows the superposition $f(i)$ of these profiles for four different widths of the Gaussian, here chosen to be fraction of the read lengths $L$. Clusters (top) panels are identified as sets of reads whose midpoints are located close to the peaks of $f(i)$. Clusters 2 and 3 correspond to the miR-125b-1 and miR-125b-1*.

smooth peaks centered at blocks of reads with nearly identical midpoints (Fig. 2.1) middle panel. Next we use a greedy procedure to extract the reads that belong to the same block:

1. Determine the location $\hat{x}$ of the highest peak.
   Set $B = \emptyset$ and $\delta = 0$.

2. Include in the block $B$ all reads $u$ such that
   $\hat{x} \in [\mu_u - (\sigma_u + \delta), \mu_u + (\sigma_u + \delta)]$.
   Set $\delta$ to the standard deviation of the $\mu_u$, $u \in B$ and repeat step (2) until not further reads are included in $B$

3. Compute $f_B = \sum_{u \in B} \rho_u$, output $B$, remove the reads in $B$, and set $f \rightarrow f - f_B$.

This procedure iteratively extracts blocks in an order that intuitively corresponds to their importance, Fig. 2.1. Since the area under a peak equals the number of reads in the block the height of the peaks provides a meaningful trade-off between the coherence of a block and its expression level. We therefore suggest to use the height of the peak to define the stop condition for blockbuster. Here, we used $s = 0.5$, a value that requires blocks to be well separated to be recognized as distinct.
We remark that block-detection could alternatively be performed using Gaussian deconvolution approaches, which are commonly used e.g. in chromatography (Vivó-Truyols et al., 2005) and many areas of spectroscopy. For the present application, the additional computational efforts do not seem justified, however. Furthermore, we still would need a heuristic to associate individual reads to peaks.

Since we are concerned only with the known miRNA precursors, we extracted the loci contained in miRBase (version 12) (Griffiths-Jones et al., 2008). Of the 701 miRBase loci, 514 showed evidence for expression in the brain libraries. A custom-track for the UCSC Genome Browser was used for visualization, Fig. 2.2.

As miRNA-offset RNAs are generated from the sequences immediately adjacent to mature miR and/or miR* sequences, the most obvious approach is to search for reads in the flanks of the 514 known microRNA precursors. We used RNAplfold (Bernhart et al., 2006a) with a window size of $L = 120$ to first predict all thermodynamically favorable local base-pairs that are robustly formed by the putative precursor sequence. Then we used an unbranched maximum matching algorithm to determine the position of the hairpin loop.

Among the 701 miRBase loci, 84 showed evidence for expression outside the canonical miR and miR* positions. Of these, we excluded 6 for various different reasons. Two of the annotated microRNAs appear to be snoRNAs: $\text{mir-1248}=\text{HBI-61}$, and $\text{mir-1826}$. The locus annotated as $\text{hsa-mir-1300}$ exhibits a read pattern that is clearly distinct from the expectation.
2.3 Results

Figure 2.3: The sequences overlapping with hsa-mir-1300 cannot form a credible precursor hairpin either at the annotated locus in red or spanning the loci with more abundant reads. In addition most reads in the most expressed cluster are too short for miRs.

For microRNAs, and it does not form a convincing precursor hairpin (see Figure 2.3). The mir-103-2 locus produces short RNAs from both the sense and the anti-sense strand, and hence might be organized differently from typical microRNA loci. The single read near mir-25, a member of one of the human mir-17 clusters (Tanzer and Stadler, 2004), is found about 20nt away from the mature microRNAs, and hence does not match the characteristics of moRNAs.

For the remaining 78 loci with moRNA reads, the cut-point between miRNA and moRNA was determined by visual inspection of the read patterns. Sequences were aligned at the cut-point to investigate potential sequence patterns associated with moRNA processing.

MicroRNAs were classified into families defined by recognizable sequence homology; with a few exceptions, these correspond to the classification of the miRBase. For each family, its evolutionary origin was mapped to the last common ancestor of all species in which a family member could be identified using procedures described in detail by Hertel et al. (2006). This entails a comprehensive homology search and the manual construction of structure-annotated alignments. We analyzed this dataset only for miRNA families in miRBase version 11.0.

In order to check whether moRNA reads are detectable also in other cell types, we analyzed the miRNA “Expression Atlas” (Landgraf et al., 2007), which contains 52,842 small RNA reads of which 12,009 mapped to 467 distinct miRBase loci. Out of these 467 miRNA loci, 218 were also found in our dataset. 23 of these clusters showed evidence for moRNAs with 11 being also predicted as moRNAs in our dataset. Further analysis showed identical patterns for these moRNAs in both libraries (see Table 2.4).

2.3 Results

In the brain libraries we found 78 annotated microRNA loci that exhibit blocks of reads at positions characteristic for moRNAs. For 11 loci, the miRNA Expression Atlas (Landgraf et al., 2007) also contain moRNA reads. In all cases, the reads match perfectly and uniquely to
Figure 2.4: 11 microRNAs from miRBase show identical mRNA patterns in libraries received from two different sequencing methods, ruling out that they are technical artifacts from the next generation sequencing procedures. The blocks from landgraf (sanger sequencing (Landgraf et al., 2007)) are highlighted in blue, these from khaitovitch (illumina sequencing (Somel et al., 2010)) in green and the annotated microRNA precursor sequence from miRBase v.12 (Griffiths-Jones et al., 2008) in red.
Figure 2.5: Statistics of moRNA expression in the brain libraries. **Left.** There is no significant correlation between the number of moRNA reads and the number of miRNA reads from the same locus (Rank Correlation test: $R = 0.1651$, $N = 78$, $p \approx 0.15$). **Right.** Offset RNAs strongly prefer the 5’ arm of the precursor hairpin (top: 153 versus 20 reads, below: 68 versus 14 loci). Only 3 loci have moRNA reads on both sides.

the human genome (hg18), strongly suggesting they are neither technical nor computational artifacts. For 71 of the 78 loci, the moRNAs are conserved together with their miRNA. In contrast to the situation in the urochordate *Ciona intestinalis*, however, human moRNAs appear to be expressed at very low levels. In particular, at least in the brain libraries examined here, moR levels are systematically below the expression levels of miR and miR* reads.

One of the most prominently regulated moRs in *Ciona intestinalis* is miR-219 (Shi *et al.*, 2009). Interestingly, three of its human paralogs also produce clear evidence for offset RNAs (miR-219-2, 3 reads; miR-124-1, 3 reads; and miR-124-2; 4 reads). The offset RNA reads at the 78 human loci share several characteristics with each other and with the moRNAs of urochordates: (1) The moRNA reads are located adjacent to the microRNA reads, and in some cases with only a few nucleotides overlapping the miR or miR*. This conforms with the findings in urochordates and is indicative of processing by a Dicer-like enzyme (Shi *et al.*, 2009). (2) In most cases, the moR sequences are located completely within the predicted hairpin structure. (3) The moRNA reads are located (almost) entirely in a well-conserved region. We observe that there are more than five times as many moRNA reads on the 5’-side of the stem. This is independent of whether the 3’- or 5’-side is predominantly processed into microRNAs. In fact, exactly half of the 78 loci show a prevalence of the 5’-miR, while the 3’-miR is more abundant in the remaining 39 cases. In contrast, the 5’-moRNA is represented...
Figure 2.6: Distribution of the evolutionary age of microRNA families with and without moRNA reads. Data refer to loci listed in miRBase Release 11.0. The lower panel shows the cumulative distribution functions. Kolmogorov-Smirnov tests significantly distinguish the evolutionary age of loci with moRNA reads from the distributions of all expressed families and all families, respectively: moRNA/expressed $D^* = 1.706$, $p = 0.006$; moRNA/all $D^* = 2.433$, $p = 1.5 \cdot 10^{-5}$.

by more reads than the 3’ side in 67 cases (86%). Correspondingly, most of the loci have moRNA reads only on the 5’-side of the precursor hairpin, Fig. 2.5. This prevalence for the 5’-side is independent of the expression patterns of the mature microRNAs. Using Fisher’s exact test, we find that there is no significant association of the 3’/5’ bias in the numbers of moRNA-reads and miR-reads, respectively. There is also no significant correlation between the number of moRNA reads and the expression levels of the corresponding mature miRs (see Fig. 2.5). An investigation of the sequence patterns around the cut-site between microRNA reads and offset reads shows no discernible sequence preferences. Interestingly, there is also no difference in the predicted length of precursor hairpins between microRNAs with and without moRNA reads.

MicroRNA families with offset RNAs are significantly over-represented among the oldest animal microRNAs. In fact, more than half of them originated already in the ancestral bilaterian. Among those that have arisen in Mammalia, again the older ones are more likely to exhibit evidence for moRNAs, Fig. 2.6. The 78 microRNA loci belong to only 54 distinct families. Of these, 4 families show moRNAs in three or more paralogs, and 7 families have two paralogs with evidence for moRNA expression. As almost all miRNA families with multiple
paralogs are evolutionary old, this observation corroborates the association of moRNAs with an early evolutionary origin.

2.4 Discussion

Despite the low level of expression compared to miRNAs, our data strongly suggest that many human pre-miRNAs are processed to produce microRNA-offset RNAs in a systematic way. Several lines of evidence suggest that these transcripts are functional, in particular the extreme level of sequence conservation and the example of miR-219 with moRNAs conserved between human and Ciona. The uncorrelated expression of miRNAs and moRNAs, and the extreme 5' bias of moRNA reads provides evidence that human moRNAs are not just a random by-product of the microRNA processing pathway. The observation that moRNAs sequences are also found by Sanger sequencing rules out that they are technical artifacts from the next generation sequencing procedures. Taken together, our analysis points to a function independent of that of the microRNAs processed from the same locus. These molecules thus may well form a distinct functional class of miRNA-like agents akin to e.g. mirtrons (Berezikov et al., 2007). This conclusion is supported further by the intriguing observation that the majority of miRNAs with moRNA expression are among the evolutionarily oldest families.
CHAPTER 3

Machine learning approach to distinguish block patterns of different ncRNA types

Contents

3.1 Introduction ................................................................. 44
3.2 Materials and methods .................................................... 45
3.3 Results ........................................................................ 47
3.4 Discussion ..................................................................... 51

In a typical short RNA-seq experiments, most of the produced reads (~ 90%) overlap
with annotated microRNAs. Nevertheless, a big fraction of the remaining reads can
be mapped to other ncRNA types, like tRNAs, snoRNAs, Y RNAs, vault RNAs, and many
more. At a first glance, these reads might be just degradation products, but when taking a
deeper look, one can easily find evidence for a more specific processing. We used blockbuster
to simplify the read patterns and compared different classes of ncRNAs. The main observa-
tion we made was the similarity of these patterns within the ncRNA types (see Figure 3.1).
Apparently, we were able to distinguish ncRNAs by just looking at the read patterns. Simple
characteristics, like the number of blocks, the distance between the blocks, or stacks of slightly
shifted blocks, clearly differ between ncRNA classes. Using this information as features, we
utilized a machine learning approach to train and classify three classes of ncRNAs, microR-
NAs, tRNAs and snoRNAs. Using this tool, we were able to predict new candidates of to date
unknown ncRNAs. The following chapter explains the selection of the features, the training of the classification algorithm and shows the performance of the approach. Furthermore, we tried to find explanations for the highly specific patterns by analyzing them. The chapter follows the resulted conference paper (Langenberger et al., 2010), that was presented at the Pacific Symposium on Biocomputing 2010.

3.1 Introduction

Whole-transcriptome analysis of many species and cell types reveals massive expression of non-coding RNA. It is widely believed that non-coding RNAs act as regulators upon transcription and translation. Recent investigations of whole RNA cDNA-libraries based on high-throughput sequencing (HTS) have shown that these libraries contain both primary and processed transcripts. Over the last years, several classes of small RNAs with a length of about 20nt have been discovered. The most prominent classes are miRNAs, piRNAs, and various variants of endogenous siRNAs (Moazed, 2009). In addition, small RNAs have been found to be associated with transcription start and stop sites of mRNAs (Kapranov et al., 2007; Taft et al., 2009c,a). Several studies reported that well-known ncRNA loci are also processed to give rise to small RNAs. MicroRNA precursor hairpins, for instance, are frequently processed to produce additional “off-set RNAs” that appear to function like mature miRs. These moRNAs were discovered in Ciona intestinalis (Shi et al., 2009), where they form an abundant class of processing products. At much lower expression levels they can also be found in the human transcriptome (see chapter 2). Specific cleavage and processing of tRNAs was observed in the fungus Aspergillus fumigatus (Jöchl et al., 2008) and later also found in human short read sequencing data (Kawaji and Hayashizaki, 2008). Small nucleolar RNAs (snoRNAs) are also widely used as a source for specific miRNA-like short RNAs (Ender et al., 2008; Taft et al., 2009b). The same holds true for vault RNAs (Stadler et al., 2009; Persson et al., 2009).

Little is known, however, about the mechanisms of these processing steps and their regulation. Here, we show that the production of short RNAs is correlated with RNA secondary structure and therefore exhibits features that are characteristic for individual ncRNA classes. The specific patterns of mapped HTS reads thus may be suitable to identify and classify the ncRNAs from which they are processed. We explore here to what extent such an approach is feasible in practise. The first step towards this goal is the identification of ncRNA loci from a collection of mapped HTS reads. We have developed the tool blockbuster (described in chapter 2) to simplify this task in genome-wide analyses. The program merges mapped HTS reads into blocks based on their location in the reference genome (Fig. 3.1a-d). After the assembly of blocks, specific block patterns for several ncRNA classes can be observed. For example, miRNAs typically show 2 blocks corresponding to the miR and miR* positions (Fig. 3.1a). A similar processing can be observed for snoRNAs (Fig. 3.1c and d).
hand, tRNAs show more complex block patterns with several overlapping blocks (Fig. 3.1b).

Figure 3.1: Non-coding RNAs exhibit specific block patterns. (a) Distribution of short reads at the hsa-mir-204 locus. There are three clearly distinct blocks of reads: they correspond to moR (5’-end), miR* (center) and miR (3’-end) transcripts. The read distribution pattern is shown below. (b) The class of tRNAs often shows a series of overlapping blocks, while H/ACA box snoRNAs tend to have miRNA-like mature and star blocks at their 5’ and 3’ hairpins with minor overlaps (c). (d) In C/D box snoRNAs most of the reads accumulate at the 5’ end next to the C-box.

### 3.2 Materials and methods

The dataset analyzed here is the same as we used in chapter 2. As before, all small RNAs, 17-28nt long, were mapped to the human genome (NCBI36.50 Release of July 2008) using segemehl (Hoffmann et al., 2009). We required small RNAs to map with an accuracy of at least 80% and only the best hit was selected. Reads mapping multiple times to the genome with an equivalent accuracy were discarded. After filtering the effective accuracy was > 97%. Subsequently, all hits were sorted by their genomic position. Two reads were assigned to the
Machine learning approach to distinguish block patterns of different ncRNA types

same putative ncRNA locus, i.e. cluster, if separated by less than 100nt. Clusters consisting of less than 10 reads were discarded because of their low information content.

Table 3.1: In total 434 of 852 clusters were found within regions of annotated miRNA, tRNA and snoRNA loci. While the average number of blocks is similar for all three ncRNA classes, the number of reads differs significantly among the classes.

<table>
<thead>
<tr>
<th>RNA class</th>
<th>source loci found</th>
<th>blocks/cluster (mean)</th>
<th>reads/cluster (median)</th>
</tr>
</thead>
<tbody>
<tr>
<td>microRNAs</td>
<td>miRBase v12</td>
<td>218</td>
<td>2.42 ± 1.04</td>
</tr>
<tr>
<td>tRNAs</td>
<td>tRNA scan SE</td>
<td>87</td>
<td>3.22 ± 1.92</td>
</tr>
<tr>
<td>snoRNAs</td>
<td>snoRNA base v3</td>
<td>129</td>
<td>2.60 ± 1.66</td>
</tr>
</tbody>
</table>

To detect specific expression patterns, we divided consecutive reads into blocks using blockbuster. Here, we used a width parameter of $s = 0.5$, a value that requires blocks to be well separated to be recognized as distinct. We required a cluster to have at least 2 blocks. In the following we refer to the number of reads comprised in a block as the block height. Using blockbuster, we identified 852 clusters across the whole human genome. This set comprises 2,538 individual blocks and 85,459 unique reads. 434 clusters were found within annotated ncRNA loci [miRBase v12 (727 entries), tRNA scan-SE (588 entries) and snoRNA base v3 (451 entries)], see Tab. 3.1.

Figure 3.2: HTS data reflects structural properties of ncRNAs. Upper panels show the number of 5'-ends of mapped HTS reads (bars) relative to the aligned 5'-ends (dashed vertical lines) of 27 ACA boxes (left), 81 CD boxes (middle) and 87 tRNAs (right). The area in the lower panel represents the number of boxes and tRNAs present at the distance relative to their aligned start sites. In accordance with Taft et al. (Taft et al., 2009b) a sudden and sharp increase of 5'-ends is seen just upstream of the snoRNAs' ACA and C boxes, resp., indicating that read blocks reflect structural properties of snoRNAs. Similarly, the number of 5'-ends increases just upstream of the tRNA and the relative start sites of its three loop regions (dotted lines). Downstream the start sites there is a sudden drop in the number of reads.
3.3 Results

We then computed secondary structures (using RNAfold (Hofacker and Stadler, 2006)) to assess the relationship of reads and structure. For each read, the base pairing probabilities were calculated for the sequences composed of the read itself and 50nt of flanking region both up- and downstream. These data were also collected separately for reads found within annotated miRNA, tRNA, and snoRNA loci, respectively. In order to investigate whether the short reads patterns carry information on the particular ncRNA class from which they originate, we selected three distinct ncRNA classes and performed a random forest classification (Witten and Frank, 2005; Breiman, 2001): tRNAs (n = 87), miRNAs (n = 218) and snoRNAs (n = 129). Based on a visual inspection of the mapped reads, ten features were selected to train the random forest model: the number of blocks within a cluster (blocks), the length of a cluster (length), the number of nucleotides covered by at least two blocks (nt overlap), the number of overlapping blocks (block overlap), the maximum, minimum and the mean block height (max, min and mean height) in a cluster as well as the maximum, minimum and the mean distance between consecutive blocks (max, min and mean distance).

3.3 Results

The 5'-ends of reads arising from known snoRNAs preferentially map just upstream of the C- and ACA-boxes. This indicates the correlation of mapping patterns with processing steps and thus with structural properties of snoRNAs (Fig. 3.2). Based on earlier findings that miRNA-like products are derived from snoRNAs (Taft et al., 2009b) and the observation that miRNA transcripts tend to have higher blocks (Tab. 3.1), the two peaks shown in the Figure 3.2 (left) probably represent small RNAs produced from the 5'- and 3'-hairpins of the HACA (see also Fig. 3.1c). CD-snoRNAs show, in contrast to the HACA-snoRNAs, only a single prominent peak at the 5'-end (Fig. 3.2, middle). An increased number of 5'-ends of HTS reads is also observed just upstream of loops of tRNAs (Fig. 3.2 (right)). The pairing probabilities of bases covered by HTS reads are significantly increased (Fig. 3.3b). Just upstream the 5'-end of these reads, the median base pairing probability increases sharply and reaches a level of > 0.9. At the 3'-end the base pairing probability drops again. However, median base pairing probabilities of bases covered by the center of reads drop down to 70%. Although this effect is boosted by reads found within miRNA loci, it can also be observed unambiguously for reads within snoRNA and tRNA loci (Fig. 3.3a).

The observation that blocks reflect structural properties of ncRNAs was exploited to train a random forest classifier to automatically detect miRNAs, tRNAs and snoRNAs. After visual inspection of block patterns for some representatives of these classes, ten features were selected. Their evaluation reveals significant statistical differences among the chosen ncRNA classes (Fig. 3.4). As expected, the number of reads mapped to miRNA loci (minimum and maximum block height) clearly distinguishes miRNAs from other ncRNA classes. In
Figure 3.3: Base pairing probabilities increase at the 5'-end and decrease at the 3'-end of reads mapped to ncRNA loci. (a) The 3'- and 5'-ends are indicated by dashed lines. The median base pairing probability increases sharply at the 5'-ends (upper left) and drops again at the 3'-ends of reads mapped to miRNA loci (upper right). A similar – but attenuated – effect is observed for snoRNAs (middle panel) and tRNAs (lower panel). (b) The median base pairing probabilities at 5'- (left panel) and 3'- ends (right panel) for all reads within the 852 clusters. The 5'- and 3'-ends are indicated by dashed vertical lines.

Contrast to tRNAs and snoRNAs the maximum block distance of miRNAs shows a very narrow distribution around 40nt, reflecting the distance between miR and miR* transcripts. Furthermore, the class of tRNAs frequently shows more block overlaps than snoRNAs and miRNAs. The distance of blocks is an important feature for snoRNAs: the maximum block as well as the minimum block distance is higher compared to both tRNAs and miRNAs.
3.3 Results

Figure 3.4: Box plots for 8 different features selected to train the random forest classifier. The number of reads mapped to miRNA loci alone (max block height and min block height) effectively distinguish miRNAs from other ncRNAs. Likewise, the distribution of block distances seems to be a specific feature for miRNAs. Compared to other regions, tRNA loci frequently show block overlaps of two or more blocks. The minimum block distance shows a median overlap of $\approx 5$nt for blocks in within tRNA loci. SnoRNAs typically have longer block distances than the other classes.

The random forest model was repeatedly trained with randomly chosen annotated loci and different training set sizes in order to determine positive predictive values (PPV) and recall rates. For the training sets comprising 150 clusters the random forest model shows a
Table 3.2: Positive predictive values (PPV) and recall rates for training sets of size 150 and 250. For each set size means, medians and standard deviations are calculated from 20 randomly sampled training sets.

<table>
<thead>
<tr>
<th></th>
<th>PPV</th>
<th>recall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#loci</td>
<td>mean</td>
</tr>
<tr>
<td>Training size 250</td>
<td></td>
<td></td>
</tr>
<tr>
<td>all</td>
<td>852</td>
<td>0.889</td>
</tr>
<tr>
<td>miRNA</td>
<td>227</td>
<td>0.932</td>
</tr>
<tr>
<td>tRNA</td>
<td>287</td>
<td>0.860</td>
</tr>
<tr>
<td>snoRNA</td>
<td>143</td>
<td>0.819</td>
</tr>
<tr>
<td>other</td>
<td>195</td>
<td></td>
</tr>
<tr>
<td>Training size 150</td>
<td></td>
<td></td>
</tr>
<tr>
<td>all</td>
<td>852</td>
<td>0.827</td>
</tr>
<tr>
<td>miRNA</td>
<td>236</td>
<td>0.900</td>
</tr>
<tr>
<td>tRNA</td>
<td>348</td>
<td>0.755</td>
</tr>
<tr>
<td>snoRNA</td>
<td>115</td>
<td>0.733</td>
</tr>
<tr>
<td>other</td>
<td>153</td>
<td></td>
</tr>
</tbody>
</table>

Positive predictive value $> 0.7$ for all three ncRNA classes. The recall rate for miRNAs is well above 80%. However, with a rate of $\approx 0.55$ the recall of snoRNAs and tRNAs is relatively poor (Tab. 3.2). For larger training sets containing 250 clusters, the positive predictive value (PPV) is $> 0.8$ for all classes. For miRNAs the classification achieves recall rates and PPVs of $> 0.9$. Likewise, the recall rates for snoRNAs and tRNAs rise to 0.7-level. In summary, for both training set sizes and all classes the random forest model achieves PPVs and recall rates of $\approx 0.8$.

We applied the classifier to unannotated ncRNA loci. This resource includes the original reads, their mapping accuracy and their mapping location in machine-readable formats. For microRNAs and snoRNAs, we also indicate whether the candidates are supported by independent ncRNA prediction tools. The 29 miRNA predictions contained 3 miRNAs (hsa-mir1978, hsa-mir-2110, hsa-mir-1974) which have already been annotated in the most recent miRBase release (v.14), as well as a novel member of the mir-548 family, and another locus is the human ortholog of the bovine mir-2355. In addition, we found two clusters antisense to annotated miRNA loci (hsa-mir-219-2 and hsa-mir-625). Such antisense transcripts at known miRNA loci have been reported also in several previous publications (Glazov et al., 2009; Bender, 2008; Stark et al., 2008; Tyler et al., 2008), lending further credibility to these predictions.

For the tRNAs and snoRNAs we expect a rather large false positive rate. The 78 tRNA predictions are indeed contaminated by rRNA fragments, but also contain interesting loci, such as sequence on Chr.10 that is identical with the mitochondrial tRNA-Ser. SnoReport (Hertel et al., 2008), a specific predictor for HACA snoRNAs based on sequence and secondary features, recognizes 44 (20%) of our 223 snoRNAs predictions. Short RNAs are processed...
from virtually all structured ncRNAs. Complex read patterns are observed, for instance, for the 7SL (SRP) RNA and the U2 snRNA. Y RNAs, which have a panhandle-like secondary structure produce short reads mostly from their 5’ and 3’ ends, see Fig. 3.5.

![Diagram of short reads production from various ncRNAs](image)

Figure 3.5: Short reads are produced from a wide variety of structured ncRNAs. Green arrows indicate the ncRNA gene and its reading direction, individual short reads are shown as orange lines. The same scale is used for all examples.

### 3.4 Discussion

In extension of previous work establishing that various ncRNA families produce short processing products of defined length (Kawaji et al., 2008; Shi et al., 2009; Taft et al., 2009b), we show here that these short RNAs are generated from highly specific loci. The dominating majority of reads from short RNAs originates from base paired regions, suggesting that these RNAs are, like miRNAs, produced by Dicer or other specific RNAases. For example, specific cleavage products have recently been reported for tRNAs (Thompson and Parker, 2009). In this work we show that the block patterns are characteristic for three different ncRNA classes and thus suitable to recognize additional members of these classes. For instance, the random forest trained with loci annotated in the mirBase v12 predicted five additional miRNAs reported in the mirBase release 14 as well as two “antisense microRNA”. The block patterns for the evaluated ncRNAs show some interesting characteristics. Although miRNA loci accumulate far more reads than tRNAs and snoRNA loci, the reads are extremely unevenly distributed across the blocks. For tRNAs we observe series of overlapping blocks that are specific enough to separate this class from other classes with high positive predictive values.
Machine learning approach to distinguish block patterns of different ncRNA types

However, the successful prediction of miRNAs heavily depends on the height of the blocks, i.e. the number of reads that map to a potential locus. In comparison tRNAs and snoRNAs show lower positive predictive values and recall rates. A relatively large training set is required to achieve PPV’s > 80%. Obviously, the selection of appropriate features is crucial for the success of the presented approach. It has to be mentioned that there is still room to improve the random forest classifier by adding other characteristic features. The integration of secondary structure information of cluster regions is likely to enhance the prediction quality. Beyond the classification by means of soft computing methods, this survey shows that HTS block patterns bear the potential to greatly improve and simplify ncRNA annotation. Given the striking relationship of HTS reads and secondary structure for some ncRNA classes, block patterns may also be used in the future to directly infer secondary structure properties of non-coding RNAs from transcriptome sequencing data. In this context, although not shown here, block patterns may also help to identify new classes of RNAs directly from transcriptome sequencing data.
DARIO: a ncRNA detection and analysis tool for next-generation sequencing experiments

Contents

4.1 Introduction ................................................. 54
4.2 Materials and methods .................................... 55
  4.2.1 Workflow ............................................ 55
  4.2.2 Sequence and annotation data ....................... 56
  4.2.3 Webserver implementation ............................ 56
4.3 Results and discussion ................................... 57
  4.3.1 Input format ........................................ 57
  4.3.2 Quality control ..................................... 57
  4.3.3 RNA quantification .................................. 58
  4.3.4 Classification ...................................... 59
4.4 Conclusion ............................................... 61

To make the classification algorithm described in chapter 3 available to the research community, we designed an easy to use web server. Interested parties can upload their short RNA-seq data and receive the results of an automated analysis, together with new predictions. The analysis contains quality control steps and quantification of all known ncRNAs. All loci (for already annotated and predicted ncRNAs) come together with a link
to the UCSC genome browser, in which the researcher can take a deeper look on the region of interest. A density track of all mapped reads is provided to visualize the uploaded data in the browser. The web service is called DARIO and is free to use. It can be accessed at http://dario.bioinf.uni-leipzig.de/. The following chapter explains the backend workflow of the service, evaluates the results and follows its publication in the Nucleic Acids Research web server issue 2011 (Fasold et al., 2011).

4.1 Introduction

High-throughput sequencing (HTS) using a small RNA preparation protocol (small RNA-seq) was primarily designed to measure the expression of microRNAs. Closer inspection of the resulting sequence libraries, however, revealed that many other ncRNA types are chopped into RNA molecules of microRNA-like length, and are hence detectable in the sequencing data as well (see chapter 3). Some of the non-miRNA sources of short RNA sequences include tRNAs (tRNA-derived fragments) (Haussecker et al., 2010; Lee et al., 2009; Cole et al., 2009), snoRNAs (snoRNA-derived small RNAs) (Taft et al., 2009b), 21U-RNAs (Ruby et al., 2006) or snRNAs (Langenberger et al., 2010). As shown in chapter 2, small RNA sequencing has helped to identify new RNA species such as microRNA offset RNAs (moRs), which derive from miRNA precursors. Although they have first been described in the simple chordate Ciona intestinalis (Shi et al., 2009), they could be verified in mammalian transcriptomes (Langenberger et al., 2009) and have later been linked to Kaposi’s sarcoma associated Herpesvirus (Umbach and Cullen, 2010; Lin et al., 2010). Hence, small RNA-seq data contains a plethora of processing and maturation products potentially including yet unknown RNA species. Despite this fact, many small RNA-seq data analysis tools such as miRanalyzer (Hackenberg et al., 2009), miRDeep (Friedländer et al., 2008) or miRNAkey (Ronen et al., 2010) focus on microRNAs – largely neglecting other types of RNAs. In addition, these programs are often restricted to specific sequencing platforms due to embedded mapping algorithms. Other tools such as deepBase do not allow the upload of own experimental data (Yang et al., 2010), or can only analyze data from prokaryotes, like nocoRNAc (Herbig and Nieselt, 2011).

In addition to finding new RNA species, the expression levels of ncRNAs have been shown to be associated with a number of different phenotypes. Various forms of neoplastic diseases such as colorectal cancer (Lanza et al., 2007), for instance, show changes in miRNA expression levels. Likewise, differential snoRNA expression has been found in a study with menigioma cells (Chang et al., 2002). RNA quantification is possible using tools such as rQuant.web (Bohnert and Rätsch, 2010) or RSEQTools (Habegger et al., 2011), however, they are not readily applicable to small ncRNA analysis as annotation data must be collected from different sources.

We have combined the ncRNA prediction method described in the last chapter (Langen-
berger et al., 2010, 2009) with tools to quantify ncRNAs in a completely platform independent and easy to use web tool. DARIO performs RNA-seq quality controls and quantifies RNA expression based on annotated ncRNAs from different ncRNA databases. The expression data and ncRNA predictions can be downloaded in the standardized BED format. We provide a script to locally convert SAM files and other mapping files to the BED format. The script is optimized to greatly reduce the amount of data that has to be uploaded to the DARIO server.

4.2 Materials and methods

4.2.1 Workflow

The DARIO web service requires previously mapped reads stored in compressed or uncompressed files in BAM or BED format. The uploaded file is uncompressed, if necessary, and

Figure 4.1: Simplified workflow of a DARIO computation. After the user upload the data is run through some quality checks with regard to read length distributions and multiple mappings. Subsequently, the mapping loci are overlapped with ncRNA annotation data for gene expression measuring. A random forest classifier predicts new ncRNAs. The results of the analysis are easily accessible from a summary web page.
examined for validity. A first analysis of the input data provides measures for quality control. The reads are then overlapped with various gene models of the selected species relevant for the analysis of small ncRNAs. Mapping loci overlapping with exonic regions are excluded from further analysis. Mapping loci overlapping with introns and intergenic regions are used to predict non-annotated ncRNAs. Finally, the results are summarized in HTML pages and data tables. A simplified workflow of the DARIO web service is depicted in Figure 4.1.

4.2.2 Sequence and annotation data

Genome assemblies of six supported species were downloaded from the UCSC Genome Browser (http://hgdownload.cse.ucsc.edu/downloads.html): Homo sapiens (hg18, NCBI 36.1 and hg19, GRCh37), Macaca mulatta (rheMac2, MGSC Merged 1.0), Mus musculus (mm9, NCBI37), Danio rerio (danRer7, Zv9), Drosophila melanogaster (dm3, BDGP Release 5) and Caenorhabditis elegans (ce6, WUSTL School of Medicine GSC and Sanger Institute version WS190). For each assembly we retrieved the UCSC Known Genes Track using the UCSC Table Browser in order to generate intron/exon lists.

cmpRNA annotation was collected from several databases. While miRNA annotation was obtained from the miRBase v16 (Griffiths-Jones et al., 2008), most of the other ncRNA loci were downloaded from the UCSC Genome Browser. For human ncRNA data sets, we additionally included tRNA track (Chan and Lowe, 2009), wgRNA track (Lestradé and Weber, 2006) for snoRNAs and the rnaGene track for other ncRNAs. For mouse, the tRNA track was used. For fly, our annotation encompasses the flyBaseNoncoding track from FlyBase (Crosby et al., 2007). The sangerRnaGgene track containing WormBase annotations (Harris et al., 2010) is provided for worm ncRNA data analysis. Where necessary, annotations were lifted to alternative assemblies with the UCSC tool liftover (http://hgdownload.cse.ucsc.edu/downloads.html). Additional ncRNA annotations were collected from the Mouse Genome Database (Blake et al., 2003) as well as from Ensembl/BioMart for zebrafish (Smedley et al., 2009). If tRNA or snoRNA annotations were not available, we predicted candidates using tRNAscan-SE (Schattnier et al., 2005) or snoReport (Hertel et al., 2008), respectively.

4.2.3 Webserver implementation

The website and the HTML results are created by a set of Python scripts and the Mako template engine. The jobs are scheduled in a queued fashion and distributed over a set of active machines. Upon completion, the results are transferred to the web server and available under a personalized link for four weeks. Mapping loci are merged to blocks based on their genomic positions and assembled to regions of blocks using blockbuster v1.0 (Langenberger et al., 2009) with default parameters. These are then classified using the random forest method in WEKA v3.6 (Langenberger et al., 2010; Breiman, 2001; Hall et al., 2009). Graphics are created using R (R Development Core Team, 2008) and the ggplot2 graphics package.
RNAs. RNAz Version 1.0 (Washietl et al., 2005) has been used to screen all supported assemblies for potential functional RNA structures. Predicted ncRNA candidates are overlapped with these screenings to provide RNAz screenings.

4.3 Results and discussion

The DARIO website provides a simple web form that allows the user to specify and upload input data. The website currently supports seven assemblies of six species: human (hg18, hg19), rhesus monkey (rheMac2), mouse (mm9), fruit fly (dm3), worm (ce6) and zebrafish (danRer6). After file upload, a job is created and queued for computation. The user may supply an email address to be notified upon job completion. A single job typically takes between 5 and 30 minutes. The results are summarized on a single web page containing job details, quality control measures and figures, ncRNA quantification and classification. All results can be downloaded for further analysis.

4.3.1 Input format

DARIO uses mapped sequences as input. The alignments may be provided in the common BAM or BED formats (http://genome.ucsc.edu/FAQ/FAQformat.html). The BED files require the fields for sequence identifier, strand and need to provide the read count in the score field. This format allows to collapse reads occurring multiple times into unique sequence tags, dramatically reducing space requirements of sequencing data. DARIO allows upload of (g)ZIPed files.

We provide a small, no-dependency perl script to convert SAM and SOAP format files into the BED input format. Virtually all common mapping tools (segemehl, BWA, SOAP, Bowtie etc.) can write their output alignment to either of these formats.

Using genome loci of previously mapped reads, and thus decoupling read alignment and analysis, has a number of advantages over using raw sequence reads. First, DARIO has no dependencies to any sequencing platform or mapping tool. Thus read data originating from any sequencing platform and aligned with any mapping program can be used. Second, this greatly reduces the required amount of data to be uploaded to the server (e.g. 1GB SAM file → 15MB compressed BED file).

4.3.2 Quality control

There are numerous errors and biases that can occur during sample handling, library preparation and sequencing in a small RNA-seq experiment, rendering an assessment of the experiments quality a necessity (Dohm et al., 2008; Linsen et al., 2009; Hansen et al., 2010). A basic set of figures (see Figure 4.2) gives the user a first impression of the quality of the experiment. This includes the read length distribution, the number and occurrence of multiple mapped
Quality Control

The following figures indicate whether problems might have occurred during sample or library preparation.

Figure 4.2: The DARIO web server provides a set of graphics for quality control. The figures provide information on the read length distribution, the number of multiple mappings, the distribution of read hits across the genome and the annotated non-coding RNAs. The user may immediately check the success of his short RNA sequencing run in terms of capturing the ncRNA of interest.

reads, the fraction of reads mapping to different genomic loci (exon, intron or intergenic) and ncRNA classes (miRNA, tRNA, snoRNA, etc.). Other measures include the number of mappable reads and the number of tags.

4.3.3 RNA quantification

For expression analysis mapping loci are overlapped with annotated ncRNAs from a variety of sources. To handle multiple mappings, the number of reads for each sequence tag is divided by the number of mapping loci. This normalized expression value is assigned to each mapping locus. These expression values are additionally normalized based on the absolute number of mappable reads (RPM), to allow subsequent differential expression analysis. Note that these measures do not necessarily reflect precursor ncRNA abundance as RNA processing
4.3 Results and discussion

Figure 4.3: The DARIO analysis output is partitioned into different ncRNA classes. For each ncRNA class a list that may be sorted by location, name or expression criteria is provided. A link to the UCSC genome browser allows the instantaneous inspection of the ncRNAs.

and sequencing protocol lead to a non-uniform read distribution across the precursor RNA.

A list of expressed ncRNAs, itemized by ncRNA classes, is generated (see Figure 4.3). The user obtains information about the normalized expression, the number of mapped reads (raw and multi-map-normalized), as well as a link to the UCSC genome browser for each expressed locus. The UCSC link helps the experimenter to quickly scan the data for new types of ncRNAs, e.g. microRNA-offset-RNAs (moRs) or vault RNAs, and to get a deeper understanding of the processing of these poorly understood ncRNA classes.

The web interface allows the upload of own annotation tracks. The specified regions are included in all downstream analysis. Predicted RNAs from previous DARIO runs can directly be used as user annotation.

4.3.4 Classification

DARIO predicts new ncRNAs using a the developed machine learning approach explained in chapter 3 (Langenberger et al., 2010). This method relies on characteristic read patterns exhibited by different classes of ncRNA. The classifier achieves positive predictive values (PPVs) and recall rates of ~0.8. With recall rates varying from 0.6 to 0.7 and PPVs between 0.7 to 0.8 snoRNA predictions mark the lower bound of the classification (see Tab. 3.2). Receiver operator characteristic curves for all predicted ncRNAs in a number of species is shown in Figure 4.5. For each candidate, a prediction score is given along with a RNAz classification (Washietl et al., 2005), if available. One of the candidate miRNAs predicted on the human chromosome 8 using the DARIO platform is shown in Figure 4.4. With the links to the UCSC genome browser it is possible to instantaneously inspect the prediction by loading multiple different annotation tracks.
DARIO: a ncRNA detection and analysis tool for next-generation sequencing experiments

Figure 4.4: Example for a DARIO prediction for a miRNA. The integrated random forest classifier predicts a miRNA on the human chromosome 8 in an intergenic region. The expression pattern shows a typical miR and miR* processing product constellation. The UCSC browser reports no annotations for known ncRNAs at this position.

Figure 4.5: Receiver-Operator characteristics (ROC) curves for the random forest ncRNA prediction in H. sapiens (solid blue), M. mulatta (solid red), M. musculus (solid green) and C. elegans (solid yellow) used by the DARIO web server. Prediction cut-off points are shown for the classifications in humans (dashed blue) as described in (Langenberger et al., 2010) for a small randomly selected training set (cf. Table 2 in (Langenberger et al., 2010)). The classifier has a true positive rate well above 80% for miRNA (A) and above 60% for tRNA (B) predictions in humans while the false positive rate is below the 12% and 3% level, respectively. Predictions for snoRNAs (C) are worse. The cut-off points where chosen to maintain a low false positive rate of approximately 6%. With the exception of snoRNAs in worms the classification of all ncRNA classes in other species is slightly more accurate.
4.4 Conclusion

We have developed the first integrated tool for the analysis and prediction of various small ncRNAs on user-provided RNA-seq data. The web service allows researchers to quickly grasp and assess the success of a short RNA-seq experiment. The web server overlaps the mapping loci with ncRNA genes from a number of ncRNA classes and annotation databases in order to quantify RNA abundance with different expression measures. Reads that do not map to annotated ncRNA genes are identified and classified. DARIO provides an easy to use web interface and thus greatly facilitates both initial evaluation and downstream analysis of read data originating from arbitrary sequencing platforms. Further versions of DARIO will allow to directly compare sets of small RNA transcriptomes to evaluate differences in expression levels of ncRNAs.
One main insight gained in the last chapters was the existence of ncRNA specific read patterns. It seems that not only microRNAs, but also other ncRNA classes are processed in vivo. The simplest explanation here would be the existence of a secondary structure, namely a hairpin, which is found and cut by Dicer. The microRNA pathway is well known and it results in molecules of the measured length. As already mentioned before, it is well known, that tRNAs and snoRNAs can be processed by Dicer, resulting in short microRNA-like molecules. This knowledge results in the quite obvious problem, that there might be a lot of wrong, or multiple annotated ncRNAs in the public databases. Since a cut by Dicer results in a similar read pattern and the hairpin structure is needed, microRNA gene prediction tools, which use this information, will mis-annotate these loci.

The avalanche of next generation sequencing data has led to a rapid increase of annotated microRNAs in the last few years. Many of them are specific to individual species or rather narrow clades. We took a closer look at the current version of miRBase, and showed, that
dozens of entries conflict with other ncRNAs, in particular snoRNAs. With few exceptions, these cases show little similarities to canonical microRNAs, however, and thus they should be considered as mis-annotations. The next chapter is based on the conference paper for the Brazilian Symposium on Bioinformatics 2012 (Langenberger et al., 2011).

5.1 Introduction

MicroRNAs and small nucleolar RNAs are thought of distinct classes of ncRNAs with very different functions. While microRNAs are matured to ∼20nt sequences that direct post-transcriptional gene silencing, snoRNAs canonically guide, in their complete form, the chemical modification of mostly rRNAs and snRNAs (Terns and Terns, 2002). On the other hand, high-throughput sequencing studies revealed that snoRNAs are a prolific source of sequence fragments of microRNA size (Kawaji et al., 2008; Taft et al., 2009b; Langenberger et al., 2010; Brameier et al., 2011), termed sdRNAs. At least some of these snoRNA-derived small RNAs, similar to microRNAs, interact human Argonaut and affect gene expression (Ender et al., 2008). Recently, efficient gene silencing has been demonstrated for 11 small RNAs derived from box C/D sno-miRNA (Brameier et al., 2011). Similar short RNAs, in a few cases with validated functions in gene silencing, are also produced from most other well-known structured RNAs including Y RNA (Langenberger et al., 2010; Meiri et al., 2010), vault RNAs (Stadler et al., 2009; Persson et al., 2009; Mosig and Stadler, 2011), snRNAs (Langenberger et al., 2010), and tRNAs (Lee et al., 2009; Cole et al., 2009; Haussecker et al., 2010; Findeiß et al., 2011). Recent work (Schopman et al., 2010), furthermore, cast doubt on the microRNA nature of several short RNA products that likely originate from the 3'-end of matured tRNAs since they include the post-transcriptionally append CCA tail. The large numbers of CCA-tagged reads from nearly all tRNAs, which are abundant in deep sequencing data, supports a tRNA-origin of a few annotated “microRNAs”. More information about CCA tails in deep sequencing data will be presented in chapter 8.

Canonical microRNAs are generated from a quite specific processing pathway (Miyoshi et al., 2010): a polymerase II transcript, the primary miRNA precursor (pri-miRNA) is cropped by the Drosha DGCR8 complex, also known as Microprocessor. The resulting pre-microRNA hairpin uses the exportin-5 pathway to reach the cytoplasm, where it is cleaved to generate the mature miRNA. Early reports (Borchert et al., 2006) of pre-microRNAs originating from pol-III transcription have recently been refuted (Bortolin-Cavaillé et al., 2009). A survey of human pol-III transcription (Canella et al., 2010), furthermore, recovered no annotated microRNA except two mis-annotations: a vault RNA (hsa-mir-886) and the Y5 RNA (hsa-mir-1975). Mirtrons, on the other hand, are short introns forming stable hairpin structures (Berezikov et al., 2007; Okamura et al., 2007; Ruby et al., 2007; Chung et al., 2011). Both ends of mirtrons are defined by the splice sites. A related, mirtron-like source of small
RNAs requires both splicing and exosome-mediated trimming to extract the pre-microRNA hairpin (Flynt et al., 2010; Chong et al., 2010). In this case only one end of the precursor hairpin is defined by the splicing reaction. The production of small RNAs from these intronic precursors is independent of Drosha (Ruby et al., 2007; Chong et al., 2010). A recent review (Miyoshi et al., 2010) lists several additional esoteric pathways, including at least two of them independent of both Drosha and Dicer.

Table 5.1: Overlap of annotation as microRNA and other ncRNA classes. Classification probabilities for microRNAs and snoRNAs are listed. SVM refers to the analysis described in section 3.

<table>
<thead>
<tr>
<th>miRBase</th>
<th>PMID</th>
<th>RNA class</th>
<th>RNA-snoReport</th>
<th>SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>gga-miR-3528</td>
<td>19891781</td>
<td>SNORA17</td>
<td>0.99</td>
<td>0.94</td>
</tr>
<tr>
<td>cfa-miR-1836</td>
<td>18392026</td>
<td>SNORA20</td>
<td>0</td>
<td>0.99</td>
</tr>
<tr>
<td>bta-miR-2427</td>
<td>19633723</td>
<td>SNORA25</td>
<td>0.82</td>
<td>0</td>
</tr>
<tr>
<td>hsa-miR-664</td>
<td>20413612</td>
<td>SNORA36</td>
<td>0.64</td>
<td>0.86</td>
</tr>
<tr>
<td>mmm-miR-1940</td>
<td>18849523</td>
<td>SNORA47</td>
<td>0.97</td>
<td>0</td>
</tr>
<tr>
<td>bta-miR-2311</td>
<td>19633723</td>
<td>SNORA61</td>
<td>0</td>
<td>0.60</td>
</tr>
<tr>
<td>mdo-miR-1543</td>
<td>17965199</td>
<td>SNORA74</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>hsa-miR-2424</td>
<td>18285502</td>
<td>SNORA81</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>hsa-miR-3651</td>
<td>20489314</td>
<td>SNORA84</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>hsa-miR-1291</td>
<td>18285502</td>
<td>SNORA34</td>
<td>0.81</td>
<td>0.97</td>
</tr>
<tr>
<td>tgu-miR-2089</td>
<td>20360741</td>
<td>SNORD74</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>mmm-miR-3096</td>
<td>20413612</td>
<td>SNORD93</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>gga-miR-3535</td>
<td>19891781</td>
<td>SNORD20</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>gga-miR-3538</td>
<td>19891781</td>
<td>SNORD38B</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>gga-miR-1454</td>
<td>18430245</td>
<td>SNORD100</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>hsa-miR-3647</td>
<td>20489314</td>
<td>SNORD11B</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>hsa-miR-3653</td>
<td>20489314</td>
<td>SNORD125</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>hsa-miR-1201 †</td>
<td>19633723</td>
<td>SNORD126</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>mIR-1843 * †</td>
<td>20413612</td>
<td>SCARNAl0</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>oan-miR-1354</td>
<td>18463306</td>
<td>snoU85</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>bta-miR-2424</td>
<td>19633723</td>
<td>SCARNAl2</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>mmm-miR-3069</td>
<td>20413612</td>
<td>SCARNAl3</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>oan-miR-1348</td>
<td>18463306</td>
<td>SCARNAl5</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>gga-miR-3540</td>
<td>19891781</td>
<td>SCARNAl6</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>bta-miR-1940</td>
<td>19633723</td>
<td>SCARNAl7</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>dre-miR-735</td>
<td>16698962</td>
<td>Y RNA (Mosig et al., 2007)</td>
<td>–</td>
<td>0</td>
</tr>
<tr>
<td>hsa-mir-1975 †</td>
<td>16698962</td>
<td>Y5 RNA (Meiri et al., 2010)</td>
<td>–</td>
<td>0</td>
</tr>
<tr>
<td>dre-miR-733</td>
<td>16698962</td>
<td>vault RNA (Stadler et al., 2009)</td>
<td>–</td>
<td>0</td>
</tr>
<tr>
<td>mir-866</td>
<td>16698962</td>
<td>VTRNA2 (Stadler et al., 2009)</td>
<td>–</td>
<td>0</td>
</tr>
<tr>
<td>mmm-miR-699 †</td>
<td>18285502</td>
<td>RNase MRP</td>
<td>–</td>
<td>0</td>
</tr>
<tr>
<td>hsa-miR-1246</td>
<td>18285502</td>
<td>U2</td>
<td>0.95</td>
<td>0</td>
</tr>
<tr>
<td>hsa-mir-1274</td>
<td>18285502</td>
<td>tRNA-Lys (Schopman et al., 2010)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>hsa-mir-1280</td>
<td>18285502</td>
<td>tRNA-Leu (Schopman et al., 2010)</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>miR-720</td>
<td>16582102</td>
<td>tRNA-Thr (Schopman et al., 2010)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>hsa-mir-1308</td>
<td>18849523</td>
<td>tRNA-Gly (Schopman et al., 2010)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mmm-miR-1937b</td>
<td>18849523</td>
<td>tRNA-Pro (Schopman et al., 2010)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>hsa-mir-151</td>
<td>–</td>
<td>–</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>hsa-mir-215</td>
<td>–</td>
<td>–</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>hsa-mir-140</td>
<td>–</td>
<td>–</td>
<td>0.99</td>
<td>0</td>
</tr>
<tr>
<td>hsa-mir-605</td>
<td>–</td>
<td>–</td>
<td>0.93</td>
<td>0</td>
</tr>
</tbody>
</table>

† indicates miRBase entries that have been removed in the most release(s) because their source has been convincingly identified as another class of ncRNAs. § reported sno-miRs from (Brameier et al., 2011) in human. ¶ also discussed in (Scott et al., 2009). * overlap of microRNA and snoRNA annotation in multiple species. ¹ the mature hsa-miR-1246 maps both to the U2 snRNA and a degraded hairpin-like structure deriving from a MLT1M ERVL-MaLR repetitive element.
MicroRNA or not microRNA?

The similarity between H/ACA snoRNAs and microRNAs has been noticed in several computational studies. For example, (Scott et al., 2009) reports twenty miRNA precursors that show significant similarity to H/ACA snoRNAs; five of these (miR-151, miR-605, mir-664, miR-215 and miR-140) even bind to dyskerin, a component of the H/ACA snoRNP. Some microRNAs, furthermore, are known to be predominantly localized in the nucleolus (Politz et al., 2009) emphasizing their snoRNA-like features. This may suggest that a subset of microRNA precursors may have evolved from snoRNAs (Scott et al., 2009). The production of small RNAs from snoRNAs, on the other hand, is independent of Drosha (Ender et al., 2008; Taft et al., 2009b; Brameier et al., 2011), and in some cases Drosha even inhibits sdRNA formation (Taft et al., 2009b), suggesting that the snoRNAs and (canonical) microRNAs are in general clearly distinguished entities.

Here we investigate systematically the conflicts in annotation between microRNAs in miRBase (Griffiths-Jones, 2004) and other classes of ncRNAs as defined by a variety of other databases. Since most of the conflicts, not surprisingly, concern overlaps of microRNA and snoRNA assignments, we focus in particular on these cases.

5.2 Materials and methods

5.2.1 Conflicting microRNA annotation

In order to determine to what extent the microRNA annotation of miRBase conflicts with non-coding RNA annotation stored in other databases, we retrieved the mature miR sequences from miRBase (v. 16) and compared them against Rfam (Griffiths-Jones et al., 2003) (v. 10.0) using the mapping tool segemehl (Hoffmann et al., 2009). We found 38 mature miRNAs mapping perfectly to other annotated ncRNAs. Stringently requiring exact hits of sequences from the same species and collapsing overlaps observed in more than one species left 26 examples. In addition, a few previously known cases from the literature have been included in Table 5.1. Most of the overlaps concern snoRNAs. In some cases, these “mature microRNAs” have length of 24 or larger, i.e., outside the range observed for canonical microRNAs.

It is important in this context to recall the common practice of annotating microRNAs. Experimental evidence is almost always only available for the mature microRNA. After mapping the mature sequence to the genome, putative precursor hairpins are then assigned based solely on computational secondary structure predictions of the surrounding genomic DNA sequence. In many of the cases listed in Tab. 5.1 we observe that the annotated precursor hairpins only partially overlap alternative annotations, while the short RNA may arise from either of the conflicting putative precursors. Crucially, annotations as snoRNAs or other RNA classes are often supported by direct evidence, such as cloning and sequencing or Northern blots, which are lacking for the putative pre-microRNA.

One possibility to distinguish evolutionarily conserved microRNAs from evolutionarily con-
Figure 5.1: Examples of putative microRNAs that are most likely mis-annotated. L.h.s.: SNORD100 is a well-conserved box C/D snoRNA, while “mir-1454” would be specific to chicken. R.h.s.: The annotated platypus mir-1348 precursor sequence is located in a putative precursor hairpin whose 5’ side is not conserved at all. The, alternative explanation, the 5’ side of the 2nd hairpin of the box H/ACA snoRNA SCARNA15, on the other hand, is highly conserved.

In many cases, however, the putative microRNA is rather poorly conserved and there is little or no conservation for the precursor hairpin, while at the same time the alternative annotation as a snoRNA or other ncRNA features a deep phylogenetic conservation. UCSC Genome Browser representations of two examples are shown in Fig. 5.1. Although there is clear block of short RNAs for chicken mir-1454, the sequence conservation is extremely poor and there is no signal for a matching miR*. Thus, if mir-1454 is indeed a microRNA, it is almost certainly specific to chicken. On the other hand, SNORD100 is conserved at least across vertebrates. Since there is no paralog of the snoRNA in the chicken genome, it is parsimonious to assume the short reads interpreted as mir-1454 constitute an sdRNA deriving from a box C/D snoRNA precursor. Another example of this kind is SCARNA15, a box H/ACA snoRNA, for which a platypus mir-1348 was annotated, Fig. 5.1. The 3’ arm of the microRNA containing the annotated mature sequence overlaps the 5’ arm of the second hairpin of the H/ACA snoRNA. The stem loop structure of the putative pre-microRNA untypically shows two larger interior loops, while the putative snoRNA shows a perfect double stem
MicroRNA or not microRNA?

loop pattern with perfect conservation of both the H and ACA boxes. Again, the detailed inspection of the locus suggests that it should be considered as a conserved snoRNA rather than a microRNA.

In addition to manual inspection, we applied the class-specific gene finders RNAmicro (Hertel and Stadler, 2006) and snoReport (Hertel and Stadler, 2006) to assess the overlaps of miRNA and snoRNA annotations of Table 5.1. The possible classifications are (1) microRNA but not snoRNA, (2) vice versa, (3) both classes predicted with high probability and (4) no classification as microRNA or snoRNA at all. As expected, the majority falls into the classes (1) or (2). There are only three candidates for case (3). Neither class is assigned in cases where the putative microRNA precursor hairpin is not conserved in related species so that RNAmicro cannot be used, and snoReport fails to recognize a box H/ACA or box C/D snoRNA structure.

The main advantage for classifying microRNAs with RNAmicro is the use of comparative information. Thus, stem loop structures of annotated microRNAs that look characteristically at a first glance are nevertheless not classified as microRNA if the conservation pattern is not as expected for typical microRNAs. Applying snoReport to those sequences (extended if necessary) almost always yields good snoRNA classification. A nice example is the overlapping annotation of mir-1940 and SNORA26 in mouse. While the secondary structure of the annotated miRNA in mouse and related species is a nicely conserved stem loop, the underlying conservation pattern is not miRNA-like (constantly high at the mature and mature-star part and low in the hairpin loop region). This is the reason for the low prediction probability ($p = 0.000017$) of the RNAmicro SVM. The clear occurrence and conservation of the H and ACA box, their distances to each other and the hairpin-hinge-hairpin-tail secondary structure prediction pattern, however, yields a high classification probability ($p = 0.97$ for box H/ACA snoRNA) of the snoReport SVM.

5.2.2 Comparative analysis of H/ACA hairpins

Most box H/ACA snoRNAs consist of two hairpins. We ask here whether hairpins that give rise to large amounts of small reads are more “microRNA-like” than hairpins of H/ACA snoRNAs that are no prolific sources of short RNA products. Hence, we employ an SVM classifier that is trained from two disjoint sets of hairpins: (1) The bona fide evolutionarily conserved microRNA precursor compiled in (Hertel et al., 2006), which contains neither repeat-derived microRNAs nor lineage-specific ones. (2) A H/ACA hairpin set consisting of those hairpins of H/ACA snoRNAs that show very low levels of short RNA production. To determine small RNA molecules originating from these loci we used mapped reads from different developmental stages of the human brain (GSE18012). Using principal component analysis, we selected the following features for the final SVM classifier: the mean pairing probability of all nucleotides, the number of bound bases, the GC content, the longest paired region, the energy $z$-score of
Figure 5.2: Histogram of the SVM decision values for all 1,048 miRNAs annotated in miRBase v16. The positions of the 12 putatively mis-annotated miRNAs are indicated. Only four of these (mir-605, mir140, mir-1291 and mir-151) are unambiguously classified as miRNAs, of these, only mir-1291 overlaps a known snoRNA. The remaining ones show conservation patterns and structural features similar to snoRNA hairpins.

5.3 Discussion

In addition to an increasing number of transcripts with multiple processing products and multiple functions, an increasingly diverse universe of small RNAs has been described. Small RNAs are produced by a wide variety of mechanisms, they originate from a broad array of source transcripts, and they exert a broad range of biological functions. This begs the question what exactly should be considered as a microRNA as opposed to the many other types of small RNAs. The most inclusive definition, favored in at least part of the literature, encompasses any short RNA that is incorporated in an Argonaute complex. This point of
view has lead to the inclusion in mirBase of significant number of small RNAs that are from snoRNAs, snRNAs, tRNAs, and other structured RNAs. We systematically search for such cases and investigated to what extent the ambiguities in the annotation can be decided. We found that short RNAs can often be recognized as products of well-know structured ncRNAs other than microRNAs, leaving also the annotated putative pre-microRNA hairpin doubtful at best.

Although the definition of “microRNA” at first glance may seem to be a purely semantic issue, it has important consequences in practice, since it determines what is included in databases such as mirBase. This in turn determines, e.g., what is used in practice as training sets for machine learning approaches. In the case of microRNAs, for which typically the precursor hairpins are utilized, one unknowingly works with contaminated datasets when “microRNAs” are included that are not produced in the canonical way or not all from the annotated data set. The inclusion of mitrons and other non-canonical precursors, for instance, precludes the identification of features associated with Drosha processing. From this point of view, a more stringent curation of microRNAs as well as an explicit annotation of the source of the short RNAs would be highly desirable.
CHAPTER 6

DeepBlockAlign: Aligning short RNA-seq block patterns

Contents

6.1 Introduction .................................................. 72
6.2 Material and methods ........................................ 74
   6.2.1 Data and their preprocessing ............................. 74
   6.2.2 Read pattern within a block group ....................... 75
   6.2.3 Alignment strategy ........................................ 76
   6.2.4 Alignment of read blocks ................................. 77
   6.2.5 Alignment of block groups ............................... 79
   6.2.6 Clustering .................................................. 81
6.3 Results ......................................................... 81
   6.3.1 Conservation of processing patterns ..................... 81
   6.3.2 Clustering of aligned block groups ...................... 81
   6.3.3 Novel ncRNA candidates clustering together with known classes ..................... 85
6.4 Discussion ..................................................... 85

Since we have shown, that the read patterns are highly similar within one ncRNA class, we were curious about the possibility of developing a fast alignment method for these patterns. After using blockbuster, we ended up with quite simplistic block structures, that can easily be aligned by comparing and maximizing the scores of two sources of information,
DeepBlockAlign: Aligning short RNA-seq block patterns

the density of reads within the blocks and the distances between the blocks. The upcoming chapter describes a modified Sankoff algorithm and shows, that a downstream clustering of the aligned block patterns validates the functionality of the algorithm and that it can give a lot information about unknown expressed loci. The developed algorithm deepBlockAlign was published in Bioinformatics (Langenberger et al., 2012a) and this chapter is based on this publication.

deepBlockAlign introduces a two-step approach to align RNAseq 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 the modified 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, but 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.

The program deepBlockAlign is available at http://rth.dk/resources/dba/.

6.1 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 (Langmead et al., 2009; Hoffmann et al., 2009; Trapnell et al., 2009). Here, we focus in particular on small RNAseq data. As shown before, 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., 2009b). These small RNAs are often, but not always, produced by Dicer (Lee et al., 2009; Haussecker et al., 2010; Cole et al., 2009; Burroughs et al., 2011; Brameier et al., 2011). Several alternative, Dicer-independent pathways that lead to similar small RNAs with microRNA-like functions have been characterized, see (Miyoshi et al., 2010) for review.
The apparent diversity of processing pathways bears the question to what extent the read patterns in RNAseq 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 (see chapter 3). As a consequence, these patterns convey information about the parent RNAs. In chapter 3, a machine learning algorithms has 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 et al. (2010); Gorodkin and Hofacker (2011)).

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 towards 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.
6.2 Material 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 (described in chapter 2). 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.

6.2.1 Data and their preprocessing

In order to construct a set of benchmark data for deepBlockAlign, we downloaded previously published Illumina sequencing data sets shown in Table 6.1. The human (hg18, Mar. 2006) and rhesus macaque (rheMac2, Jan. 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 less than 30nt. 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). 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 data set 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_crb dataset with more than one block, at least 50 reads and the size range between 50 nt and 200 nt. This
6.2 Material and methods

Table 6.1: The HTS dataset used in this study along with possible ID from GEO, the number of reads and number of block groups

<table>
<thead>
<tr>
<th>Dataset (species)</th>
<th>Tissue</th>
<th>GEO ID</th>
<th>#reads</th>
<th>#block groups^a</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human_eeb</td>
<td>embryoid cells^b</td>
<td>-</td>
<td>7,351,304</td>
<td>1,136</td>
</tr>
<tr>
<td>Human_hesc</td>
<td>embryonic stem cells^b</td>
<td>-</td>
<td>7,836,912</td>
<td>1,386</td>
</tr>
<tr>
<td>Human_34 brain</td>
<td>(34 days)^c</td>
<td>GSM450598</td>
<td>7,299,034</td>
<td>1,103</td>
</tr>
<tr>
<td>Human_98 brain</td>
<td>(98 years)^c</td>
<td>GSM450608</td>
<td>8,371,772</td>
<td>1,109</td>
</tr>
<tr>
<td>Human_14 brain</td>
<td>(14 years)^c</td>
<td>GSM450605</td>
<td>8,538,940</td>
<td>1,614</td>
</tr>
<tr>
<td>Monkey_9 brain</td>
<td>(9 years)^c</td>
<td>GSM450615</td>
<td>10,698,419</td>
<td>1,738</td>
</tr>
</tbody>
</table>

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. ^aBlock groups with >1 blocks, >50 nt and <200 nt in length. ^bMorin et al. (2008) ^cSomel et al. (2010)

data set 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 the results from chapter 3 we observe that different ncRNAs give 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.

6.2.2 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

\[
I = - \sum_i q_i \log_2 q_i .
\]

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 6.1. 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 the
different ncRNA classes.

![Frequency distribution of Entropy in block groups from human](image)

Figure 6.1: 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.

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.

### 6.2.3 Alignment strategy

The purpose of **deepBlockAlign** is the comparison of the read mapping patterns of two block groups obtained from short RNAseq 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.
6.2 Material and methods

6.2.4 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 \( \alpha \) and \( \beta \) 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 practice.

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

\[
x_i = \frac{x_{1i} - x_{2i}}{N_X}
\]

(6.2)

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

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

(6.3)

where \( \delta \) 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

\[
\epsilon(i, j) = |x_i - y_j|/\max\{x_i, y_j\}
\]

(6.4)

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
\[
\eta^+(i, j) = \zeta \cdot \left\| \frac{|x_i - y_j| - |x_{i-1} - y_{j-1}|}{\max\{|x_i - y_j|, |x_{i-1} - y_{j-1}|\}} \right\|,
\]
otherwise we use \(\eta^-(i, j) = 0\). The functions \(\epsilon\) and \(\eta\) tune the match and mismatch scores according to the difference in expression and shape of the two read blocks, respectively. \(\zeta\) is a parameter tuning the relative importance of \(\eta\), and hence of the variation between adjacent positions.

Let \(D_{i,j}\) and \(E_{i,j}\) denote the optimal score of a sub-alignment ending in a deletion \((x_i, -)\) and an insertion \((-, y_j)\), respectively, and \(M_{i,j}\) denote the optimal score of a sub-alignment ending in a substitution \((x_i, y_j)\), i.e., a match or mismatch. We furthermore define
\[
S_{i,j} = \max\{M_{i,j}, D_{i,j}, E_{i,j}\}.
\]

These scores satisfy the recursions
\[
M_{i,j} = \max \left\{ \begin{array}{l} M_{i-1,j-1} + \Psi^+(i, j) \\ D_{i-1,j-1} + \Psi^-(i, j) \\ E_{i-1,j-1} + \Psi^-(i, j) \end{array} \right.,
\]
\[
D_{i,j} = \max \left\{ \begin{array}{l} S_{i,j-1} + C_i \\ D_{i,j-1} + C_e \end{array} \right.,
\]
\[
E_{i,j} = \max \left\{ \begin{array}{l} S_{i-1,j} + C_i \\ E_{i-1,j} + C_e \end{array} \right.,
\]

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 (6.6), 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 (Crooks et al., 2005; Akbasli, 2007). 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 Eq. (6.3) redundant. Other parameter values (with smaller \(\delta\)) give comparable results. We tested a range of values for \(\delta\) 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 6.2a.
6.2 Material and methods

6.2.5 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 $C = C_1 \cdots C_n$ and $K = K_1 \cdots K_m$, 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 Eq.(6.6). We furthermore need the differences

$$\Delta_{i,j:k,l} = |\tau(C_j) - \tau(C_i)| - |\tau(K_l) - \tau(K_k)|$$

(6.7)

of the distances between the pairs of blocks $C_i, C_j \in C$ and $K_i, K_j \in K$, respectively. Here $\tau(B)$ 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 $\{C_i, C_{i+1}, \ldots, C_{j-1}, C_j\} \subseteq C$ with the subsequence $\{K_k, K_{k+1}, \ldots, K_{l-1}, K_l\} \subseteq K$. 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

$$S_{i,j:k,l} = \max \begin{cases} S_{i+1,j:k,l} + \gamma \\ S_{i,j:k+1,l} + \gamma \\ S_{i+1,j:k+1,l} + S_{i,k} \\ \max_{h \leq j, q \leq l} (S^M_{i:h:k,q} + S_{h+1,j:q+1,l}) \end{cases}$$

and

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

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:

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

where $\nu_{\text{dist}}$ and $\nu_{\text{block}}$ are weight factors.
DeepBlockAlign: Aligning short RNA-seq block patterns

Figure 6.2: 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 6.4c 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 $v_{\text{dist}}$ and $v_{\text{block}}$ are parameters to weight the influence of the distance between the blocks and the block scores, respectively. Their default values of the parameters for block group alignment are $\gamma = -1$, $v_{\text{dist}} = 6$ and $v_{\text{block}} = 1$.

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 6.2b.
6.3 Results

6.2.6 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. 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 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-value of < 0.1.

6.3 Results

6.3.1 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 to 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 data sets in Table 6.1 all those loci that are expressed in multiple experiments. We then aligned each block group with all block groups from another data set and ranked the block groups by their deepBlockAlign scores. Figure 6.3 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.

6.3.2 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
DeepBlockAlign: Aligning short RNA-seq block patterns

Figure 6.3: 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_48 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.

benchmark dataset. Using average linkage hierarchical clustering, we obtained the tree of significant clusters as shown in Figure 6.4. Two well separated clusters were observed, one containing mainly microRNAs (red) and the other comprised of tRNAs (blue). Within these two large clusters, 33 distinct sub-clusters were identified (p-value<0.1), the largest one containing 90 and the smallest with only 2 block groups.

Within the miRNA cluster two significant (p-value<0.1) sub-clusters (see Figure 6.4a III and IV) contain most of the microRNAs. Sub-cluster IV represents miRNAs with an additional block directly upstream or downstream of the mature microRNA. As shown in chapter 2, these moRNAs are a distinct class of small RNAs that arise from pre-miRNA proximal regions in chordates as well as in humans (Shi et al., 2009; Langenberger 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 to miRNAs or tRNAs in our dataset. (ii) No precise
Figure 6.4: 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 sub-cluster (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.
demarcation of entropy for snoRNAs (Figure 6.1). 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 (cf. Supplementary Table 1 in Langenberger et al. (2012a)). 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 (a) precise start position of the reads, (b) 1-3 read blocks, and (c) 10-20 nt distance between the blocks (miR and miR*), see Figure 6.4b. 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 (Figure 6.4a II).

The tRNA cluster is more variable compared to 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 (see Figure 6.1), we assumed that tRNAs sharing the same anticodon would have similar processing patterns. Unfortunately we were not able to find sub-clusters 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 (Figure 6.4a I).

Interestingly, Lee et al reported a set of individual and characteristic tRNA-derived fragments. They claimed that these fragments are not just random degradation products but are actively derived from mature tRNAs by specific endonucleotic cleavage or exonuclease digestion by a number of enzymes (Lee et al., 2009). Also a Dicer-dependent processing was suggested for a few tRNAs (Cole et al., 2009; Babiarz et al., 2008). 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 (Haussecker et al., 2010; Burroughs et al., 2011). Therefore, we examined the 13 tRNAs clustered significantly (p-value<0.1) within the microRNA cluster (Figure 6.4a V and VI). 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).

### 6.3.3 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 6.4c. 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 RNAseq data. The antisense reads, however, do not necessarily imply the actual transcription of such an RNA, since the complementary stem regions in some cases can not be distinguished. Upon a detailed inspection, we observed some strand specific tags for both hsa-mir-486 and hsa-mir-625 (cf. Supplementary Figure 6 and 7 in Langenberger et al. (2012a)). However, considering the high rate of sequencing error in some tags and perfect complementarity of hairpins in the two miRNAs, 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 can not be expected that all the results will overlap with ncRNA prediction programs. These results thus require further validation. Two candidates clustered together with a 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.

### 6.4 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 microR-
NAs as well as the majority of the tRNAs fall into well separated clusters (see Figure 6.4). Within the microRNA cluster, a sub-cluster 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. Nevertheless, we indicated that our approach is fairly robust for parameter variation.

Qualitatively, the read-based clusters closely resemble the results of clustering known and predicted ncRNAs based on their secondary structure (Will et al., 2007; Kaczkowski et al., 2009). We suspect that this is not a coincidence, since small RNAs are preferentially produced from base-paired regions (see Figure 3.3 in chapter 3). This suggests that read mapping patterns are likely to be influenced, or even determined, by the secondary structure of the parental RNA.

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 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.
As shown before, canonical microRNAs are excised from their hairpin-shaped pre-
cursors by Dicer. In order to find possible exceptions to this rule and to identify
additional substrates for Dicer processing we re-evaluate the small RNA sequencing data of
the Dicer knockdown experiment in MCF-7 cells originally published by Friedländer et al. (2012). While the well-known non-Dicer mir-451 is not sufficiently expressed in these data, there are several additional Dicer-independent microRNAs, among them the important tumor suppressor mir-663a. Among the non-miRNA Dicer substrates we recover previously described examples such as tRNA-Gln and some snoRNAs. Interestingly, sdRNAs derived from box C/D snoRNAs are Dicer-independent, while those derived from box H/ACA snoRNAs are often Dicer dependent. Several pol-III transcripts, in particular the vault RNAs and the great ape specific snaRs are processed by Dicer, while the small RNAs originating from Y RNAs follow a different pathway. This chapter follows the manuscript published in the Journal of Experimental Zoology (Langenberger et al., 2012b).

### 7.1 Introduction

For microRNAs, several alternative processing pathways that bypass Drosha have been reported. The most prominent example are mirtrons (Okamura et al., 2007; Ruby et al., 2007), whose precursor hairpins are produced by splicing. A related, mirtron-like source of small RNAs requires both splicing and exosome-mediated trimming to extract the pre-microRNA hairpin (Flynt et al., 2010; Chong et al., 2010). More recently, it was shown that a few microRNAs, in particular mir-451, are matured without the help of Dicer (Cheloufi et al., 2010; Cifuentes et al., 2010). For a recent review of the many alternative pathways for the biogenesis of microRNAs and other, microRNA-like small RNA species see e.g. (Yang and Lai, 2011).

Dicer is not only involved in microRNA biogenesis, however. It appears to be involved also in the processing of small RNA species beyond canonical microRNAs. Short, microRNA-like RNAs are processed from a diverse set of usually well-structured non-coding RNAs that includes tRNAs (Lee et al., 2009; Cole et al., 2009; Haussecker et al., 2010; Findeiß et al., 2011; Sobala and Hutvagner, 2011), snoRNAs (Kawaji et al., 2008; Taft et al., 2009b; Langenberger et al., 2010; Brameier et al., 2011), vault RNAs (Stadler et al., 2009; Persson et al., 2009), Y RNAs (Langenberger et al., 2010; Meiri et al., 2010; Verhagen and Pruijn, 2011), and snRNAs (Langenberger et al., 2010). Not much is known about the processing of most these small RNAs. The importance of Dicer has been demonstrated in only a few cases: the small RNAs derived from human tRNA(Gln) are dependent on Dicer both in vivo and in vitro (Cole et al., 2009), see also (Babiarz et al., 2008). A few snoRNA derived sdRNAs show altered expression in mouse Dicer1 and Dgcr8 mutants (Taft et al., 2009b), and processing of ACA45 derived sdRNAs requires Dicer activity but not Drosha/DGCR8 (Ender et al., 2008). Endogenous siRNAs resulting from Dicer cleavage of long hairpins, typically deriving from SINEs with tandem inverted repeat structure have been reported in (Babiarz et al., 2008).

Here we reevaluate a previously published set of RNA sequencing data (GSE31069) that
compare the expression of small, microRNA-sized RNAs before and after \textit{Dicer} knock-down in a MCF-7 cell line (Friedländer \textit{et al.}, 2012). Our analysis focuses on the identification in particular of microRNAs that fail to respond to the depletion of \textit{Dicer}, and conversely on those loci that are strongly \textit{Dicer}-dependent but are not classified as microRNAs.

7.2 Materials and methods

7.2.1 Data and mapping

We downloaded a previously published sequencing data set series (GSE31069, (Friedländer \textit{et al.}, 2012)) from the Gene Expression Omnibus (GEO) database (Edgar \textit{et al.}, 2002). The data consists of four different samples, two containing short reads from the total cell content and two containing reads from the cytoplasmic fraction only. Both pairs contrast small RNA expression before and after \textit{Dicer} knock-down in a MCF-7 cell line. The analysis reported here uses only the cytoplasmic sample pair (GSM769509 and GSM769511). Since short RNA processing takes place in this compartment we expect to reduce the noise from the nucleus.

All the adapter-free reads were mapped against the human genome (NCBI36.50 Release of July 2008) using \textit{segemehl} (Hoffmann \textit{et al.}, 2009): we activated the poly-A clipping, required small RNAs to map with an accuracy of at least 90\% and selected the “best scoring hit strategy”. With these settings we mapped 8,743,377 of 15,493,265 reads (56\%) of the control sample and 5,471,242 of 9,237,490 reads (59\%) of the \textit{Dicer} knock-down sample. The resulting \texttt{sam} files were converted to \texttt{bam} format, using samtools (Li \textit{et al.}, 2009a) and subsequently translated to \texttt{bigWig} files using a custom perl script. The read density at each position in the \texttt{bigWig} files was normalized by the number of multiple hits of each read and the absolute number of mapped reads of each experiment (RPM) in order to make the two experiments comparable. The \texttt{bigWig} files were uploaded to the UCSC Genome Browser (Kent \textit{et al.}, 2002) to make them publicly available.

7.2.2 Expressed sites and annotation

In order to identify previously un-annotated loci with small RNA expression we created sorted bed files and then used \textit{blockbuster} (Langenberger \textit{et al.}, 2009) with default parameters to identify regions showing accumulations of at least 50 reads in at least one of control or \textit{Dicer} knock-down data. We used \texttt{mergeBed} from BEDtools (Quinlan and Hall, 2010) to obtain the final list of expressed regions of interest (1,946 for control and 1,798 for the knock-down set), which we call “sites” from now on.

We downloaded the latest annotations from different sources (1523 microRNA loci from miRBase v18 (Griffiths-Jones, 2004); 631 tRNA loci from gtRNAdb (Chan and Lowe, 2009); 402 snoRNA loci as well as 4528 other RNAs from UCSC annotation (Karolchik \textit{et al.}, 2004)).
This combined annotation track comprising 7,084 annotated ncRNA loci was compared with our list of sites using `intersectBed` (Quinlan and Hall, 2010). Furthermore, all reads were overlapped with the UCSC repeat masker track (Jurka et al., 2000) and as soon as one read was mapped to a repeat associated region, all multiple hits of it were flagged with the type of repeat. If more than 50% of the expression of one site is caused by reads which are flagged as repeat associated, the whole site was flagged accordingly. In order to remove low-complexity sequences, which have a high probability of being random matches in short read data, we discarded all sites with a Shannon entropy of less than 1.6 bit.

### 7.2.3 Expression levels

The expression level of each site, expressed in reads per kilobase of locus per million mapped reads (RPKM) was computed using the UCSC tool `bigWigAverageOverBed` (Kent et al., 2002). From these values we derived, for each site, the log$_2$-fold change $\lambda$ between the Dicer knock-down sample and the RPKM of the control sample. All sites with $\lambda < 0$ are interpreted as Dicer processed.

### 7.2.4 Processing pattern

Cleavage of a nearly double-stranded RNA by Dicer leads to a characteristic 2 nt overhang at the 3’ end, see e.g. (Ji, 2008). In order to assess how important the thermodynamic stability of the precursor structure is for processing, we computed for a pair of putative single-stranded cleavage products, the following stability measure: `RNAcofold` (Bernhart et al., 2006b) is used to compute the energy of the duplex with the constraint that the joint structure exhibits the 2nt overhang at the 3’ ends. Then the inner part of both sequences is shuffled 100 times so that the dinucleotide composition is preserved, while the terminal base pairs and overhanging nucleotides were left untouched. The resulting z-score of the co-folding energies is recorded. For each site we considered the two consecutive tags with the largest expression as candidates for Dicer processing.

In order to assess the overall similarity of a site with canonical microRNAs we use `RNAmicro` (Hertel and Stadler, 2006). This tool evaluates structural features as well as the pattern sequence conservation. We retrieved alignments of all sites with 20nt flanking sequence on both sides from the 8way-multiZ alignment (human, chimp, orangutan, rhesus macaque, marmoset, mouse, opossum, platypus) (Blankenberg et al., 2011). We extracted sequences from 8way-multiZ file, re-alignes them using clustalw (Larkin et al., 2007) and used it to run RNAmicro. Then, the RNAmicro decision value (decV) was used to rate the sites, if they microRNA-like structures and conservations.

Dicer is well known to generate products in the narrow length range 21–28 nt, see e.g. (Starega-Roslan et al., 2011). We therefore recorded the distribution of read lengths for each locus. In addition, we determined the lengths of blocks of reads `blockbuster` (described in...
Figure 7.1: Summary of expression changes of small, microRNA-sized RNAs in response to a knock-down of Dicer. The entire dataset is shown in grey, specific groups are highlighted as black dots. (a) Almost all annotated microRNAs are down-regulated, i.e., exhibit log₂-fold changes λ < 0. (b) Only a few tRNAs are downregulated. (c) None of the sdRNAs derived from box C/D snoRNAs is depleted in response to the Dicer knock-down, while (d) the majority of the (small number of) sdRNAs derived from box H/ACA snoRNAs is Dicer dependent. (e) The small RNAs originating from Y RNAs and almost all Y RNA derived loci are not downregulated in response to Dicer knock down. (f) Mitochondrial transcripts and/or NUMTs are also a prolific source of small RNAs. These are independent of Dicer processing. Among repetitive elements, a substantial fraction of (g) expressed SINEs and (h) expressed LINEs shows Dicer dependent processing.
chapter 2) with default parameters. Read blocks summarize groups of reads that overlap nearly perfectly, hence its lengths is typically larger than that of individual reads.

7.3 Results

7.3.1 Identification of Dicer-dependent small RNAs

The Dicer knock-down (GSM769509) and control (GSM769511) datasets (Friedländer et al., 2012) together identify 2,115 expressed sites. Of these, 1,048 overlap with the 7,084 annotated ncRNAs and 1,067 remain unannotated. After filtering out the low-complexity sites, we retain 1,002 annotated and 539 unknown sites for further analysis.

Fig. 7.1 summarizes the response of the small RNA sites to Dicer knockdown. The log₂-fold change λ exhibits the expected bi-modal distribution separating in particular microRNAs from other small RNA products. Consistent with the original analysis of these datasets (Friedländer et al., 2012), microRNAs are strongly reduced upon reduction of Dicer activity. A closer inspection, however, shows a more differentiated pictures.

On the one hand, a small subgroup of microRNAs does not respond to the knockdown of Dicer. On the other hand, a sizable number of unannotated sites (some of which might constitute previously undescribed microRNAs) are associated with well-known structured RNAs exhibiting large negative values of λ, see Table 7.1.

A substantial fraction of sites expressing small RNAs are annotated repetitive elements, Table 7.2. Disregarding a moderate number of simple repeats and low complexity regions, which cannot be unambiguously distinguished from artefacts without further experimental evidence, we observe that about one fifth of repeat-associated small RNAs react to Dicer. This is not unexpected, as repetitive elements are a prolific source of novel microRNAs. Smalheiser and Torvik (2005), for instance showed that a subset of conventional mammalian microRNAs is derived from LINE-2 transposable elements. A family of miRNAs deriving from miniature inverted-repeat transposable elements (MITES) has been characterized by

<table>
<thead>
<tr>
<th>type</th>
<th>Dicer processed</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>yes</td>
<td>no</td>
<td>all</td>
<td>processed</td>
</tr>
<tr>
<td>miRNA</td>
<td>255</td>
<td>10</td>
<td>265</td>
<td>96.2%</td>
</tr>
<tr>
<td>tRNA</td>
<td>32</td>
<td>376</td>
<td>408</td>
<td>7.9%</td>
</tr>
<tr>
<td>H/ACA snoRNA</td>
<td>8</td>
<td>4</td>
<td>12</td>
<td>66.0%</td>
</tr>
<tr>
<td>C/D snoRNA</td>
<td>0</td>
<td>53</td>
<td>53</td>
<td>0.0%</td>
</tr>
<tr>
<td>misc RNA</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>60.0%</td>
</tr>
<tr>
<td>snRNA</td>
<td>2</td>
<td>92</td>
<td>94</td>
<td>2.1%</td>
</tr>
<tr>
<td>scRNA</td>
<td>10</td>
<td>90</td>
<td>100</td>
<td>10.0%</td>
</tr>
<tr>
<td>rRNA</td>
<td>28</td>
<td>254</td>
<td>282</td>
<td>9.9%</td>
</tr>
</tbody>
</table>

Table 7.1: Fraction of Dicer processed sites among the annotated ncRNAs.
(Piriyapongsa and Jordan, 2007). A recent comprehensive analysis of microRNAs originating from transposable elements can be found in (Borchert et al., 2011).

7.3.2 Characterization of Dicer-processed sites

*Dicer*-processed small RNAs come from small RNAs having a duplex structure that is significantly more stable than those from *Dicer*-independent ones. Fig. 7.2a shows that in particular putative precursor structures that give rise to the typical processing patterns with 2nt overhangs are substantially stabilized *Dicer*-responsive small RNAs.

Canonical microRNAs also exhibit a characteristic pattern of sequence conservation that can help to distinguish them from other, similar, sources of small RNAs and from hairpin-like structures that are not processed into small RNAs, see e.g. (Lai et al., 2003). *RNAmicro* (Hertel and Stadler, 2006) implements such a classifier based on a Support Vector Machine taking only a small number of structural and conservation based descriptors as input. Only sites that form hairpin structures can be scored by *RNAmicro*’s SVM. In chapter 5, we have used *RNAmicro* to distinguish microRNA-like from snoRNA-like small RNA sites (Langenberger et al., 2011). Fig. 7.2b shows that the *RNAmicro* decision value is also correlated with $\lambda$. With few exceptions, large decision values are limited to *Dicer* responsive sites.

Fig 7.2c summarizes the distribution of read lengths. As expected, nearly all reads arising from sites with $\lambda < -0.2$ have lengths between 20 and 25nt, consistent with *Dicer* processing (Starega-Roslan et al., 2011). In contrast, short reads from sites with $\lambda > 0.2$, i.e., those that are clearly not resulting from *Dicer* cleavage, are typically longer and show a flat distribution. We also observe a difference in the length of read blocks as determined by *blockbuster* (Langenberger et al., 2009). Sites with $\lambda < 0$ have on average much shorter block sizes, often consisting only of a single block of microRNAs, Fig. 7.2d. Since the start and end position of mature microRNAs can vary by a couple of nucleotides (Ebhardt et al., 2010) such that

<table>
<thead>
<tr>
<th>type</th>
<th>Dicer processed yes</th>
<th>Dicer processed no</th>
<th>all processed</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUMT / chrM</td>
<td>1</td>
<td>66</td>
<td>67</td>
</tr>
<tr>
<td>SINE</td>
<td>126</td>
<td>427</td>
<td>553</td>
</tr>
<tr>
<td>LINE</td>
<td>141</td>
<td>368</td>
<td>509</td>
</tr>
<tr>
<td>LTR</td>
<td>81</td>
<td>306</td>
<td>387</td>
</tr>
<tr>
<td>DNA</td>
<td>27</td>
<td>106</td>
<td>133</td>
</tr>
<tr>
<td>Simple repeat</td>
<td>18</td>
<td>66</td>
<td>84</td>
</tr>
<tr>
<td>Low complexity</td>
<td>15</td>
<td>15</td>
<td>30</td>
</tr>
<tr>
<td>Other</td>
<td>1</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>Satellite</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>RNA</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>tRNA</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
overlapping microRNAs read blocks have a length of around 30 nt.

Figure 7.2: Correlation of Dicer regulation with sequence-derived descriptors: (a) Free energy $z$-scores of structures with 2-nt overhangs that conform to the canonical Dicer processing pattern. (b) Decision value of RNAmicro, a SVM-based machine learning tool trained to recognized canonical microRNAs. Its decision value combines the stability of the hairpin structure with patterns of sequence conservation but is agnostic about the location of the small RNA products. Only sites that form a hairpin structure can be scored by this method. (c) Reads arising from Dicer processed regions are within the expected size range between 20 and 25nt, while these without evidence for processing are mainly shorter or longer. (d) Dicer cutting results in shorter read blocks, indicating clearly stacked reads and thus specific processing.
7.3 Results

Figure 7.3: Six un-annotated non-repetitive loci that are processed by Dicer: (a) intergenic chr2:81,100,049-81,100,134(+) ; (b) a “semi-mirtron” in an intron of SYT12 chr11:66,569,729-66,569,790(+) ; four mirtrons: (c) SLC4A2 chr7:150,394,782-150,394,835(+) ; (d) FLNA chrX:153,235,873-153,235,943(-) ; (e) MAP3K4 chr1:27,559,917-27,559,998(-) ; (f) TRIM28 chr19:63,753,464-63,753,555(+). The color scale represents the coverage on a logarithmic scale.
7.3.3 Structured regions processed by Dicer indicate potential microRNA candidates

The data set used here has been generated specifically for the purpose of detecting novel microRNAs with mirdeep2 by its original authors (Friedländer et al., 2012). Nevertheless, we found among the un-annotated, non-repetitive sites with $\lambda < 0$ six additional structured regions, Figure 7.3. Four of these loci coincide with short introns. Three of these intronic sequences (Figure 7.3d-f) fold into hairpins and the short reads map exactly to the stem positions expected for mirtrons (Okamura et al., 2007; Ruby et al., 2007). The 3’ end of the candidate located in an intron of SYT12 is determined by the splice acceptor. Together with its stable hairpin structure this suggests that it belongs to the recently described class of “semi-mirtrons” that require both splicing and exosome-mediated trimming to extract the pre-microRNA hairpin (Flynt et al., 2010; Chong et al., 2010). The structure and the positions of mapped reads of candidate Figure 7.3c do not show the typical characteristics of a microRNA. Nevertheless, in the wildtype, more than 70 reads and only five in the Dicer knockdown set map to this locus, indicating a strong processing by Dicer. Since these reads map not only uniquely, but also perfectly to the intronic region, this candidate is of particular interest for further analysis. The final candidate, Figure 7.3a, is located in intergenic region without any annotation in its vicinity.

7.3.4 Dicer-processed non-microRNAs

Surprisingly, there is a large number of well-known structured non-coding RNAs from which Dicer-sensitive small RNAs are produced.

A prominent example are the vault RNAs. The largest response is observed for vtRNA2-1 with $\lambda = -2.12$. This locus was originally classified as hsa-mir-886 but later-on recognized as a polymerase-III transcript (Canella et al., 2010) and vault RNA paralog (Nandy et al., 2009; Stadler et al., 2009). The other three vault RNA loci also give rise to short RNAs (Persson et al., 2009) and respond negatively to the Dicer depletion: $\lambda$(vtRNA1-1) = −0.14, $\lambda$(vtRNA1-2) = −0.76. The vtRNA1-3 locus is not sufficiently expressed.

The snaR ncRNAs (Parrott and Mathews, 2007) are a group pol-III transcripts that

![Figure 7.4: SNARs are processed by Dicer. Highlighted are the tags showing the highest expression.](image-url)
emerged in the ancestor of the African Great Apes from an Alu-derived precursor (Raha et al., 2010; Parrott et al., 2011). Fig. 7.4 shows that microRNA-like small RNAs are processed from the lower end of the stem-loop structure, which resembles a canonical pre-microRNA hairpin except for its length of more than 100nt. The snRN-derived small RNAs show the typical 2 nt 3’ overhangs. Their expression depends very strongly on the Dicer concentration.

The situation is more complex for tRNAs and snoRNAs. While many of them give rise to small RNA products, the majority is not influenced by the Dicer knockdown. A small subset of tRNAs, on the other hand is clearly subject to Dicer processing. These include in particular tRNA-Gln-CTG with $\lambda = -2.05$ as noted already previously by Cole et al. (2009). Other tRNAs with a clear Dicer signature are tRNA-Asn-GTT ($\lambda = -1.47$), tRNA-Asn-ATT ($\lambda = -0.83$), tRNA-Ala-CGC ($\lambda = -1.28$), tRNA-Ile-TAT ($\lambda = -1.19$), tRNA-Glu-TTC ($\lambda = -0.79$). None of the four mirbase “microRNAs” that are derived from tRNAs (mir-1274/tRNA-Lys, mir-1280/tRNA-Leu, mir-720/tRNA-Thr, mir-1308/tRNA-Gly) are expressed at sufficiently high levels to estimate $\lambda$.

Small nucleolar RNAs can share several characteristics with microRNAs, including similar components in their processing, see (Scott and Ono, 2011) for a recent review. The structural

![Figure 7.5: Several tRNAs are processed into small RNAs by Dicer. The processing patterns shown some similarities, in particular a tendency to have large short read coverage on the 3’ side of the tRNA clover leaf. With the exception of tRNA-Ile-TAT the small RNAs are derived from within the mature tRNA.](image-url)
Dicer-Processed small RNAs: Rules and exceptions

Table 7.3: Box H/ACA snoRNAs processed by Dicer. SNORA36B (Ender et al., 2008) (also annotated as mir-664) does not reach a sufficient expression level in MCF-7 cells.

<table>
<thead>
<tr>
<th>snoRNA</th>
<th>λ</th>
<th>snoRNA</th>
<th>λ</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNORA45</td>
<td>-1.55</td>
<td>SNORA36B</td>
<td>*</td>
</tr>
<tr>
<td>SNORA51</td>
<td>-2.42</td>
<td>SNORA46</td>
<td>-1.00</td>
</tr>
<tr>
<td>SNORA36A</td>
<td>-1.33</td>
<td>SNORA56</td>
<td>-0.93</td>
</tr>
<tr>
<td>SNORA17</td>
<td>-1.19</td>
<td>SNORA7B</td>
<td>-0.66</td>
</tr>
<tr>
<td>SCARNA3</td>
<td>-1.13</td>
<td>SNORA7A</td>
<td>-0.28</td>
</tr>
</tbody>
</table>

Similarities between H/ACA snoRNAs and microRNAs are most obvious and have been noticed in several computational studies. Scott et al. (2009), for instance, report twenty miRNA precursors that show significant similarity to H/ACA snoRNAs; of these miR-151, miR-605, mir-664 = SNORA36B, miR-215, and miR-140 even bind to dyskerin, a component of the H/ACA snoRNP. On the other hand, Dicer processing has been demonstrated previously for SNORA45 (Ender et al., 2008). Consistently, we find λ(SNORA45) = -1.55. Of the 12 H/ACA snoRNAs with sufficient expression 8 have λ < 0 (Table 7.3), indicating that short reads from H/ACA snoRNAs are typically a product of Dicer processing. Interestingly, two H/ACA snoRNAs were classified as novel microRNAs by mirdeep2 (Friedländer et al., 2012): SNORA36A (λ = -1.33) and SNORA33 λ = 0.15. We emphasize, however, that only a minority of H/ACA snoRNAs leads to abundant processing products. In addition these small RNAs are independent of Drosha (Ender et al., 2008; Taft et al., 2009b; Brameier et al., 2011), and in some cases Drosha even inhibits sdRNA formation (Taft et al., 2009b), emphasizing that the snoRNAs and (canonical) microRNAs are in general clearly distinguished entities.

A quite different picture emerges for box C/D snoRNAs. Although small RNAs are abundantly produced from box C/D snoRNAs in our data set, Tab. 7.1, there is no indication that any of them is a Dicer substrate. The box C/D snoRNAs that are discussed as possibly microRNA-like in chapter 5 show only marginal expression levels and no indication for Dicer processing. On the other hand, of the five microRNAs that resemble box C/D snoRNAs (having C and D boxes in close proximity in the precursor and binding to fibrillarin) (Ono et al., 2011), four are Dicer substrates (miR-27b λ = -0.90, miR-16-1 λ = -0.40, mir-28 λ = -0.95, and let-7g λ = -1.16) and the fifth (mir-31) is not sufficiently expressed in MCF-7 cells. It appears, thus, that Dicer-processing clearly distinguished between bona fide microRNAs and small RNAs derived from box C/D snoRNAs.

Y RNAs are small pol-III transcripts that originate from RNA component of the Ro RNP particle and have a role in DNA replication (Christov et al., 2006). The four paralogous human Y RNAs form a cluster on Chr.7(148M) (Mosig et al., 2007; Perreault et al., 2007). The canonical loci show no evidence of Dicer processing hY3 λ = 0.12, hY4 λ = 1.25, hY1
7.3 Results

Figure 7.6: Small RNAs derived from Y RNAs. miR-hY3-5p and miR-hY4-5p (Verhagen and Pruijn, 2011) are highlighted.

$\lambda = 1.70$, hY5 $\lambda = 1.74$. We note that fragments from hY5 have also been annotated as mir-1975.

In addition to the canonical Y RNA cluster, however, there are more than a thousand Y RNA pseudogenes scattered across the genome (Perreault et al., 2007). The deep sequencing data shows that several of these loci appear to be expressed and form a source of short reads. A few of the Y4-derived loci sites have negative values of $\lambda$. We note, however, these have relatively low expression levels and might be confounded by mapping artefacts. In total, 11 sites that are derived from Y RNA sequences are classified as microRNAs by RNAmicro, six of which have moderate negative values of $\lambda$.

### 7.3.5 MicroRNA not processed by Dicer

The best-studied microRNA that is not produced by Dicer is mir-415. Unfortunately this site is not significantly expressed in MCF-7 cells, so that we cannot use it as a control. There are ten additional microRNAs with $\lambda > 0$. Six of them (mir-30a, mir-143, mir-374a, mir-379, mir-381, and mir-134) derive from precursor hairpins that are recognized by RNAmicro. Two
Figure 7.7: MicroRNAs that are not processed by Dicer

of these, mir-30a and mir-374a, exhibit exceptionally high levels of expression and feature short RNAs derived from both sides of the precursor stem, Figure 7.7. We suspect that they are exceptionally good substrates for Dicer so that their maturation is least affected by Dicer concentrations. The evolutionarily ancient mir-125b-2 also exhibits both a canonical read pattern and a canonical pattern of sequence conservation. Nevertheless, it shows no reaction to Dicer knockdown, \( \lambda = 0.03 \).

For mir-143, mir-381, mir-134, mir-4417, and mir-4516 no mir\(^*\) reads were detectable, a pattern that is quite atypical for microRNAs. Mir-4417 is present in monkeys only, and no homologs are detectable for hsa-mir-4516, precluding the analysis of patterns of sequence conservation for these two microRNAs.

The entire precursor hairpin of mir-3676 is covered by small RNA sequences. A closer inspection shows, however, that mir-3676 coincides with tRNAThr-AGT and is thus clearly an erroneous annotation. The mis-annotated “mir-3195”, furthermore, corresponds to a GC-rich low-complexity region located with the first exon of the TAF4 gene.

The sequence of mir-663a is very GU-rich and does not meet our exclusion criterion for low-complexity sequences. We retained it in our data set because it is well document as an important tumor suppressor (Pan et al., 2010; Yi et al., 2012). In contrast to canonical microRNAs, its primary sequence is quite poorly conserved although it can be found throughout the major eutherian groups. Its read pattern also strongly deviates from the expectation for microRNAs. In contrast to mir-451 there does not seem to be a single coherent block of reads that defined the mature microRNAs.
7.4 Discussion

A rapidly expanding zoo of diverse small RNA species has emerged following the discovery of RNA interference (Fire et al., 1998) and microRNAs (Lee et al., 1993) almost two decades ago. With the rapid increase of high throughput sequencing data the boundaries between the different subdivisions of small RNAs have become more and more blurry.

Here we have focussed on the generation of small RNAs from their double-stranded precursors. Making use of a publicly available dataset (Friedländer et al., 2012) we find, consistent with the well-established knowledge, that the overwhelming majority of miRBase microRNAs is processed by Dicer. There are, however, several notable exceptions. Cole et al. (2009) argue that Dicer knockdown with siRNAs for a short period of time sometimes does not result in a significant change in the miRNA steady state level due to slow microRNA turnover. At least some of the Dicer-unresponsive miRNAs, however, exhibit unusual structural features and/or read patterns that deviate substantially from canonical microRNAs. While $\lambda > 0$ in itself is course not sufficient proof for Dicer-independence, it is at least a strong indication to identify candidates for further analysis.

Dicer-processing is not limited to microRNAs. Several polymerase-III transcripts are prolific Dicer substrates, including human vault RNAs, the great ape specific snaRs, and a small set of about a dozen tRNAs. While the vault RNAs products function like microRNAs, small RNAs derived from tRNA-Gln-CTG do not function in this way: they do not associate with argonaught presumably due to the fact that these small RNAs are just too small (Cole et al., 2009). Despite their similarity with vault RNAs, including a secondary structure with a long terminal stem, there is no evidence that the abundant small RNAs deriving from Y RNAs are produced by Dicer cleavage. Both main classes of small nucleolar RNAs are sources of abundant small RNAs. While all of the highly expressed box C/D snoRNAs are processed independently of Dicer, the situation is different for H/ACA snoRNAs. Most box H/ACA snoRNAs are a source of small RNAs, but in most cases the expression levels are small, at least in MCF-7 cells. Among the highly expressed ones, however, the majority clearly is a Dicer substrate.

In summary, there does not seem to be a clear separation between processing pathways resulting in small RNAs. Instead, the picture of an intricate network of interleaved alternatives emerges, in which the individual processing steps can be freely combined. As a consequence, it appears that a particular sequence of processing steps is neither a sufficient nor a necessary condition for a particular role.
Several aspects of the RNA maturation may leave traces in RNA sequencing data in the form of deviations from the reference genomic DNA. This includes in particular genomically non-encoded nucleotides and chemical modifications. The latter leave their signatures in forms of mismatches and noticeable patterns of sequencing reads. Modified mapping procedures focusing on particular types of deviations can help to unravel post-transcriptional modification, maturation and degradation processes. Starting from the recovery of many well-known modified sites in tRNAs we provide evidence that modified nucleotides are a pervasive phenomenon in these data sets. Regarding non-encoded nucleotides we concentrate on CCA...
Traces of post-Transcriptional RNA modifications in deep sequencing data

tails. Although small RNA sequencing libraries alone are insufficient to obtain a complete picture, they can inform on many aspects of the complex processes of RNA maturation. This chapter is based on parts of the publication of (Findeiß et al., 2011).

8.1 Introduction

Mature functional RNAs frequently deviate from their DNA templates. The maturation of a primary RNA transcript usually involves various forms of RNA processing (such as endo- and exonucleolytic trimming, splicing, or polyadenylation). More than a dozen mechanistically distinct types of RNA editing, i.e., targeted nucleotide insertions, deletions, and exchanges, have been described in a wide diversity of clades (Knoop, 2011). Chemical modifications, furthermore, introduce a variety of non-standard nucleotides and affect the majority of non-coding RNAs (ncRNAs) (Ishitani et al., 2008). As a consequence, a mature RNA sequence may differ substantially from its genomic DNA template. RNA editing and modification can have massive effects on both the secondary structure and the interpretation of mRNAs. Chemical modifications in tRNAs, for instance, are instrumental for the integrity of their 3D structures. A→I editing, on the other hand, influences protein sequences since I is read as G by the translation machinery.

Most eukaryotic and many prokaryotic RNAs undergo processing at their 3'-ends. Following cleavage or trimming of the primary transcript, additional nucleotides that are not encoded in the genome are added in many cases. The best-known examples are the polyadenylation of most mRNAs (Millevoi and Vagner, 2010) and the addition of CCA to the 3'-end of tRNAs (Phizicky and Hopper, 2010). Several ncRNAs, including signal recognition particle (SRP) RNA, U2 small nuclear RNA (snRNA) and 7SK RNAs are post-transcriptionally adenylated; U6 snRNA and ribosomal 5S RNA can be both adenylated and uridylated on their 3'-ends (Chen et al., 2000; Perumal et al., 2000; Perumal and Reddy, 2002). Several mature microRNAs are also 3'-adenylated and/or 3'-uridylated (Katoh et al., 2009; Lu et al., 2009; Ebhardt et al., 2009; Burroughs et al., 2010; Fernandez-Valverde et al., 2010).

High-throughput sequencing data from public data bases are likely to encompass unburied treasures. This gold rush, however, may be significantly slowed down since little is known about potential sources of error and bias. Even less is understood about the biology of many RNA species. Compared to DNA sequencing, cDNA sequences exhibit much higher error rates often resulting in frustrating alignment results. Depending on sequencing technology, cDNA preparation protocol, and organism under investigation about 20% of the sequences may not be alignable to the reference genome (Li et al., 2010). This may be caused by mismatches, insertions, or deletions, as well as strict mapping policies to purge reads with multiple hits in the reference genome. Disregarding technical artifacts in the RNA sequence read and errors or missing data in the reference genome, which make it impossible to map the read at all, there
are at least three reasons why RNA reads do not match exactly to the reference genome: (1) sequencing errors, (2) polymorphisms, and (3) RNA maturation. Hence, analysis of RNAseq data in general requires a significantly higher sensitivity in comparison to DNA variation analysis.

We were concerned with the question to what extent chemical modifications, editing, and non-encoded nucleotides in matured RNAs are visible in deep sequencing data. Previous work already indicates that such an approach is feasible: Analyzing reads that map with a single mismatch to the genome, more than 1000 sites with possible RNA base modifications were found in *Arabidopsis thaliana* and *Oryza sativa*, predominantly in tRNAs, microRNAs, and rRNAs (Iida *et al.*, 2009; Ebhardt *et al.*, 2009).

A primary source of information on ncRNAs are short read libraries that are prepared and analyzed with a focus on microRNAs. Here, total RNA is size-selected so that RNAs larger than 30 nt, and hence all complete “house-keeping” ncRNAs, are removed before sequencing. Surprisingly, these libraries contain a large number of reads deriving from nearly all ncRNA classes (Kawaji and Hayashizaki, 2008). These originate from cleavage of larger transcripts. For instance, tRNAs are under certain conditions specifically cleaved into fragments of different lengths in the anticodon loop or anticodon left arm (Lee and Collins, 2005; Li *et al.*, 2008b; Jöchl *et al.*, 2008). MicroRNA-sized products are derived from position specific processing at the 5'- or 3'-end of mature or precursor tRNAs (Cole *et al.*, 2009; Lee *et al.*, 2009).

Such small RNA fragments, for which in individual cases a microRNA-like function has been demonstrated, are also derived from small nucleolar RNAs (Taft *et al.*, 2009b), vault RNAs (Persson *et al.*, 2009; Stadler *et al.*, 2009), and Y RNAs (Meiri *et al.*, 2010), as well as some long ncRNAs such as MALAT1 (Stadler, 2010). The generation of these small RNA fragments is tied closely to the stable double-stranded regions in the parental RNA (Langenberger *et al.*, 2010). Here, we wanted to explore to what extent small RNA sequencing libraries are suitable for a systematic investigation of RNA maturation.

### 8.2 Materials and methods

**8.2.1 Data sets and mapping**

We use a combination of small RNA libraries, sequenced on an Illumina platform, from *Homo sapiens* (human) and *Macaca mulatta* (Rhesus macaque) brains, respectively (Somel *et al.*, 2010). The human library comprises 71,307,445 sequencing reads with an average length of 22.04 ± 1.03 nucleotides. With 114,619,534 reads the macaque data set is roughly twice as big as the human one and has a similar read length distribution with an average of 23.19 ± 3.37 nucleotides. To allow error-tolerant mapping of cDNA sequences (>14nt) we used the short read aligner *segemehl* with standard parameters and an E-value cutoff of 500 to also align short sequences (Hoffmann *et al.*, 2009). The *segemehl* software is able to detect mismatches,
insertions and deletions alike and reports multiple equally good scoring hits. Multiple best hits to the genome were explicitly allowed. When measuring levels of expression, the number of reads, represented by each tag was divided by the number of hits in the genome with equally good scores. This procedure ensures that the redundancy of multiple (nearly) identical copies (e.g. of tRNAs) is properly taken into account.

Genome sequences and annotation tracks were downloaded from the UCSC genome browser (Kent et al., 2002). Coding sequence (CDS) annotation was taken from the RefSeq gene tracks for both species. An RNA gene track was available for human only. The two primate species are so closely related, however, that all macaque homologs of human ncRNAs considered in this study are reliably identified by a simple blast search. MirBase (release 12) was used as source of pre-miRNAs as well as mature miR and miR* sequences and annotation.

8.2.2 CCA ends

To measure the activity of nucleotidyltransferases, tags ending with 3'-CCA were selected. The CCA was removed and the truncated tag was mapped to the reference genome. Tags with a genomically encoded CCA end downstream of the mapped tag were excluded from further analysis. Since short reads deriving from nuclear copies of mitochondrial DNA (NUMT) (Hazkani-Covo et al., 2010) and reads truly deriving from the mitochondrial DNA cannot be reliably distinguished in the data at hand, we also excluded all tags matching to the mitochondrial genome of the respective species. Overlapping tags passing the filtering steps were then joined into blocks. Finally, blocks representing less than 10 reads were excluded from further analysis (cf. Table 8.1).

8.2.3 tRNAs

The tRNAscan-SE program (downloaded from ftp://selab.janelia.org/pub/software/tRNAscan-SE/tRNAscan-SE-1.23.tar.Z) was applied to the reference genomes analyzed in this contribution. The predicted intact tRNAs and pseudogenes, respectively, were treated separately. Positions of tRNA modifications were extracted from the tRNAdb (Jühling et al., 2009). To safely map these modifications to the predicted tRNA genes and to avoid biases due to differently sized isoacceptor sequences only tRNA genes coding for the same amino acid and having the same length as those listed in the database were used. This set of mapped tRNA modifications was intersected with the tag variation data obtained from RNAseq read analysis. Raw counts of variant nucleotides were normalized by the number of tags mapping to the position with the variation.

To test whether post-transcriptional modifications are visible in RNAseq data, all blocks overlapping with tRNA 3’-ends were extracted and aligned at the tRNAse Z cleavage site. tRNA sequences, as well as the positions and type of the chemical modifications were retrieved from the tRNAdb.
8.3 Results

We have analyzed a combination of two RNA libraries obtained from *Homo sapiens* (human) and *Macaca mulatta* (Rhesus macaque) brains, respectively (Somel *et al.*, 2010).

Table 8.1: Statistics of the data sets used in this study. Reads with identical sequences were merged into tags. All tags matching to the mitochondrial genome of human and macaque, respectively, were removed to avoid contamination of nuclear copies of mitochondrial DNA (NUMT). Overlapping tags that survived the filtering steps were joined into blocks.

<table>
<thead>
<tr>
<th></th>
<th>Human</th>
<th>Macaque</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Entire library</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>reads</td>
<td>71,307,445</td>
<td>114,619,534</td>
</tr>
<tr>
<td>tags</td>
<td>355,453</td>
<td>14,240,332</td>
</tr>
<tr>
<td><strong>3'-CCA tails</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>tags</td>
<td>3,925</td>
<td>138,895</td>
</tr>
<tr>
<td>NUMT cleaned tags</td>
<td>3,017</td>
<td>118,298</td>
</tr>
<tr>
<td><strong>non-genomically encoded</strong> and NUMT filtered 3'-CCA tails</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tags</td>
<td>1,431</td>
<td>90,208</td>
</tr>
<tr>
<td>blocks</td>
<td>246</td>
<td>1,289</td>
</tr>
</tbody>
</table>

In our analysis we distinguish between individual reads and tags. The advantage of using tags lies in a drastic reduction of data that have to be handled. A statistical overview of the analyzed data sets is given in Table 8.1.

8.3.1 Inference of chemical modifications from mismatches

Some chemical modifications of nucleotides are detectable as mismatches between RNAseq data and the genomic reference. In contrast to PCR artifacts the mismatches appear in many different tags, and the frequency distribution of nucleotides deviates from that expected for SNPs. Two recent studies showed that tRNA modifications are detectable in plants (Iida *et al.*, 2009; Ebhardt *et al.*, 2009). The modification 5\* in Figure 8.1 shows that this is the case also in mammals, using the well-known 1-methyl-adenosine modification found at position 58 of many tRNAs (Roovers *et al.*, 2004) as an example.

The 1-methyl-adenosine modification is pivotal for the stability and thus the function of tRNAs (Anderson *et al.*, 1998). It has been reported that the methylated adenosine residue 58 serves as a pause signal for plus-strand strong-stop DNA synthesis and termination site during reverse transcription (Renda *et al.*, 2001).

This modification is the most prominent one that is directly visible from the superposition of the error profiles of all tRNAs. Several other modifications are detectable as conspicuous
Figure 8.1: Normalized read counts of coverage (gray) and variation (black) along a tRNA structure, given as dot-bracket notation. Gaps in the coverage within the T-loop and V-region are caused by gaps in the sequence alignment. Peaks (numbered stars) along the variation curve correspond to common tRNA modifications at the respective position. The modification 5\* within the T-loop corresponds to the 1-methyladenosine modification present in most tRNA sequences. Other modifications are N2-methylguanosine (1\*), 1-methylguanosine (2\*), 2-O-methylguanosine (3\*) and dihydrouridine (4\*).

accumulations of mismatches in individual tRNAs. Notably, most of the detectable positional variations are located either towards the 5’- or towards the 3’-end of the tRNA. This is caused by the very uneven coverage of tRNAs with small sequencing reads, which is heavily biased towards the ends and the fact that the error-prone 3’- and 5’-termini of sequencing reads naturally coincide with the ends of the tRNA (see Figure 8.1). Besides the very strong effect of 1-methyladenosine on the accuracy of the cDNA, RNAseq data additionally exhibits moderately increased error rates for dihydrouridines and methylguanosine modifications such as N2-methylguanosines. This is consistent with the findings that the major substitution sites in plant tRNAs correspond to known RNA base modifications: N1-methyladenosine (m1A), N2-methylguanosine (m2G), and N2,N2-methylguanosine (m22G) (Iida et al., 2009; Ebhardt et al., 2009).

8.3.2 Inference of chemical modifications from read patterns

Some nucleotide modifications act as road blocks for the reverse transcriptase (Motorin et al., 2007). Thus we expect to observe non-random termination of DNA products from the initial reverse transcription of the RNA. Since the sequencing protocols used to generate the data that we analyzed here are strand-specific, sequencing reads are reported in the reading
Figure 8.2: Number of tags covering a given sequence position in a human tRNA-Glu gene. Known chemical modifications are indicated below the genomic reference sequence. (See Table 8.2 for the key to symbols mapping.) The most prominent modification, the N2-methylguanosine (L) at position 10, is detectable as a G-to-T transversion in 23% of all tags as well as a sharp increase of read starts at the following position. The block of tRNA 3'-end reads including the genomically not encoded CCA is indicated. Also note that the read coverage is smallest around the anti-codon.

direction of the original RNA in the sample. An obstacle in the reverse transcription step thus results in the enrichment of mapped read starts at the position following the chemical modification (Motorin et al., 2007). As shown in Figure 8.2, this leads to an upward jump of the read and tag coverage at the position following the modification.

In order to determine whether this effect can be seen in the analyzed RNAseq data, we compared the start positions of reads with the positions of known modifications in human tRNA sequences compiled in the tRNAdb (Jühling et al., 2009). For example, we observe a nearly 7-fold enrichment of read starts on tRNA position 59 compared to the modified position 58 (Table 8.2), corresponding to reverse transcription products that terminate before the modified base. Some of these reads extend beyond the tRNase Z processing site and hence derive from the unprocessed precursor. This suggests that these modifications might precede the formation of the 3'-terminus.

A road block function of several modifications is observable in our data in particular for modifications close to the 5'- and 3'-ends of the tRNA. For instance, the N4-acetylcytidines and N2-methylguanosines modifications, which are located close to the 5'-end of mature tRNAs, are detectable by a high incidence of reads starting immediately downstream of the site of modification. Many of the more centrally located modifications are not detectable. This bias is largely caused by the imbalance in the read coverage, which is much higher towards the 5'- and 3'-ends of tRNA (Figure 8.1).

8.3.3 Processing of immature and mature tRNAs

The analysis of tRNA loci, surprisingly, shows evidence for the production of small RNA species not only from mature tRNAs but also from unprocessed precursors. This is evidenced, on the one hand, by reads with CCA ends extending the genomically encoded 3'-end and reads
Table 8.2: Patterns of read starts and tRNA modifications. The first two columns give the common name of the modification and its RNAMods abbreviation (Dunin-Horkawicz et al., 2006); #: number of experimentally verified modifications in human tRNAs; pos: median position of the modification within tRNAs; exp: number of genomic loci for which a modification is expected and reads are mapped. The last column (ratio) gives the number of read starts one nt downstream of the modification divided by the number of read starts observed at the modified position. Road block modifications that impair reverse transcription are expected to exhibit large ratios.

<table>
<thead>
<tr>
<th>modification</th>
<th>*</th>
<th>#</th>
<th>exp</th>
<th>pos</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>N-acetylcytidine</td>
<td>M</td>
<td>2</td>
<td>38</td>
<td>2.5</td>
<td>12.80</td>
</tr>
<tr>
<td>5-methylcytidine</td>
<td>1</td>
<td>17</td>
<td>298</td>
<td>49</td>
<td>10.70</td>
</tr>
<tr>
<td>1-methyladenosine</td>
<td>L</td>
<td>12</td>
<td>164</td>
<td>9.5</td>
<td>2.66</td>
</tr>
<tr>
<td>N2-methylguanosine</td>
<td>K</td>
<td>4</td>
<td>52</td>
<td>19</td>
<td>1.02</td>
</tr>
<tr>
<td>1-methylguanosine</td>
<td>T</td>
<td>6</td>
<td>78</td>
<td>54</td>
<td>1.60</td>
</tr>
<tr>
<td>5-methyluridine</td>
<td>D</td>
<td>6</td>
<td>87</td>
<td>46</td>
<td>0.45</td>
</tr>
<tr>
<td>pseudouridine</td>
<td>B</td>
<td>4</td>
<td>53</td>
<td>32.5</td>
<td>0.08</td>
</tr>
<tr>
<td>dihydrouridine</td>
<td>J</td>
<td>5</td>
<td>82</td>
<td>33</td>
<td>0.07</td>
</tr>
<tr>
<td>2-O-methylcytidine</td>
<td>4</td>
<td>46</td>
<td>26</td>
<td></td>
<td>0.02</td>
</tr>
</tbody>
</table>

showing the hallmarks of chemical modifications, and on the other hand, by reads spanning across the RNase P (5') and RNase Z (3') cleavage sites (Figure 8.3). Lee et al. (2009) discovered three types of these short RNA fragments: tRF5 and tRF3 sequences are located at the 5'- and 3'-ends of the mature tRNAs, respectively. tRF5 sequences have the RNase P cleavage site at their 5'-end, while tRF3s have a CCA end at the correct position following the tRNase Z processing site. Thus they derive from a matured tRNA. In contrast, tRF1 sequences are entirely located in the 3'-part of the precursor that is cleaved off by tRNase Z. A detailed study (Haussecker et al., 2010) showed that such tRF1-like small RNAs are involved in the global regulation of RNAi, suggesting that many of them could be functional.

In order to obtain at least a rough relative quantification of mature versus precursor processing we quantified the fraction of reads that derived from mature tRNAs and their precursors, respectively (Figure 8.3). While the tRFs arising from the mature tRNA dominate in most cases, the situation is different for tRNA-Ile-TAT, tRNA-Leu-CAA, tRNA-Leu-TAG, tRNA-Ser-TGA, tRNA-Thr-AGT, and tRNA-Thr-CGT. Similar data were obtained for human and macaque. When comparing relative expression of tRNAs one would expect that the read counts, normalized relative to the overall coverage and multiple mappings, of different species
Figure 8.3: Processing of mature tRNAs and their precursors in human. (A) different types of read blocks derive from different processing stages of tRNAs. Blocks shown as filled boxes are assumed to derive from mature tRNA molecules after RNase P and Z processing. Blocks illustrated as open boxes are located completely or partially outside the mature tRNA region and hence are derived from precursors. The origin of internal reads (grey boxes) cannot be assigned to either mature or precursor tRNA molecules. The classes tRF5, tRF3, and tRF1 of tRNA-derived small RNA fragments were defined in (Lee et al., 2009). (B) fraction of reads mapping to a tRNA locus that are derived from mature tRNAs (black) or precursor sequences (white).

should show similar expression patterns. There are, however, several significant differences observed between macaque and human (Figure 8.4).

Interestingly, the most abundant human tRF1-type small RNA fragment, tRF-1001 deriving from tRNA-Ser-TGA, behaves very differently between human and macaque. Lee et al. (2009) observed that tRF-1001 is expressed highly in a wide range of cancer cell lines, where its expression is tightly correlated with proliferation. While the precursor-derived products dominate the human library, most of the macaque sequences arise from the mature tRNA, suggesting that tRF-1001 might be a very recent innovation in human evolution. One might speculate that, like other evolutionarily very recent ncRNAs such as BC1 and BC200 (Kon-drashov et al., 2005) or HAR1 (Pollard et al., 2006), at least some of the tRFs detected in brain RNA libraries have functions in brain development.
Figure 8.4: Differences in relative expression of tRFs derived from mature tRNAs (●) and precursors (×) between human and macaque. The tRNAs with the largest deviations between the two species are labeled.
Non-coding RNAs largely influence the behavior of cells by fine-tuning the expression of a large amount of genes. Small errors in their sequences can lead to major mis-regulations and thus diseases. High-throughput sequencing offers the unique opportunity to find these errors and helps to better understand the regulatory mechanism. We have shown that RNA editing events and RNA modifications can be found using short RNA-seq data. By just looking at RNA-seq data, it is still not possible to definitely say, if the found mutation comes from RNA editing, or if it is encoded on the DNA and thus a single nucleotide mutation (SNP). To make this final statement, one has to sequence the DNA of the individual and see, which nucleotide is encoded at the position of interest. Nevertheless, using RNA-seq it is possible to measure these mutations and generate hypothesis about possible consequences, like a loss of binding for microRNAs and thus a mis-regulation of targeted genes.

Furthermore, in this work I showed how HTS can be used to not only validate already annotated ncRNAs, but also to find new gene candidates. I discovered that different ncRNA classes form specific read patterns after mapping the reads back to a reference genome. Most of these patterns can be explained by secondary structures of longer pre-ncRNA molecules which are found by enzymes and cut in a very specific way. As shown in the last chapter, another explanation might be the blocking of the reverse transcriptase used for the library preparation by post-transcriptionally modified nucleotides. This technically induced bias results in an accumulation of reads starting at the modified nucleotide. If the patterns with no clear structural or chemical explanation are just degradation products, or if they are processed by other enzymes, is subject to further investigation. The full understanding of the genesis
of read patterns has to be a goal for future work. The experiment analyzed in chapter 7, in which Dicer was knocked down to study its processing targets, is a first step in this direction. It highlighted some microRNAs, which are not cut by this enzyme. How these microRNAs are maturated is still unknown and thus this experiment is just a starting point. Finding other enzymes or even completely new pathways that process this set of microRNAs seems to be of interest. Nevertheless, the resulting read patterns are highly non-random and several applications, like alignment search tools or approaches evaluating pattern conservation, are thinkable to improve the results of already used prediction algorithms.

The developed algorithms for ncRNA prediction and clustering using the read patterns can help to find new types of ncRNAs or to better understand the known ones. The finding of microRNA-like RNAs, processed out of tRNAs or snoRNAs, resulted in a rethinking of discarding mapped reads overlapping with annotated ncRNAs before data analysis. Since these two types of ncRNAs do not only function in their traditional way, but also regulate the expression of hundreds of genes, they should not be filtered out by default. They should be analyzed separately, and if hints for functionality are found, be included in downstream analyses. To my knowledge, there are no publications including microRNA-like processing products, like moRNAs, tRFs, or sdRNAs in their microRNA studies. As shown in chapter 5, some of the known examples are found as mis-annotated microRNAs in miRBase, but most of these special kinds of regulating RNAs are yet unknown.

The results received when using a random forest classifier, trained on features extracted from read patterns, confirms the observation, that these patterns are specific for different ncRNA types. The resulting DARIO web server, that allows researchers to classify tRNAs, snoRNAs, and microRNAs in their own short RNA-seq dataset, is just a first step in the direction of analyzing short RNAseq data. There still is a lot of potential in this algorithm. One might think of introducing other types of ncRNAs, like vault RNA, Y-RNAs, snRNAs, and many more. We did not include these types of ncRNAs, since they were not expressed high enough to measure them accurately. But with the improvement of HTS, i.e. the ultra-high throughput of current HiSeq2000 machines (illumina), this should not be a problem any more. Another possibility would be the adoption of new features. We just used features given by the read patterns, since the intention was to prove their uniqueness. Adding secondary structure or conservation information to the algorithm will surely improve the results.

The developed alignment algorithm deepBlockAlign assigns similarity measures to read patterns and thus allows a clustering of ncRNAs with microRNA-like read patterns together with real microRNAs. Thus, this approach helps finding microRNA-like molecules and including them in downstream analyses, like microRNA:mRNA target site predictions and differential
expression analyses for building up or refining regulatory networks. The resulting clusters can also help to find sub-clusters, like microRNA-offset RNAs, within the microRNA cluster. The alignment algorithm just opened a new way of looking at ncRNA read patterns, comparing them and understanding them. It also offers a great opportunity to handle completely unknown, but expressed loci. Using unsupervised clustering may also improve the clustering for finding new classes of ncRNAs, which then appear in completely separated clusters. There are two major ideas of how \texttt{deepBlockAlign} can be used in future applications: 1) All expressed loci overlapping with known ncRNAs are flagged, all patterns are aligned against all and clustered using a clustering approach. By investigating annotated ncRNAs, with which the unknown patterns clustered together, an assumption about the type of ncRNA can be made. Another potential approach is 2) a searching tool. Having a huge database of known read patterns stored, one can easily use the alignment algorithm to find these patterns, showing the highest similarity to the unknown one. These two examples highlight the potential of the algorithm in better understanding unknown ncRNA candidates.


Curriculum Scientiae

EDUCATION:

since 01/2009 PhD student at University of Leipzig
• Group of Prof. Peter F. Stadler, Chair of Bioinformatics
• Thesis: *High Throughput Sequencing and small non-coding RNAs*

10/2008 – 12/2008 Master course student of bioinformatics at Technical University of Munich and Ludwig-Maximilians-University, Germany

10/2003 – 06/2008 Diploma student in bioinformatics at Fachhochschule Weihenstephan in Freising, Germany
• Diplom-Ingenieur (FH)
• Thesis: *Reduced RNAi off-target effects by artificially designed siRNAs in mammals*

WORKING EXPERIENCE:

since 09/2012 Chief Executive Officer (CEO) of *ecSeq Bioinformatics*
• Bioinformatics solution provider offering the analysis of high-throughput sequencing data

since 08/2011 Member of the Program Committee of the Leipzig Research Centre for Civilization Diseases (LIFE)
06/2008 – 06/2010
Scientific assistant at the Technical University of Munich
• Group of Prof. Dmitrij Frishman, Department of Genome-Oriented Bioinformatics
• Project: siRNA and microRNA research

09/2006 – 02/2007
Internship at Roche Palo Alto, USA
• Group of Dr. Guochun Liao, Department of Genetics and Genomics
• Project: Polymorphisms in microRNA binding sites and pre-microRNAs in mouse

03/2005 – 08/2005
Internship at New York University, New York City, USA
• Group of Prof. Dr. Nikolaus Rajewsky, Center for Comparative Functional Genomics
• Project: Post transcriptional gene regulation with microRNAs in flies and humans.

01/2004 – 03/2004
Internship at Siemens Medical Solutions, Germany
• Project: Programming in C for the software of the directif Lab Chip System

**IT-KNOWLEDGE:**

- Operating systems: UNIX, Mac, Linux, Windows
- Programming: C, C++, Java, Perl, R, PHP, PostScript, Pascal, BASIC
- Markup languages: Latex, HTML
- Database systems: MySQL

**LANGUAGE SKILLS:**

- German: native speaker
- English: fluent
Publications

JOURNALS:


Recurrent mutation of the ID3 gene in Burkitt lymphoma identified by integrated genome, exome and transcriptome sequencing. * Nature genetics 11;44(12):1316-1320. * authors contributed equally


**Multi-platform next generation sequencing of the domestic turkey (Meleagris gallopavo): Genome assembly and analysis.** *PLoS Biology.* 8,9: e1000475.


**Identification and Classification of Small RNAs in Transcriptome Sequence Data** *Pac Symp Biocomput.* 2010:80-7.


BOOK CHAPTER:


CONFERENCE / SEMINARS:

2012 Bio-IT World Expo Europe - Cambridge Healthtech Institute (Presenter)
Langenberger D: *DARIO - analysis of small RNAs sequencing data*
10/2012; Vienna, Austria

Meeting on Advances and Challenges of RNA-Seq Analysis (Presenter)
Langenberger D: *DARIO - analysis of small RNAs sequencing data*
06/2012; Halle/Saale, Germany

RECOMB 2012 (Poster)
Fasold M, Langenberger D, Binder H, Stadler PF and Hoffmann S.: *DARIO: A ncRNA detection and analysis tool for next-generation sequencing experiments.*
04/2012; Barcelona, Spain

Workshop: Programming for Evolutionary Biology (Presenter)
Langenberger D: *Analysis of Next Generation Sequencing Data*
03/2012; Leipzig, Germany

27th TBI winter seminar 2012 (Presenter)
Langenberger D: *W.W.D.D - What Would Dicer Do?*
02/2012; Bled, Slovenia

ICGC MMML-Seq-Meeting 2011 (Participant)
Langenberger D:
10/2011; Cambridge, UK

EBI - RNA-seq workshop (Participant)
Langenberger D:
09/2011; Cambridge, UK

4th Berlin Summer Meeting (Participant)
Langenberger D:
06/2011; Berlin, Germany
6th Microsymposium on Small RNAs (Participant)
Langenberger D:
05/2011; Vienna, Austria

TBI Theoretical Biochemistry Group (Presenter)
Langenberger D: DARIO (A Web-service for deep sequencing data)
05/2011; Vienna, Austria

14th LIFE day, University Leipzig (Presenter)
Langenberger D: DARIO (A Web-service for deep sequencing data)
05/2011; Leipzig, Germany

26th TBI winter seminar 2011 (Presenter)
Langenberger D: DARIO (A Web-service for deep sequencing data)
02/2011; Bled, Slovenia

RTH seminar, Center for non-coding RNA in Technology and Health, University of Copenhagen (Presenter)
Langenberger D: Analysis of Short Read Libraries
05/2011; Copenhagen, Denmark

RECOMB 2010 (Poster)
Bermudez-Santana C, Langenberger D, Hoffmann S. and Stadler PF: Searching tRNA processing patterns in transcriptome sequencing data.
08/2010; Lisbon, Portugal

25th TBI winter seminar 2010 (Presenter)
Langenberger D: Detecting new miRNAs from deep sequencing data: A field study in worm
02/2010; Bled, Slovenia

Systembiologie/F-Seminar, IZBI Leipzig (Presenter)
Langenberger D: Classification and Identification of Non-coding RNAs using High Throughput Sequencing Data
10/2009; Leipzig, Germany

Bioinformatics autumn seminar 2009 (Presenter)
Langenberger D: Classification and Identification of Non-coding RNAs using High Throughput Sequencing Data
10/2009; Vysoka Lipa, Czech Republic
8th Leipzig Research Festival (Poster)
Langenberger D, Hoffmann S., Bermudez-Santana C. and Stadler PF: *Classification and Identification of Non-coding RNAs using High Throughput Sequencing Data*
12/2009; Leipzig, Germany

Conference on Next Generation Sequencing 2009 (Participant)
Langenberger D:
10/2009; Barcelona, Spain

German Conference on Bioinformatics 2009 (Poster)
Langenberger D, Hoffmann S., Bermudez-Santana C. and Stadler PF: *Classification and Identification of Non-coding RNAs using High Throughput Sequencing Data*
09/2009; Halle/Saale, Germany

Otto Warburg International Summer School and Workshop on Regulatory (Epi-)Genomics (Participant)
Langenberger D:
09/2009; Berlin, Germany

24th TBI winter seminar 2009 (Presenter)
Langenberger D: *Into the Deep: microRNA Detection using Next-Generation Sequencing Data*
02/2009; Bled, Slovenia

COLLABORATIONS:

Prof. Dr. Daniel Teupser, Institute of Laboratory Medicine
LMU Munich; Munich, Germany

Prof. Dr. Jan Gorodkin, Center for non-coding RNA in Technology and Health
University of Copenhagen; Copenhagen, Denmark

REVIEWER ACTIVITIES:

- Bioinformatics, Oxford Journals
- Nucleic Acids Research, Oxford Journals
- BMC Genomics, BioMed Central
- Transactions on Computational Biology and Bioinformatics, IEEE/ACM
- Frontiers in Non-Coding RNA, Frontiers Media S.A.
Selbständigkeitserklärung


(David Langenberger)