A Study in RNA Bioinformatics

Identification, Prediction and Analysis

EVA FREYHULT
Dissertation presented at Uppsala University to be publicly examined in B42, BMC, Husargatan 3, Uppsala, Friday, December 7, 2007 at 13:00 for the degree of Doctor of Philosophy. The examination will be conducted in English.

Abstract

Research in the last few decades has revealed the great capacity of the RNA molecule. RNA, which previously was assumed to play a main role only as an intermediate in the translation of genes to proteins, is today known to play many important roles in the cell in addition to that as a messenger RNA and transfer RNA, including the ability to catalyze reactions and gene regulations at various levels.

This thesis investigates several computational aspects of RNA. We will discuss identification of novel RNAs and RNAs that are known to exist in related species, RNA secondary structure prediction, as well as more general tools for analyzing, visualizing and classifying RNA sequences.

We present two benchmark studies concerning RNA identification, both de novo identification/characterization of single RNA sequences and homology search methods.

We develop a novel algorithm for analysis of the RNA folding landscape that is based on the nearest neighbor energy model adopted in many secondary structure prediction programs. We implement this algorithm, which computes structural neighbors of a given RNA secondary structure, in the program RNAbor, which is accessible on a web server.

Furthermore, we combine a mutual information based structure prediction algorithm with a sequence logo visualization to create a novel visualization tool for analyzing an RNA alignment and identifying covarying sites.

Finally, we present extensions to sequence logos for the purpose of tRNA identity analysis. We introduce function logos, which display features that distinguish functional subclasses within a large set of structurally related sequences, as well as the inverse logos, which display underrepresented features. For the purpose of comparing tRNA identity elements between different taxa we introduce two contrasting logos, the information difference and the Kullback-Leibler divergence difference logos.

Keywords: RNA, bioinformatics, secondary structure, structure prediction, dynamic programming, energy landscape, homology search, sequence logo, tRNA

Eva Freyhult, The Linnaeus Centre for Bioinformatics, Box 598, Uppsala University, SE-75124 Uppsala, Sweden

© Eva Freyhult 2007

ISSN 1651-6214
urn:nbn:se:uu:diva-8305 (http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-8305)
List of Papers

This thesis is based on the following papers, which are referred to in the text by their Roman numerals.


* The authors contributed equally to the paper.

Reprints were made with permission from the publishers.

The following papers are not included in the thesis.

- Freyhult, E., Prusis, P., Lapinsh, M., Wikberg, J.E.S., Moulton, V. and Gustafsson, M.G. (2005), Unbiased Descriptor and Parameter Selection

## Contents

1 Introduction ......................................................... 7  
   1.1 Biological background ........................................ 7  
      1.1.1 The Central Dogma .................................... 8  
      1.1.2 ncRNA examples ...................................... 13  
   1.2 RNA structure ................................................ 14  
      1.2.1 Secondary structure .................................. 15  
      1.2.2 RNA evolution ...................................... 19  
2 Identification ...................................................... 21  
   2.1 Protein gene finding ....................................... 21  
   2.2 RNA gene finding .......................................... 22  
      2.2.1 *Ab initio* identification ......................... 22  
      2.2.2 Comparative approach .............................. 23  
   2.3 Homology search .......................................... 23  
      2.3.1 Homology search methods ......................... 24  
3 Prediction ........................................................ 27  
   3.1 Secondary structure prediction algorithms ............... 27  
      3.1.1 Maximization of number of base-pairs ............. 28  
      3.1.2 Minimal free energy (MFE) minimization ........... 30  
      3.1.3 Suboptimal folding .................................. 33  
      3.1.4 Partition function ................................ 33  
      3.1.5 Pseudoknots ....................................... 36  
   3.2 Comparative methods .................................... 36  
      3.2.1 Alignment folding .................................. 36  
      3.2.2 Simultaneous alignment and folding ............... 38  
      3.2.3 Aligning secondary structures .................... 38  
4 Analysis .......................................................... 39  
   4.1 Analysis of energy landscape .............................. 39  
      4.1.1 Thermodynamic stability of functional RNA ...... 40  
      4.1.2 Uniqueness of fold ................................ 41  
      4.1.3 Measures of well-definedness ..................... 42  
      4.1.4 Software exploring the energy landscape ........ 43  
   4.2 Analysis of function .................................... 46  
      4.2.1 Information ........................................ 47  
      4.2.2 Covariation measures .............................. 48  
      4.2.3 Sequence logos .................................... 50  
      4.2.4 Structure logos .................................... 51
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.5 MIfold</td>
<td>51</td>
</tr>
<tr>
<td>4.2.6 tRNA logos</td>
<td>53</td>
</tr>
<tr>
<td>5 Summary of papers</td>
<td>57</td>
</tr>
<tr>
<td>6 Sammanfattning på svenska</td>
<td>61</td>
</tr>
<tr>
<td>7 Acknowledgements</td>
<td>65</td>
</tr>
<tr>
<td>References</td>
<td>67</td>
</tr>
</tbody>
</table>
1. Introduction

Until the 1980’s RNA was believed to mainly function as intermediates in the translation of genes to proteins. However, RNA has many more roles than those of messenger RNA (mRNA) and transfer RNA (tRNA). In the early 1980’s Altman and Cech independently discovered catalytic RNAs (Cech, 1989; Altman et al., 1989). For their discoveries they were awarded the Nobel prize in Chemistry in 1989. Since then the interest in RNA has been growing and today RNA is known to have many important functions in the cell. For example, RNA is involved in translation where it catalyzes the peptidyl-transferase reaction in peptide bond formation (Weinger et al., 2004; Nissen et al., 2000) and splicing (Vicens and Cech, 2006); both are examples of reactions where RNA functions catalytically and the RNA molecules that carry out these reactions are now termed ribozymes (ribonucleic enzymes) (Doudna and Cech, 2002). RNA is also known to be involved in both transcriptional and translational gene regulation, as well as post-transcriptional regulation by RNA interference (Fire et al., 1998). In RNA interference, the regulation is performed by small interfering RNAs (siRNAs) (Harborth et al., 2003; Tuschl, 2003; Tucker and Breaker, 2005) and microRNAs (miRNAs) (Lim et al., 2003). In 2006, Fire and Mello were awarded the Nobel Prize in Physiology or Medicine 2006 for their discovery of RNA interference and gene silencing by double-stranded RNA. In addition, RNA is known to play critical roles in various other cellular mechanisms such as dosage compensation (Brown et al., 1992), protein shuttling (Walter and Blobel, 1982), as well as retranslation events such as selenocysteine insertion (Commans and Bock, 1999) and programmed ribosomal frameshifting (Bekaert et al., 2003; Moon et al., 2004).

In this thesis we demonstrate the use of bioinformatics to study RNA sequence, structure and function as well as the relationships among these three.

1.1 Biological background

This section gives a short overview of basic biological processes, such as transcription and translation, in which we point out where RNA plays important roles.
1.1.1 The Central Dogma

The flow of (genetic) information in the cell is traditionally described in the central dogma of molecular biology, see Figure 1.1. The genome (DNA) encodes the sequence information for all the proteins synthesized by the cell. The segment of DNA that holds the information of how to construct a certain protein, is called a protein-coding gene. However, DNA is not directly the template in the protein synthesis. DNA is transcribed by an enzyme, RNA polymerase, into a messenger RNA (mRNA) carrying the same information as the transcribed gene. The mRNA is then translated into protein by the ribosome.

A gene can also be transcribed into an RNA that is never translated into a protein. Such genes are called non-coding genes and the RNAs are called functional or non-coding RNAs (ncRNAs), as they are not translated into protein, but they have a function themselves. Figure 1.2 shows a modified version of the central dogma, in which a few ncRNAs and their functions are indicated.

**Biological macromolecules**

**DNA**

The genetic code is stored in DNA (deoxyribonucleic acid), a heteropolymer1 of deoxyribonucleotides. A deoxyribonucleotide consists of a phosphate group, a sugar molecule, more specifically deoxyribose, and one of four nitrogenous bases, see Figure 1.3 (a). The nitrogenous bases are two purines,

1A heteropolymer is a polymer where the building blocks are not all identical.
adenine (A) and guanine (G), and two pyrimidines, cytosine (C) and thymine (T), see Figure 1.4 (a)-(d).

Figure 1.3: The building blocks of DNA and RNA (a) deoxyribonucleotide, and (b) ribonucleotide, respectively. A nucleotide consists of a five-carbon sugar, deoxyribose or ribose, a phosphate group and one of four nitrogenous bases: Adenine, Cytosine, Guanine and Thymine (DNA) or Uracil (RNA). In this figure the base Thymine is shown attached to the deoxyribose and Uracil is shown attached to the ribose.

Figure 1.4: The nitrogenous bases. The purines adenine and guanine as well as the pyrimidines cytosine, thymine and uracil.

The sugar and the phosphate group form the backbone of DNA, whereas the bases carry the genetic information. The nucleotides are connected in the backbone with a phosphodiester bond between the 3'-hydroxyl in the sugar and the 5'-hydroxyl in the sugar in the next nucleotide. This type of bond gives the DNA molecule a direction, one end of the chain has a free 5'-OH (the 5' end), the other has a free 3'-OH (the 3' end)). DNA is transcribed in the 5' → 3' direction and when printing a DNA sequence it’s printed in the
5’ → 3’ direction. Hence, the ACGTAC and CATGCA refer to two different molecules, the former with adenine in its 5’ end.

The bases can base-pair to each other with hydrogen bonds. These base pairs are specific, and in DNA, adenine pairs with thymine and guanine with cytosine. This unique complementarity is important to both the structure and function of DNA. DNA has a double-helical structure where two complementary DNA strands are coiled around a common axis with the nitrogenous bases on the inside. The two chains are held together by hydrogen bonds between the complementary bases. As the DNA is replicated, the double-helix is opened up and a polymerase synthesizes a complementary sequence to each of the now single-stranded parts of the DNA. In the replication process mistakes are made, but thanks to the unique complementarity of DNA, these can to some degree be detected and corrected.

**RNA**

RNA (ribonucleic acid) is a heteropolymer similar to DNA. RNA is built of four different kinds of ribonucleotides. A ribonucleotide consists of a phosphate group, a sugar, and one out of four nitrogenous bases, just like DNA. However, the chemical structure of RNA differs from DNA in two ways. RNA has a different five-carbon sugar than DNA, ribose instead of deoxyribose and the base thymine (T) in DNA is in RNA replaced by uracil (U), see Figure 1.3 and 1.4.

The RNA molecule has a broad repertoire of functions. The most well-known is as a messenger. The messenger RNA (mRNA) carries a working copy of a gene and is translated by the ribosome into protein. However, this thesis will focus on the non-coding RNAs, see section 1.1.2 for a few examples of ncRNAs.

**Protein**

Proteins have a vast range of functions in the cell, there are e.g. structural proteins, regulating proteins, and enzymes that catalyze biochemical reactions. A protein is a heteropolymer, built up by amino acids. There are 20 canonical amino acids in nature, with varying properties (charge, polarity, size etc). An amino acid consists of an amino group (-NH₂), a carboxyl group (-COOH), a hydrogen atom (H) and a variable side chain, all covalently coupled to an α-carbon (Cα) atom, see Figure 1.5. (The Cα is so called because it is adjacent to the acidic carboxyl group.)

In proteins, the amino acids are connected in peptide bonds formed between the carboxyl group in one amino acid and the amino group in another. Several amino acids in a chain form a polypeptide or a protein. The Cα atom, the carboxyl carbon atom, and the nitrogen atom in the amino group form the backbone of the protein, whereas the variable parts are the side chains. The protein molecule will in one end have a free amino group (the N-terminal), in the other a free carboxyl group (the C-terminal), this gives the protein di-
rectionality. A protein sequence is always read from the N-terminal to the C-terminal.

Protein structure can be defined at several different levels. The primary structure is the amino acid sequence, read from the N-terminal to the C-terminal. The secondary structure of a protein describes local conformations in the protein, like \( \alpha \)-helices and \( \beta \)-strands. Tertiary structure describes interactions between amino acids that are further apart in the protein sequence and how the secondary structure is spatially arranged. The quaternary structure describes how different subunits in a protein are arranged (if the protein consists of more than one linear sequence).

**Transcription**

Transcription is the cellular process in which DNA is transcribed by an RNA polymerase to form a complementary RNA copy. The transcription starts at promoters in the DNA. The promoters are certain sequence motifs important for recognition by RNA polymerase. When the RNA polymerase is bound to the promoter, a small part of the DNA helix is unwound and transcription starts. The RNA polymerase reads the DNA sequence, always from 5' to 3' end, and as it reads the sequence the RNA polymerase synthesizes an RNA molecule complementary to the DNA template. The transcription continues until a terminator sequence is reached. A protein can act in the transcription termination or, as e.g. in *E.coli*, the termination signal can be an RNA hairpin formed in the newly synthesized RNA followed by a sequence of U's. The hairpin is formed between complementary sequences high in G and C and causes the RNA polymerase to terminate the transcription.

**Pre-processing**

Many transcribed RNAs need to undergo certain pre-processing reactions before they can function as mature mRNA or ncRNA. Most eukaryotic genes consist of both exons (protein coding regions) and introns (non-protein-coding regions). Both introns and exons are transcribed into a pre-mRNA. The introns are cut out from the mRNA in a process called splicing by the spliceosome, a complex consisting of small nuclear RNAs (snRNAs) and proteins. Also RNA editing of precursor RNAs is common in the form of base substitutions (modifications), insertions, or deletions. Some of these modifications are mediated by enzymes (catalytic proteins), others by guide RNAs, such as the small nucleolar RNAs (snoRNAs). In eukaryotes the rRNAs, snRNAs and the
snoRNAs carry a large number of modified nucleotides that are a result of RNA editing.

The mature mRNA contains the protein-coding sequence, the open reading frame (ORF), but flanking the ORF are untranslated regions (UTR), the 5′-UTR upstream of the ORF and the 3′-UTR downstream. Both the 5′- and the 3′-UTR of eukaryotic mRNAs have been shown to often contain elements important for gene regulation.

**Translation**

The genetic code describes the relation between the DNA (or mRNA) sequence and the protein sequence. A sequence of three nucleotide bases, a *codon*, specifies an amino acid, see table 1.1. The translation always starts at a specific codon, the start codon AUG, continuing in the 5′ → 3′ direction until it reaches one of three stop codons. The translation of the mature mRNA involves several ncRNAs. The translation take place in the ribosome, a large complex of ribosomal RNAs (rRNAs) and proteins. The codons are read one-by-one by transfer RNAs (tRNAs), adapter molecules carrying specific amino acids, that in the ribosome are connected with peptide bonds to form a protein.

<table>
<thead>
<tr>
<th>Codon</th>
<th>Amino Acid</th>
</tr>
</thead>
<tbody>
<tr>
<td>UUU</td>
<td>Phe</td>
</tr>
<tr>
<td>UUC</td>
<td>Phe</td>
</tr>
<tr>
<td>UUA</td>
<td>Leu</td>
</tr>
<tr>
<td>UUG</td>
<td>Leu</td>
</tr>
<tr>
<td>CUU</td>
<td>Leu</td>
</tr>
<tr>
<td>CUC</td>
<td>Leu</td>
</tr>
<tr>
<td>CUA</td>
<td>Leu</td>
</tr>
<tr>
<td>CUG</td>
<td>Leu</td>
</tr>
<tr>
<td>AUU</td>
<td>Ile</td>
</tr>
<tr>
<td>AUC</td>
<td>Ile</td>
</tr>
<tr>
<td>AUA</td>
<td>Ile</td>
</tr>
<tr>
<td>AUG</td>
<td>Met*</td>
</tr>
<tr>
<td>GUU</td>
<td>Val</td>
</tr>
<tr>
<td>GUC</td>
<td>Val</td>
</tr>
<tr>
<td>GUA</td>
<td>Val</td>
</tr>
<tr>
<td>GUG</td>
<td>Val</td>
</tr>
<tr>
<td>UCU</td>
<td>Ser</td>
</tr>
<tr>
<td>UCC</td>
<td>Ser</td>
</tr>
<tr>
<td>UCA</td>
<td>Ser</td>
</tr>
<tr>
<td>UCG</td>
<td>Ser</td>
</tr>
<tr>
<td>CCU</td>
<td>Pro</td>
</tr>
<tr>
<td>CCC</td>
<td>Pro</td>
</tr>
<tr>
<td>CCA</td>
<td>Pro</td>
</tr>
<tr>
<td>CCG</td>
<td>Pro</td>
</tr>
<tr>
<td>ACU</td>
<td>Thr</td>
</tr>
<tr>
<td>ACC</td>
<td>Thr</td>
</tr>
<tr>
<td>ACA</td>
<td>Thr</td>
</tr>
<tr>
<td>ACG</td>
<td>Thr</td>
</tr>
<tr>
<td>GCU</td>
<td>Ala</td>
</tr>
<tr>
<td>GCC</td>
<td>Ala</td>
</tr>
<tr>
<td>GCA</td>
<td>Ala</td>
</tr>
<tr>
<td>GCG</td>
<td>Ala</td>
</tr>
<tr>
<td>UAU</td>
<td>Tyr</td>
</tr>
<tr>
<td>UAC</td>
<td>Tyr</td>
</tr>
<tr>
<td>UAA</td>
<td>stop</td>
</tr>
<tr>
<td>UAG</td>
<td>stop</td>
</tr>
<tr>
<td>CAU</td>
<td>His</td>
</tr>
<tr>
<td>CAC</td>
<td>His</td>
</tr>
<tr>
<td>CAA</td>
<td>Gln</td>
</tr>
<tr>
<td>CAG</td>
<td>Gln</td>
</tr>
<tr>
<td>AAU</td>
<td>Asn</td>
</tr>
<tr>
<td>AAC</td>
<td>Asn</td>
</tr>
<tr>
<td>AAA</td>
<td>Lys</td>
</tr>
<tr>
<td>AAG</td>
<td>Lys</td>
</tr>
<tr>
<td>AGU</td>
<td>Ser</td>
</tr>
<tr>
<td>AGC</td>
<td>Ser</td>
</tr>
<tr>
<td>AGA</td>
<td>Arg</td>
</tr>
<tr>
<td>AGG</td>
<td>Arg</td>
</tr>
<tr>
<td>GAU</td>
<td>Asp</td>
</tr>
<tr>
<td>GAC</td>
<td>Asp</td>
</tr>
<tr>
<td>GAA</td>
<td>Glu</td>
</tr>
<tr>
<td>GAG</td>
<td>Glu</td>
</tr>
<tr>
<td>GGU</td>
<td>Gly</td>
</tr>
<tr>
<td>GGC</td>
<td>Gly</td>
</tr>
<tr>
<td>GGA</td>
<td>Gly</td>
</tr>
<tr>
<td>GGG</td>
<td>Gly</td>
</tr>
</tbody>
</table>

Table 1.1: *The genetic code. *AUG is the methionine codon, but it is also the start codon.*

12
1.1.2 ncRNA examples

In the previous section many ncRNAs were mentioned, here is a short summary of some important ncRNAs.

**tRNA**
The transfer RNA (tRNA) is a well-studied RNA responsible for bringing the correct amino acid to the ribosome in translation. The tRNA anticodon reads off the mRNA codon thus bringing the amino acid that it is carrying to the ribosome for insertion to a growing peptide. This is one part of the identification of tRNA that is important for a correct translation, but the tRNA also needs to be recognized by the correct aminoacyl-tRNA synthetase, which charges the tRNA with a specific amino acid.

The tRNA has a clover leaf shaped secondary structure (see Figure 1.6). The 5’- and 3’-ends base-pair and form the acceptor stem. Opposite to the acceptor stem is the C-stem and the anticodon loop. The anticodon consists of three nucleotides binding complementary to a codon in the mRNA. Some tRNAs have a fifth stem in the variable region between the C- and T-stems. In particular tRNAs charged with Leucine (L), Serine(S) and Tyrosine (Y) have long variable regions with a fifth stem-loop structure.

**rRNA**
The ribosomal RNAs (rRNAs) constitute in complex with proteins the ribosome. The rRNAs have a catalytic function (hence they are ribozymes). The rRNAs are usually the most abundant RNAs in the cell.

**snRNA**
The small nuclear RNAs (snRNAs) are small RNA molecules located in the eukaryotic nucleus. The snRNAs are involved in various processes, such as RNA splicing and maintenance of the telomeres.

**snoRNA**
The small nucleolar RNAs (snoRNAs) guide various chemical modifications of other ncRNAs (such as rRNA and tRNA). The C/D box snoRNAs guide methylation and the H/ACA snoRNAs guide pseudouridylation.
siRNA
Small interfering RNA (siRNA) are double stranded RNAs (dsRNAs) of 20-25 nucleotides (nts) length involved in the process termed RNA interference (RNAi), the discovery of which was awarded the Nobel prize in Physiology and Medicine in 2006. The siRNA is included in the RNA-induced silencing complex (RISC) as single-stranded RNA and programs the RISC through complex complementarity to bind an mRNA, inducing cleavage of the mRNA at that site, which is then degraded.

The siRNAs are produced by the enzyme dicer either from exogenous or cellular double stranded RNA or from a hairpin structured RNA.

miRNA
The microRNAs (miRNAs) have similar functions as siRNAs, but are produced from a single stranded genomic ncRNA. The pre-miRNA is a ∼70 nts long hairpin structure that is processed by dicer to produce a usually 21 nts long mature miRNA. The mature miRNA can in complex with RISC inhibit the translation of many mRNAs. In contrast to siRNA, the miRNA usually does not bind perfectly to its target mRNA.

riboswitch
A riboswitch (Winkler et al., 2002; Penchovsky and Breaker, 2005) is a special part of an mRNA involved in gene regulation. A small molecule (such as such as ferritin, adenine, or lysine) binds to the riboswitch, usually inducing a conformational change in the riboswitch. The structural change affects gene expression, usually by repressing translation of the mRNA, but sometimes by derepressing translation.

1.2 RNA structure
The RNA molecule is, as described in section 1.1.1, a heteropolymer consisting of the nucleotides (bases) adenine (A), cytosine (C), guanine(G), and uracil (U) and possibly some modified bases. The bases have the ability to bind to each other, enabling an RNA to form base-pairs to another RNA or DNA molecule, but also to itself. Long stretches of base-paired nucleotides can form helical structures, similar to the DNA double helix. The base-pairings and interactions at higher level can give RNA diverse structures as proteins.

Like proteins, the structure of an RNA can be described in different levels (see Figure 1.9). The primary structure of an RNA molecule is the nucleotide sequence, the secondary structure of an RNA molecule is the set of base-pairings within the RNA, and the tertiary structure describes the three-dimensional structure, i.e. all the interactions and three-dimensional conformation that are not already described in the secondary structure.
The secondary structure of an RNA is often assumed to be sufficient for being able to predict the RNA function. This assumption can be justified by observations of well conserved secondary structures (Cech, 1988; Le and Zuker, 1990) and the fact that secondary structures fold fast, while tertiary interactions need much more time to form (Onoa and Tinoco, 2004). The fact that it is possible to predict secondary structures using nearest-neighbor parameters (Xia et al., 1998; Mathews et al., 1999; Mathews and Turner, 2002b) also suggests that secondary structure contributes much more to the stability of the RNA structure, than tertiary interactions. This is a great advantage in RNA structure prediction compared to protein structure prediction, where secondary structure information is far from sufficient to reasonably describe a protein structure.

In this section we will describe elements of RNA secondary structure and give some detail of how the energy can be computed of a particular secondary structure using the nearest-neighbor parameters.

1.2.1 Secondary structure

Although every base actually can base-pair with every other base in up to 12 different ways (Leontis and Westhof, 2001), only the six canonical base pairs are usually considered in structure predictions as they are by far the most common base-pairs. The canonical base-pairs are the Watson-Crick base-pairs G-C, C-G, A-U, and U-A and the wobble base pair G-U, and U-G (Varani and McClain, 2000). Of the canonical base-pairs, G-C is the strongest and G-U the weakest.

An example of a secondary structure is shown in Figure 1.7. The figure shows different combinations of paired and unpaired regions that can occur. Two consecutive base-pairs stack together to form a stacking loop and several consecutive base-pairs form a stem. The stem can be interrupted by a region of unpaired nucleotides, a so-called interior loop or, if unpaired nucleotides occur only on one strand, a bulge loop. The unpaired region at the end of a stem is called a hairpin loop. A multi-branch loop is a loop with several adjacent stems. In addition to stems and loops an RNA structure can, of course, also have unpaired nucleotides external to all the stems.

The energy of a given RNA secondary structure can be computed using the nearest-neighbor model (Tinoco, Jr et al., 1971) and parameters (Xia et al., 1998; Mathews et al., 1999; Mathews and Turner, 2002b). The nearest-neighbor model assumes that the energy of an RNA depend on neighboring base-pairs. The nearest-neighbor energy model is sometimes called a loop based model as it is based on energy contributions from loops (where two stacking base-pairs are also considered to form a loop). By adding energies for all loops in an RNA the total energy of the molecule is achieved. The loops are classified based on the number of base-pairs in the loop.
A 1-loop is a hairpin loop, it has one closing base-pair and a couple of unpaired nucleotides. The minimum number of unpaired bases in the loop is 3, since the backbone is not able to make a too sharp bend (there are of course exceptions to this rule, hairpin loops with only 2 unpaired bases have been observed). The energy of the 1-loop is dependent on the identity of the closing base-pair and the number of unpaired nucleotides in the loop. The larger the 1-loop is, the greater the energy, is the general principle, but there are some exceptions. Short hairpin loops have energies that vary a lot depending on the actual nucleotides in the loop and the most favorable hairpin loops are certain tetraloops.

There are three different types of 2-loops, i.e. loops with two closing base-pairs. The first type, and the only loop that actually give the molecule a negative energy contribution, is the stack. A stack consists of two consecutive base-pairs and no intervening unpaired bases. The energy of a stack depends on the identity of the base-pairs involved. The other types of 2-loops are the interior and bulge loop, both which increase the overall energy. The larger and more asymmetric the 2-loop is, the greater the energy.

The $k$-loops, $k > 2$, are the multiloops, i.e. loops with at least three closing base-pairs. The stability of the multiloop is still not well studied, but at least it is known to depend on the number of base pairs and unpaired bases in the loop, i.e the energy is computed by $a + b$(number of base-pairs) + $c$(number of bases), where $a$, $b$ and $c$ are constants.
In most secondary structure prediction programs based on the nearest-neighbor energy model, the energy of unpaired bases in the ends of helices (stacking of single bases on an adjacent base-pair) and coaxial stacking (stacking of two adjacent helices) are added as well (Mathews, 2004; Tyagi and Mathews, 2007).

For computational reasons, in general, only “nested” base-pairs are considered as part of the secondary structure. Non-nested structures, so called pseudoknots (see Figure 1.8), are thus considered as part of the tertiary structure rather than the secondary structure.

![Figure 1.8: Two example first order pseudoknot structures.](image)

**Primary structure**

5’-GCGCCGUGCUCAUUGGAGAGCGUUGACUACGGAUCAAAGGUUGGGGUUCGACUCUCUCUCGGCGCGC-3’

**Secondary structure**

![Secondary structure](image)

**Tertiary structure**

![Tertiary structure](image)

*Figure 1.9: The primary, secondary and tertiary structure of a tRNA.*

**Secondary structure representations**

The secondary structure of an RNA can be represented in several different ways. Here we will describe four common representations; see Figure 1.7 for examples.
Figure 1.10: Four ways to represent the secondary structure of a 74 nucleotide tRNA molecule. Shown in the figure are the squiggle plot generated by RNAplot (Hofacker et al., 1994) (top left), the mountain plot (top right), the circle plot (lower right) and the dot-bracket notation (at the bottom).

Several of the secondary structure representations are planar graphs, where the nodes represent the nucleotides and the edges represent interactions in the molecule, both the covalent bonds in the RNA backbone and the base-pairs. In the squiggle plot the planar graph is displayed so that interacting pairs are displayed close to each other (see Figures 1.7 and 1.10 for examples).

The circle plot is similar to the squiggle plot, but here the nucleotides are displayed at positions along a circle (representing the backbone), so that the nucleotides in the 5′ and 3′ ends will be adjacent in the circle, and the base-pairs are represented by connecting lines. When represented in circle plots, nested structures have the characteristic that their edge lines never cross. Structures where the edges do cross (if the edges are always drawn on the inside of the circle) are pseudoknot structures.

The mountain plot (Hogeweg and Hesper, 1984) is a graph, where the x-coordinate is the nucleotide position i in the RNA and the y-coordinate is the number of base-pairs enclosing nucleotide i.

The dot-bracket notation is very useful since it represents a secondary structure as a character string. An unpaired nucleotide is represented as a dot, “ . ”, and a base-pair is represented as a pair of opening and closing brackets, “ ( ” and “ ) ”.

Non-nested secondary structure, i.e. pseudoknots, are not generally representable by a planar graph. A pseudoknot that can be drawn without cross-
ing lines in the circle plot, if lines are allowed to be drawn both inside and outside the circle, are called first order pseudoknots. Second or higher order pseudoknots need more dimensions to avoid crossing lines. The dot-bracket notation can be extended to represent pseudoknots (at least low-order pseudoknots), by introducing squared and curly brackets to represent different levels of pseudoknots. For example the pseudoknot structure in Figure 1.8(a) can be represented by the string .(((((. . . [ [ [] ]] )))) . . .)].

1.2.2 RNA evolution

RNAs under so called ‘purifying selection’ evolve so as to conserve their function. Since secondary structure is usually important for RNA function, this means that base-pairings are conserved. Hence, two nucleotides in a base-pair do not evolve individually, rather they coevolve. Patterns of coevolving sites can be very useful to identify RNA genes and also for predicting secondary structure among a set of homologous RNA sequences (see section 3.2).

Compensatory mutations

Homologous RNA sequences from closely related species often differ by a pair of compensatory mutations, rather than by a single mutation (Higgs, 2000). Even though the two mutations appear to have occurred simultaneously, a compensatory mutation is almost always a two-step process. Compensatory mutations can be divided into two classes, those that have an intermediate with a wobble base-pair (GU or UG) (double transitions\(^2\)) and those with a mismatch intermediate (double transversion\(^3\)). The former is expected to occur at a higher rate since the selective disadvantage of a wobble base-pair should be less than that of a mismatch.

\(^2\)Transition: Mutation from a purine (A or G) to another purine, or from a pyrimidine (C or U (T)) to another pyrimidine.

\(^3\)Transversion: Change from purine to pyrimidine or vice versa.
2. Identification

Protein-coding genes are generally easier to identify than RNA genes. Many gene-finding programs have been developed both for scanning a genome for novel genes, but also for homology searching. Most of these programs are developed for finding protein-coding genes and can often not be applied at all, or with moderate accuracy, to the RNA gene-finding problem. This chapter will give a short overview of gene-finding approaches, mainly those than are designed for or at least can be applied to the RNA gene finding problem.

2.1 Protein gene finding

Protein coding genes have certain signals, e.g. in the promoter and transcription binding sites, which can be screened for, such as the Pribnow box in prokaryotic promoters. These signals are not enough for gene identification, but together with the information in the gene itself, regions in the genome can be classified as putative protein coding genes with high accuracy. Proteins are generally at least 100 amino acids long, which means that a protein coding gene requires an open reading frame (ORF) at least 100 codons long (300 nts), i.e. a stretch of DNA, starting with the start codon, which lacks an in-frame stop codon. There are 3 stop codons of in total 64 possible codons. In a random sequence where all codons are equally probable, the probability of finding a stretch of 100 codons without a stop codon is \( \left( \frac{61}{64} \right)^{100} \approx 0.0082 \).

Eukaryotic protein coding genes are more difficult to identify. Poly-A tails and CpG islands in eukaryotic promoters can be searched for, but in general the signals from promoters and other regulatory regions are more complex than those in prokaryotic genes. Another difficulty with the eukaryotic genes is that they generally contain introns (see section 1.1.1). However, the exon/intron signals can actually be used in the gene finding, which they are in e.g. GENESCAN (Burge and Karlin, 1997), which is an example of a program developed for protein gene finding in eukaryotes (in particular human).

Comparative methods can be used to detect protein coding genes since the sequence in a coding part of the genome tends to have much higher conservation than the rest of the genome. Also, so called synonymous mutations can be helpful signals in gene prediction. Synonymous mutations are mutations in protein coding sequences that do not change the resulting polypeptide. In
general, the third nucleotide in a codon is more redundant than the first two nucleotides and hence it will be less conserved (see table 1.1).

2.2 RNA gene finding

2.2.1 *Ab initio* identification

Once a genome is sequenced the second step in the analysis is usually gene finding, at least protein gene finding. Also the identification of certain ncRNA genes is important. For example, tRNAs are usually identified early in genome analysis. *Ab initio* gene finding is the computational problem of identifying stretches in a genomic sequence that code for a protein or a functional RNA.

RNA genes are much less studied than protein genes and there are no known general signals such as specific promoters for RNA genes that can be used for their identification. Also, since ncRNA genes are not translated, they do not contain comparable general structural signals like an ORF either. This makes the general RNA gene finding problem more difficult than protein gene finding and many approaches have instead focused on single classes of RNAs for more specific gene finding. For example, tRNAscan-SE (Lowe and Eddy, 1997) and ARAGORN (Laslett and Canback, 2004) scans for tRNAs, BRUCE (Laslett *et al.*, 2002) scans for tmRNAs (transfer-messenger RNAs), Fisher (Edvardsson *et al.*, 2003) and snoGPS (Schattner *et al.*, 2004) search for H/ACA snoRNAs, snoScan (Lowe and Eddy, 1999) searches for C/D box snoRNAs, snoReport (Hertel *et al.*, 2007) searches for both types of snoRNAs, and there are several approaches for miRNA identification, e.g. promir (Nam *et al.*, 2005), miralign (Wang *et al.*, 2005), mirscan (Lim *et al.*, 2003), and mirseeker (Lai *et al.*, 2003).

Today, when the amount of sequenced genomes is large, it is usually possible to find evolutionarily related sequences in other organisms. When a set of potential homologous sequences have been obtained (through the UCSC genome browser (Karolchik *et al.*, 2003) or Ensembl (Hubbard *et al.*, 2007) or similar resources), the problem reduces to a problem that can be solved using comparative genomics approaches, see section 2.2.2. Irmtraud Meyer wrote a recent review on RNA gene finding (Meyer, 2007) that describes various approaches for RNA gene finding, especially the comparative approaches, since these are the most common and most successful approaches.

Another approach to identify RNA genes is to search for signals that distinguish functional RNAs from random/coding sequences, e.g. the MFE of a sequence (Seffens and Digby, 1999; Rivas and Eddy, 2000; Bonnet *et al.*, 2004) (paper I). See section 4.1 and paper I for a description of various measures that can be used for identification of a structural RNA in a single genome sequence.
2.2.2 Comparative approach

If several orthologous sequences are known a comparative approach can be adopted to identify an RNA gene with conserved sequence and structure features. Conserved features are interesting since if they are conserved, they will tend to have a functional importance. Homologous RNA genes in general conserve not only their sequence but also their structure, or sometimes actually the structure rather than the sequence is conserved.

Programs developed for classification of a given alignment of potential RNA genes as functional RNA or not, can be applied also for the RNA gene finding problem. QRNA (Rivas and Eddy, 2001), RNAz (Washietl et al., 2005) and EvoFold (Pedersen et al., 2006) can all be used in a scanning mode, where an alignment of sequences is scanned for regions of potential ncRNAs, i.e. regions that seem to have conserved structure.

QRNA (Rivas and Eddy, 2001) classify a pairwise alignment as protein coding, functional RNA or else. The idea is to identify synonymous mutations in protein coding regions, or compensatory mutations consistent with some secondary structure in ncRNAs.

RNAz (Washietl et al., 2005) combine a measure of structure conservation with a measure of thermodynamic stability (see section 4.1.1) to classify a multiple alignment of RNA sequences as functional ncRNAs or not. RNAz is much faster and more reliable than QRNA.

EvoFold (Pedersen et al., 2006) uses a phylogenetic stochastic context free grammar (phylo-SCFG) for identification of ncRNAs in multiple alignments. An advantage of EvoFold is that phylogenetic information can be included.

2.3 Homology search

To find homologous sequences to a known gene in a genomic sequence is a common problem and for protein coding genes it is quite straightforward. Protein coding genes are generally well-conserved, although some variations can be allowed, especially in the third codon position, without a change in the resulting protein product. There are several programs available for this problem, a commonly used one is BLAST (Altschul et al., 1990).

For RNA genes the homology search problem is more difficult, because of the fact that intra- and intermolecular base-pairs are, in evolutionary terms, preserved to a higher degree than the sequence. The wobble GU and other non-canonical base-pairs allow RNA sequences to evolve seemingly unrelated sequences along nearly neutral paths through structure space (e.g., AU ↔ GU ↔ GC). Thus, specialized homology search techniques, such as nucleotide specific scoring schemes (States et al., 1991), profile hidden Markov model (profile HMMs) (Krogh et al., 1994), and covariance models (CMs) (Eddy and Durbin, 1994), are necessary for accurate ncRNA homology search. Paper II
presents a benchmark study evaluation the performance of homology search tools on ncRNA.

2.3.1 Homology search methods

Homology search methods can be divided into three classes; sequence based methods, profile HMM methods, and structure based methods. Here, we will give a short overview of these three types of search methods.

**Sequence based methods**

The most straight forward approach for finding homologous RNA genes is the local matching approach, i.e. a pure sequence based approach. The local matching problem was solved in the early 1980’s by Smith and Waterman (Smith and Waterman, 1981) with a dynamic programming algorithm. The Smith-Waterman algorithm is implemented in the program SSEARCH (Pearson, 1991). However, this method is slow, and generally heuristic methods, such as BLAST (Altschul et al., 1990) and FASTA (Pearson and Lipman, 1988), are used instead.

**Profile HMM methods**

When several orthologous sequences (or at least two) are known (and alignable) profile HMM methods (Krogh et al., 1994) can be a good alternative to the sequence based methods, as these can only consider one of the orthologous sequences at a time. The profile HMMs create a probabilistic model from an input alignment, which is then used to search the genome. Several studies have shown that, applied on protein data, profile HMMs are more accurate than sequence based methods (Brenner et al., 1998; Park et al., 1998; Lindahl and Elofsson, 2000; Madera and Gough, 2002). In paper II, we show that the same holds for RNA data.

**Structure based methods**

A functional RNA usually has an important secondary structure. If this structure is known for the query RNA, it can be taken into account in a homology search. This does of course have both advantages and disadvantages. The main advantage is a more accurate search result, as the sequence is often less important than the secondary structure for functional RNAs. However, if the secondary structure is not known perfectly it might be partially wrong leading to incorrect hits. However, the most common reason for not including structural information in the search is the time complexity. Structure based search methods are very time consuming.

RSEARCH (Klein and Eddy, 2003) is a stochastic context free grammar (SCFG) based method that takes a single sequence with a corresponding secondary structure as input. RSEARCH searches a genome for homologous RNAs, reporting BLAST-like output statistics. RSEARCH scores the
output alignments using RIBOSUM matrices, i.e. base-pair and nucleotide substitution matrices developed for this specific purpose.

If the structure and a set of orthologous sequences are known a SCFG, or covariance model (CM) can be created using a program such as Infernal (Eddy, 2002). The CM can then be used to scan a genome for homologs. Even if the structure is not known, it might still be possible to use Infernal for searching for homologs, since a secondary structure can usually be predicted from a large enough alignment, see section 3.2 and (Gardner and Giegerich, 2004).

The main disadvantage of the CM-based methods is the time complexity. Programs such as RaveNna (Weinberg and Ruzzo, 2006) solves this problem by converting a CM to a profile HMM, which can be used to rapidly scan the database.
3. Prediction

The three-dimensional structure of both proteins and functional ncRNAs is fundamental for the function of the molecules. Both in RNA and protein folding, the secondary structure usually forms first and once the base-pairings in the RNA or the \( \alpha \)-helices and \( \beta \)-sheets in the protein have formed the tertiary interactions form. When it comes to structure prediction, RNA has a great advantage over protein, both since the secondary structure contributes to a very large extent to the stability of the tertiary structure (see section 1.2), but also because the secondary structure can be predicted with reasonable accuracy using computationally feasible algorithms. In this chapter, we will describe algorithms for RNA secondary structure prediction based on dynamic programming.

RNA secondary structure prediction is the inference of a secondary structure based on the nucleotide sequence of one or several homologous RNAs. These algorithms have their limitations, e.g. they will not take into account base modifications or interactions with other molecules (such as chaperones). Furthermore, pseudoknots are generally not predicted and transcription is generally assumed to be completed before the RNA starts to fold.

3.1 Secondary structure prediction algorithms

The secondary structure of an RNA describes the set of base-pairs in the molecule. In secondary structure prediction algorithms the definition of a base-pair is usually more restricted than it is \textit{in vivo}.

Let \( s = s_1 s_2 \ldots s_n \) denote an RNA sequence of length \( n \), where the nucleotides \( s_i \in \{A, C, G, U\} \). Here we have introduced the first assumption, that \( s \) consists of one of the four unmodified nucleotides adenine (A), cytosine (C), guanine (G), and uracil (U). Let \((i, j)\) denote a base-pair between nucleotides \( s_i \) and \( s_j, i < j \), then an RNA secondary structure, \( \mathcal{S} \), is a set of base-pairs that fulfill the following three constraints:

1. if \((i, j) \in \mathcal{S}\) then \(1 < i \leq i + \theta < j \leq n\), where \( \theta \) is a non-negative integer (usually \( \theta = 3 \)),
2. if \((i, j) \in \mathcal{S}\) and \((k, l) \in \mathcal{S}\) then \(i = k \iff j = l\), and
3. if \((i, j) \in \mathcal{S}\) and \((k, l) \in \mathcal{S}\) then \(i < k < j \iff i < l < j\).

In words this means that (1) there must be at least \( \theta \) unpaired bases in a hairpin loop, (2) there are no base triples, i.e. a nucleotide can base-pair to at most one nucleotide, and (3) the structure must be nested, i.e. pseudoknots are not
allowed. A secondary structure $S$ is said to be compatible with $s$ if, for every base-pair $(i, j) \in S$, $s_i s_j \in B$, where $B$ denotes the possible base-pairings. Here $B = \{AU, UA, GC, CG, GU, UG\}$, i.e. the canonical base-pairs (the Watson-Crick base-pairs and the GU wobble).

### 3.1.1 Maximization of number of base-pairs

An RNA secondary structure is built up by a set of stabilizing base-pairs. As a first approximation of RNA structure stability, we can count the number of base-pairs. It is not true in general that more base-pairs give a more stable structure, but it is a simple and reasonable first approximation. An algorithm for computing the maximum number of base-pairs is described in (Nussinov et al., 1978) and is usually referred to as the Nussinov algorithm (it is sometimes referred to as the loop matching (LM) algorithm). The algorithm is implemented using dynamic programming, but can be defined as a recursive algorithm that computes $M_{ij}$, the maximum number of base-pairs of a subsequence $s_i \ldots s_j$. The algorithm is initialized by letting

$$M_{i,j} = 0, \forall j \leq i + \theta,$$

since a base-pair cannot be formed in a sequence of length $\theta + 1$ or shorter. The maximum number of base-pairs in a sequence longer than $\theta + 1$ is computed by the following recursion;

$$M_{i,j} = \max \begin{cases} M_{i,j-1}, \\ M_{i+1,j-1} + 1, \text{ if } s_is_j \in B, \\ \max_{i < k < j - \theta} [M_{i,k-1} + M_{k+1,j-1} + 1]. \end{cases}$$

In words, the maximum number of base-pairs in the subsequence $s_i \ldots s_j$, $M_{i,j}$, is the maximum taken over the following three cases; (i) either $s_j$ is unpaired, then the number of base-pairs is $M_{i,j-1}$, or (ii) $s_j$ is base-paired to $s_i$ (if $s_is_j \in B$), then $M_{i,j} = M_{i+1,j-1} + 1$, i.e. the number of base-pairs inside $(i, j)$ and the base-pair $(i, j)$ itself, or (iii) $s_j$ is base-paired to another nucleotide $s_k$, $i < k < j$ (if $s ks_j \in B$), then the number of base-pairs in $s_i \ldots s_j$ is the sum of $M_{i,k-1}$ (the base-pairs outside $(k, j)$), $M_{k+1,j-1}$ (the base-pairs inside $(k, j)$) and $(k, j)$.

In the dynamic programming algorithm, the values $M_{i,j}$ will be filled in in a matrix $M$, starting with the shortest subsequences first so that the values of $M_{i,j-1}$, $M_{i,k-1}$, and $M_{k+1,j-1}$ are always known when computing $M_{i,j}$. The below pseudocode shows the dynamic programming algorithm for filling in the matrix $M$. 

28
for $d = 0..\theta$
  for $i = 1..n-d$
    $j = i + d$
    $M_{i,j} = 0$
  for $d = \theta+1..n-1$
  for $i = 1..n-d$
    $j = i + d$
    $M_{i,j} = M_{i,j-1}$
    if $s_i s_j \in \mathbb{B}$
      $M_{i,j} = \max\{M_{i,j}, M_{i+1,j-1+1}\}$
      for $k = i+1..j-\theta-1$
        if $s_k s_j \in \mathbb{B}$
          $M_{i,j} = \max\{M_{i,j}, M_{i,k-1} + M_{k+1,j-1} + 1\}$

The Nussinov algorithm runs in $O(n^3)$ time and requires $O(n^2)$ space. The maximum number of base-pairs for the entire sequence is $M_{1,n}$.

**Backtracking**

The above algorithm determines the maximum number of base-pairs for a given RNA sequence. Once the matrix $M$ is computed (using the above algorithm), a secondary structure with the maximum number of base-pairs can be computed in a backtracking procedure. The below function $\text{maxstr}(i, j)$ can be used to find a secondary structure (a set of base-pairs) with the maximum number of base-pairs (there can be more than one structure with the maximum number of base-pairs) in the subsequence $s_i...s_j$. The function $\text{maxstr}(i, j)$ returns a set of base-pairs. In the below pseudocode $\{\}$ denotes the empty set, $\{(i, j)\}$ denotes a set with the single base-pair (i,j), and a sum between sets means to take the union.

```plaintext
function maxstr(i, j):
  if $j - i \leq \theta$
    return {};
  else if $M_{i,j} == M_{i,j-1}$
    return maxstr(i, j-i)
  else
    if $(M_{i+1,j-1+1}) == M_{i,j}$ and $s_i s_j \in \mathbb{B}$
      return maxstr(i+1, j-1) + \{(i, j)\}
    for $k = i+1..j-\theta$
      if $(M_{i,k-1} + M_{k+1,j-1+1}) == M_{i,j}$ and $s_k s_j \in \mathbb{B}$
        return maxstr(i,k-1) + maxstr(k+1,j-1) + \{(k, j)\}
```

The optimal secondary structure for the entire sequence $s$ is computed by $\text{maxstr}(1, n)$.

To reduce the runtime for the backtracking procedure a second matrix, the backtracking matrix $T$, can be introduced. The matrix $T$ is of the same size as $M$ and is filled in simultaneously with $M$. $T$ is defined such that the element
$T_{i,j}$ gives the index $k$ to the nucleotide in the subsequence $s_i\ldots s_j$ that $s_j$ should base-pair to for maximal number of base-pairs in the subsequence, or 0 if $s_j$ is unpaired. $T_{i,j}$ is assigned its value in the dynamic programming algorithm when $M_{i,j}$ is computed. An optimal secondary structure for a subsequence $s_i\ldots s_j$ can then be computed using the function $\text{maxstrT}(i,j)$ defined below:

\begin{verbatim}
function maxstrT(i,j):
  if $j-i \leq \theta$
    return {}
  else if $T_{i,j} == 0$
    return maxstrT(i,j-1)
  else if $T_{i,j} == i$
    return maxstrT(i+1,j-1) + {(i,j)}
  else
    $k = T_{i,j}$
    return maxstrT(i,k-1) + maxstrT(k+1,j-1) + {(k,j)}
\end{verbatim}

An optimal structure for the entire sequence is computed by $\text{maxstrT}(1,n)$.

### 3.1.2 Minimal free energy (MFE) minimization

The algorithm for computing maximum number of base-pairs in a sequence is simple, but it has several problems. Experiments have shown that the stability contribution from a single base-pair is dependent on the base-pair’s neighborhood. Two adjacent base-pairs form a base-pair stack that stabilizes the RNA secondary structure, but the secondary structure is destabilized by unpaired bases in hairpin loops, bulges, interior loops and multi branch loops.

Nearest-neighbor energy models (Tinoco, Jr et al., 1971) or loop-based energy models compute the free energy of an RNA secondary structure by adding up energy contributions from loops, i.e. interior, bulge, hairpin, stack and multi-branch loops. See section 1.2.1 for details.

In 1980, Nussinov and Jacobson improved the Nussinov algorithm for computing maximum number of base-pairs (Nussinov et al., 1978) by including simple nearest-neighbor energies in their model (Nussinov and Jacobson, 1980). The recursion in equation 3.2 was replaced by

\begin{equation}
E_{i,j} = \min \left\{ \begin{array}{l}
  E_{i,j-1}, \\
  E_{i+1,j-1} + e_{i,j}, \text{if } s_i s_j \in \mathbb{B}, \\
  \min_{i < k < j - \theta, \ s_k s_j \in \mathbb{B}} [E_{i,k-1} + E_{k+1,j-1} + e_{k,j}],
\end{array} \right. \tag{3.3}
\end{equation}

where $E_{i,j}$ is the minimum free energy (MFE) of the subsequence $s_i\ldots s_j$ and $e_{k,j}$ is the energy contribution from the base pair $(k,j)$. The energies $e_{k,j}$ de-
pend on whether the base-pair \((k, j)\) is adjacent to another base-pair, a bulge, an interior loop or a branched structure and is determined by backtracking.

In 1981, Zuker and Stiegler (Zuker and Stiegler, 1981) presented a refined dynamic programming algorithm that computes the minimum free energy structure based on energy contributions from loops. The loop energy contributions are determined by experiments. The programs mfold (Zuker et al., 1999) and RNAfold (Hofacker et al., 1994) both implement this algorithm using the energy parameters described in (Mathews et al., 1999).

A \(k\)-loop consists of \(k - 1\) base-pairs (excluding the closing one) and \(u\) unpaired bases. Energies for 1-loops (hairpins) and 2-loops (stack, interior and bulge loops) are experimentally determined (Mathews et al., 1999) and depend on \(k\) and \(u\) as well as the RNA sequence. Energies for multi-branch loops \((k > 2)\) are computed according to the linear model

\[
e_{\text{multi}} = a + b(k - 1) + cu,\]

where \(a\), \(b\), and \(c\) are constants. See section 1.2.1 for more details.

The Zuker-Stiegler algorithm needs three types of matrices to compute the MFE for a sequence; \(E\), \(EB\), and \(EM\). \(E\) is the main matrix and the element \(E_{i,j}\) gives the MFE of the subsequence \(s_i \ldots s_j\). \(EB\) is a restriction, where \(EB_{i,j}\) gives the MFE of the same subsequence if \(s_i\) and \(s_j\) form a base-pair in the subsequence. If \(s_i\) and \(s_j\) cannot pair, \(EB_{i,j}\) will be \(\infty\). \(EM\) gives the MFE of subsequences in multi loops (a region in a multi loop should have at least one base-pair otherwise \(EM_{i,j}\) is \(\infty\)).

The three matrices are filled in starting with subsequences of length shorter than or equal to \(\theta + 1\);

\[
E_{i,j} = 0, \quad EB_{i,j} = EM_{i,j} = \infty, \quad \forall j - i \leq \theta
\]  

(3.4)
since no base-pair can be formed between nucleotides that are less than \(\theta\) base-pairs apart. For subsequences longer than \(\theta + 1\) the matrices are filled in, for shorter subsequences first, according to equations 3.5-3.7.

For a subsequence \(s_i \ldots s_j\) the minimum free energy \(E_{i,j}\) can be computed by considering three possibilities, either \(s_j\) is unpaired in \(s_i \ldots s_j\), \(s_j\) is paired with \(s_i\) or \(s_j\) is paired with \(s_k\), \(i < k < j - \theta\) (compare with equation 3.2). Hence, \(E_{i,j}\) is computed according to the following;

\[
E_{i,j} = \min \left\{ \begin{array}{ll}
E_{i,j-1}, & \\
EB_{i,j} + e_{\text{dangle}}, & \text{if } s_is_j \in \mathbb{B}, \\
\min_{i < k < j, \ s_is_j \in \mathbb{B}} \left( E_{i,k-1} + EB_{k,j} + e_{\text{dangle}} \right), & 
\end{array} \right.
\]

(3.5)

where \(e_{\text{dangle}}\) is the energy contribution due to dangling ends (single bases stacking on adjacent base-pairs) and non GC base-pairs closing a stem (non GC base-pairs closing a stem have a destabilizing effect).

Note that \(E_{i,j}\) depend on \(E\) and \(EB\) for subsequences that are shorter or of the same length as \(s_i \ldots s_j\), hence we need to compute the energy for shorter
subsequences first, and we need an equation for computing \( E_{Bi,j} \). \( E_{Bi,j} \) is computed by taking the minimum of three possible cases, either \((i, j)\) is the only base-pair in \( s_i \ldots s_j \) (closing a hairpin loop), or there is another base-pair \((k, l)\) interior to \((i, j)\), resulting in an interior loop with the two closing base-pairs \((k, l)\) and \((i, j)\), or \((i, j)\) is closing a multi-branch loop. Hence, \( E_{Bi,j} \) can be computed according to the following:

\[
E_{Bi,j} = \min \left\{ \begin{array}{ll}
e_{\text{hairpin}(i,j)}, \\
\min_{i<k<l<j} \left( e_{\text{interior}(i,j,k,l)} + E_{B_{k,l}} \right), \\
\min_{i<k<l<j} \left( a + b + c(j-l-1) + EM_{i+1,k-1} + E_{B_{k,l}} \right),
\end{array} \right.
\]

(3.6)

where \( e_{\text{hairpin}(i,j)} \) is the energy contribution from the hairpin loop closed by \((i, j)\), \( e_{\text{interior}(i,j,k,l)} \) is the energy of a stack, interior or bulge loop closed by \((i, j)\) and \((k, l)\), \( a \), \( b \) and \( c(j-l-1) \) are energy contributions for the multi loop \((a \) for it being a multi loop, \( b \) for the \((k, j)\) base-pair, and \( c(j-l-1) \) for the unpaired bases between \( l \) and \( j \)).

Finally the multi loop energy matrix, \( EM \), is filled out in the following manner:

\[
EM_{i,j} = \min \left\{ \begin{array}{ll}
c + EM_{i,j-1}, \\
\min_{i<k<j} \left( b + c(k-i) + E_{B_{k,j}} \right), \\
\min_{i<k<j} \left( b + EM_{i,k-1} + E_{B_{k,j}} \right),
\end{array} \right.
\]

(3.7)

based on the three cases where either \( j \) is unpaired, \((k, j)\) form the only base-pair in the multi loop between \( i \) and \( j \), or \((k, j)\) is a base-pair in the multi loop, but there is at least one more in the region between \( i \) and \( k - 1 \).

The matrices are filled in starting with the shortest subsequences. Since \( E_{Bi,j} \) might be needed both to compute \( EM_{i,j} \) and \( E_{i,j} \), \( EB \) values are computed first, then the \( EM \) and \( E \) values. The algorithm requires \( O(n^2) \) space and runs in \( O(n^4) \) time. The run time can be reduced to \( O(n^3) \) by limiting the maximum size of an interior and bulge loop to a fixed upper bound of number of unpaired nucleotides (for example 30) and introducing an additional matrix \( EM1 \) that stores the energy of subsequences that are in multiloops, but contains exactly one closing base-pair.

It is possible to also include coaxial stacking energies (stacking energies of adjacent helices) into account, but this requires more complicated recursive formulas than those presented here and an additional storage matrix (Mathews et al., 2004).

**Backtracking**

The optimal secondary structure according to the nearest-neighbor energy model can be obtained in a backtracking procedure similar to the one de-
scribed for the Nussinov model, but here three backtracking matrices will be needed.

3.1.3 Suboptimal folding

An RNA sequence can generally fold in several very different secondary structures that are close to the minimal free energy, this makes it many times of interest to predict several suboptimal foldings and not just a single minimal free energy structure. Suboptimal structures can be computed through suboptimal backtracking. This means that during the backtracking procedure slightly suboptimal cases may be chosen. The difficulty in these computations is to limit the suboptimality allowed, an algorithm for limiting the suboptimality level is presented in (Williams, Jr and Tinoco, 1986). A method for computing all possible foldings within a given energy range from the MFE is presented in (Wuchty et al., 1999) and implemented in the program RNAsubopt included in the Vienna RNA package (Hofacker et al., 1994).

An alternative to predicting suboptimal foldings can sometimes be constrained folding. The secondary structure is not always completely unknown, perhaps the RNA is known to interact with another molecule in a specific region of the RNA, some bases are known to be unpaired because they have undergone specific chemical modifications, or maybe the secondary structure is known for a homologous sequence. In this case the secondary structure can be predicted, while certain bases are forced to be unpaired or base-paired by using large energy penalties in the minimum free energy calculations (Zuker and Stiegler, 1981).

3.1.4 Partition function

Generally, a given sequence, s, can fold into a large number of alternative folds, the MFE structure is only one among many possible folds. We call the set of all possible folds of s the ensemble, S, of structures compatible with s. The partition function of the ensemble of structures is defined as;

\[ Z = \sum_{T \in S} e^{-E_T/RT}, \]

(3.8)

where the sum is over all structures T in the ensemble, \( E_T \) are the corresponding free energies for the structures in kcal/mol, R is the molar gas constant and T is the temperature in Kelvin. The factor \( e^{-E_T/RT} \) is the so called Boltzmann factor for the particular structure T. The Boltzmann factor is proportional to the probability of a structure and once the partition function \( Z \) is computed the probability of a single structure can be computed;

\[ p_T = \frac{e^{-E_T/RT}}{Z} \]

(3.9)
However, especially for larger structures it is rarely a single structure that is biologically interesting, but rather a subensemble of similar, rapidly interchanging structures. The structures in the subensemble probably have a core structure in common, but a few base-pairs of less importance can either be paired or not. The interesting probability would then be the accumulated probability for all structures in the subensemble.

The partition function calculation (McCaskill, 1990) is very similar to the minimum free energy computation, except that minimizations are replaced by additions and additions are replaced by multiplications. The dynamic programming algorithm for computing the partition function requires three matrices $Z$, $ZB$, and $ZM$. The formula for computing the partition function contribution from the subsequence $s_i \ldots s_j$, $Z_{i,j}$, is given in equation 3.10. As for the MFE computation two additional, restricted matrices are needed, $Z_{Bi,j}$ computes the partition function for $s_i \ldots s_j$ when $(i, j)$ is base-paired (see equation 3.12) and $Z_{Mi,j}$ gives the partition function contribution from a subsequence $s_i \ldots s_j$ inside a multi loop (see equation 3.13).

$Z_{i,j} = Z_{i,j-1} + \Delta(s_is_j)Z_{Bi,j}e^{-e_{\text{dangle}}/RT} + \sum_{i<k<j \atop s_is_j \in \mathbb{B}} Z_{i,k-1}Z_{Bk,j}e^{-e_{\text{dangle}}/RT}, \tag{3.10}$

where

$\Delta(s_is_j) = \begin{cases} 1 & \text{if } s_is_j \in \mathbb{B} \\ 0 & \text{otherwise} \end{cases}. \tag{3.11}$

$Z_{Bi,j} = e^{-e_{\text{hairpin}}(i,j)/RT} + \sum_{i<k<l<j \atop s_is_j \in \mathbb{B}} e^{-e_{\text{interior}}(i,j,k,l)/RT}Z_{Bk,l} + \sum_{i<k<l<j \atop s_is_j \in \mathbb{B}} e^{-(a+b+c(j-l-1))/RT}Z_{M_{i+1,k-1}}Z_{Bk,l} \tag{3.12}$

$Z_{Mi,j} = e^{-c/RT}Z_{Mi,j-1} + \sum_{i<k<j \atop s_is_j \in \mathbb{B}} e^{-(b+c(k-i))/RT}Z_{Bk,j} + \sum_{i<k<j \atop s_is_j \in \mathbb{B}} e^{-b/RT}Z_{Mi,k-1}Z_{Bk,j} \tag{3.13}$

As for the MFE calculation the matrices are filled in, in a dynamic programming manner, starting with the shorter subsequences. The partition function matrices are initialized by

$Z_{i,j} = 1$, $Z_{Bi,j} = Z_{Mi,j} = 0$, $\forall j - i \leq \theta \tag{3.14}$
Once the matrices are filled in the total partition function for the sequence is $Z = Z_{1,n}$, but also the other entries in the matrices can be of interest. Based on the the partition function matrices, the probability of individual base-pairs can be computed (McCaskill, 1990).

RNAfold has an option (-p) for computing the partition function and hence also the base-pair probabilities. A base-pair probability matrix is usually displayed in a *dot plot*, where the base-pair probabilities are shown in a matrix as dots of sizes proportional to their probabilities. A dot plot can be used to visualize the uniqueness of an RNA fold as alternative folds will be shown with dots of varying sizes depending on their probabilities. For an example dot plot see Figure 3.1. In chapter 4.1.2 and paper I we describe measures that can be used to summarize the dot plot in a single number as a measure of uniqueness of fold.

**Figure 3.1:** An example dot plot. The upper triangular part of the dot plot shows dots that are proportional to the corresponding base-pair probability in size. The lower triangular part shows dots representing the base-pairs predicted in the MFE structure. The structures to the right of the dot plot are the MFE structure (upper) and the structure with the most probable base-pairs (lower). The structures were predicted and the figures created using RNAfold -p (Hofacker et al., 1994).

The energy matrices in the Zuker-Stiegler algorithm can be backtracked to get a minimal free energy structure. The partition function matrices $Z$, $Z_B$ and $Z_M$ can in a similar manner be used for statistical sampling (Ding and Lawrence, 2003). Sfold (Ding and Lawrence, 2003; Ding et al., 2005) implements an algorithm for statistical sampling of RNA secondary structures from the Boltzmann ensemble.
3.1.5 Pseudoknots

Both the Nussinov and the Zuker-Stiegler algorithm exclude pseudoknots. There have been many attempts to write an algorithm that predicts secondary structure including some restricted types of pseudoknots. It has been proven that the general problem of predicting RNA secondary structures with pseudoknots using an energy model is NP complete (Lyngsø and Pedersen, 2000).

The most general pseudoknot predicting algorithm (includes most types of relevant pseudoknots) is implemented in pknots (Rivas and Eddy, 1999). However, this algorithm is expensive both in time and space (the complexity is \(O(n^6)\) in time and \(O(n^4)\) in space, where \(n\) is the sequence length). In addition to the complexity, there is a problem with the thermodynamic parameters; there is not much thermodynamic information available for pseudoknots.

Some algorithms for pseudoknot prediction that are based on Nussinov like algorithms, such as ilm (Ruan et al., 2004) and our program MIfold (paper V), include comparative information (see section 3.2) to increase their accuracy.

3.2 Comparative methods

The methods described in section 3.1 predict RNA secondary structure from a single sequence. In many cases we actually have more than one sequence available, which are all believed to be homologous and fold into the same secondary structure. In this case a structure in common to all the RNA sequences is probably closer to the native (functional) structure than the structures predicted from the individual sequences.

There are several approaches to the problem of consensus structure prediction (for a comparison of RNA folding methods see (Gardner and Giegerich, 2004)), the three main approaches are:

1. Align the sequences, then fold the alignment.
2. Fold and align the sequences simultaneously.
3. Fold the sequences, then align the secondary structures.

3.2.1 Alignment folding

The first approach requires that the sequences have sufficient sequence similarity so that they can be aligned reliably using an alignment tool such as ClustalW (Thompson et al., 1994) or t-coffee (Notredame et al., 2000) (see (Gardner et al., 2005) for a benchmark study on multiple alignment tools applied to RNA sequences). Once the sequences are aligned, sites with structurally neutral mutations are identified and used to infer a consensus structure. Structurally neutral mutations can be identified by measures of covariation, such as mutual information (equation 3.15). The mutual information of two independent random variables \(X\) and \(Z\) over the the values \(x \in \mathcal{X}\) and \(z \in \mathcal{Z}\),
respectively, is generally defined as

\[
I^\text{mut}(X, Z) = \sum_{x \in \mathcal{X}} \sum_{z \in \mathcal{Z}} p(x, z) \log_2 \frac{p(x, z)}{p(x)p(z)},
\]

where \( p(x) = P(X = x) \) and \( p(z) = P(Z = z) \) are the probability density distributions of \( X \) and \( Z \), respectively, and \( p(x, z) = P(X = x, Z = z) \) is the joint probability density distribution of \( X \) and \( Z \).

Here, we will compute the mutual information between two columns in an alignment. Let \( X_i \) and \( Z_j \) denote two random variables over the values \( x \in \mathcal{X} \) and \( z \in \mathcal{X} \), respectively. \( X_i \) and \( Z_j \) represent alignment columns \( i \) and \( j \), respectively and hence \( \mathcal{X} \) represent the alignment alphabet, e.g. for an RNA alignment \( \mathcal{X} = \{ \text{A, C, G, U} \} \) (if the gap state is included. Since the probability density distributions of \( X_i \) and \( Z_j \) are not known, we will instead use observed frequencies to compute the mutual information between columns \( i \) and \( j \). Hence,

\[
I^\text{mut}_{i,j} = I^\text{mut}(X_i, Z_j) = \sum_{x \in \mathcal{X}} \sum_{z \in \mathcal{X}} f_{i,j}(xz) \log_2 \frac{f_{i,j}(xz)}{f_i(x)f_j(z)},
\]

where \( f_i(x) (f_j(z)) \) is the frequency of \( x (z) \) in position \( i (j) \) \((0 \leq f_i(x), f_j(z) \leq 1, \sum_{x \in \mathcal{X}} f_i(x) = 1, \sum_{z \in \mathcal{X}} f_j(z) = 1) \) and \( f_{i,j}(xz) \) is the joint frequency of \( x \) in position \( i \) and \( z \) in position \( j \) \((0 \leq f_{i,j}(xz) \leq 1, \sum_{x \in \mathcal{X}} \sum_{z \in \mathcal{Z}} f_{i,j}(xz) = 1) \). In paper V we adopt a restricted version of equation 3.16, where the double sum is taken only over \( xz \in \mathcal{B} \), i.e. only over the canonical base-pairs. This is to reduce the noise in the structure prediction where we are only interested in covariations due to base-pairings.

\textsc{RNAalifold} is a program that combines free energy and a covariation measure, see equation 3.17, in a score that is minimized in a dynamic programming algorithm, similar to the Zuker-Stiegler algorithm (Zuker and Stiegler, 1981), to predict a consensus structure. The covariation measure used in \textsc{RNAalifold} is, for alignment columns \( i \) and \( j \), computed as

\[
C_{i,j} = C(X_i, Z_j) = \sum_{xz \in \mathcal{B}} \sum_{x'z' \in \mathcal{B}} f_{i,j}(xz)D(xz, x'z')f_{i,j}(x'z'),
\]

where \( D(a, b) \) is defined as the Hamming distance between two sequences \( a = a_1a_2\ldots a_n \) and \( b = b_1b_2\ldots b_n \), i.e.

\[
D(a, b) = \sum_{i=1}^{n} d(a_i, b_i), \text{ where } d(a_i, b_i) = \begin{cases} 
0 & \text{if } a_i = b_i \\
1 & \text{if } a_i \neq b_i.
\end{cases}
\]

Also \textsc{pfold} (Knudsen and Hein, 1999, 2003) and \textsc{ilm} (Ruan et al., 2004) predict a consensus structure from an alignment. The program \textsc{pfold} is based both on SCFG and an evolutionary model. The program \textsc{ilm} implements
an interactive dynamic programming algorithm that combines a thermodynamic score and mutual information (the thermodynamic score is a “helix plot score” as described in (Tabaska et al., 1998) and not based on nearest-neighbor energy parameters (Mathews et al., 1999)). ilm can predict pseudo-knotted structures.

In paper V we describe the program MIfold that is mainly useful for visualizing RNA alignments, but that can also predict secondary structure (including some pseudoknots) from a covariation score, see section 4.2.5 for more details.

3.2.2 Simultaneous alignment and folding

When the sequence similarity between sequences is not high enough to make reliable alignments the first approach (section 3.2.1) cannot be applied. The sequences can instead be simultaneously aligned and folded using the Sankoff algorithm (Sankoff, 1985) or a restricted version of the Sankoff algorithm (the Sankoff algorithm is very inefficient; it runs in $O(n^3m)$ time and uses $O(n^2m)$ memory, where $n$ is sequence length and $m$ is the number of sequences), such as foldalign (Gorodkin et al., 1997b, 2001) or dynalign (Mathews and Turner, 2002a).

3.2.3 Aligning secondary structures

The third approach is applied when there is no (or very little) sequence similarity. The sequences are folded individually and the predicted secondary structures are aligned using a program such as RNAforester (Hochsmann et al., 2003, 2004).

ConStruct (Lück et al., 1999) has a slightly different approach to the problem. A base-pair probability matrix is computed for each of the input sequences. The sequences are aligned using a multiple sequence alignment tool and the base-pair probability matrices are aligned correspondingly. The advantage of ConStruct is a graphical user interface that helps the user to adjust the alignment, so as to be able to sum the probability matrices into a consensus probability matrix from which the consensus structure can be derived.

The consensus structure predicted by any of the three approaches should not be regarded as the native structure for any of homologous sequences, but merely as part of the individual structures that generally has additional base-pairs to those in the consensus structure. The consensus structure can be used to infer the structure of the individual sequences, e.g. by constrained secondary structure folding (which can be achieved using RNAfold –C or by constraining certain base-pairs in mfold).
4. Analysis

To be able to develop new and better algorithms for identification and structure prediction of RNA, more knowledge of RNA is needed. By analyzing functional RNAs and identify properties that are unique to functional RNAs in general but also to specific RNAs, much knowledge can be gained. This information can be used in future RNA gene identification or classification, but it can also help in the understanding of how RNAs function in the cell.

4.1 Analysis of energy landscape

Here, we will denote the Boltzmann ensemble of all possible secondary structures of an RNA, where each secondary structure is associated with a free energy value as computed by an energy model, the energy landscape of the RNA. The MFE structure is the global minimum in the energy landscape, i.e. the structure with the lowest free energy among all structures in the ensemble. However, the MFE is not necessarily the native structure and also other locally optimal structures in the energy landscape are of interest. Locally optimal structures are structures that have lowest free energy among all its neighboring structures (that differ by only one base-pair), i.e. locally optimal structures are found at the bottom of energy valleys in the energy landscape. Deep valleys (i.e. large energy difference between a structure and its closest neighbors) correspond to stable structures (high energy barrier to leave the structure), whereas a shallow valley indicates an unstable structure. An energy landscape with one very deep valley indicates a stable, uniquely defined structure likely to be functionally important, whereas a rugged energy landscape with no significant deep valleys indicate that the investigated RNA has a very vaguely defined structure, perhaps the structure is not at all important to the function of the RNA. An energy landscape with two deep, clearly separated deep valleys indicates that the RNA might have two alternative functionally important structures, i.e. that the RNA can undergo a conformational switch.

The folding energy landscape of an RNA is important for understanding the folding kinetics of the RNA and also for locating the final functional structure. The great advantage of the RNA folding problem is that it is computationally feasible to compute the MFE structure, enumerate, and compute the partition function for all structures in the ensemble using dynamic programming (see section 3.1). In this section we will discuss the significance of the MFE and what can be learned by studying the folding energy landscape of an RNA.
4.1.1 Thermodynamic stability of functional RNA

In the past couple of decades there has been a discussion of whether functional ncRNAs can be distinguished from random sequence thanks to their unique folding abilities or not. The discussion started in the late 1980’s when Maizel et al (Le et al., 1988, 1989; Chen et al., 1990) developed programs for assessing the significance of RNA secondary structures, suggesting that thermodynamic stability could be used to identify functional RNAs. In 1999 Seffens and Digby (Seffens and Digby, 1999) presented a study showing that mRNA has significantly lower MFE than random sequences of the same mononucleotide frequencies. A couple of months later Workman and Krogh (Workman and Krogh, 1999) published a paper examining 46 of the 51 mRNAs studied in (Seffens and Digby, 1999) showing that mRNA does not have significantly lower MFE than random sequences with the same mono- and dinucleotide frequencies. In (Schultes et al., 1999) and (Rivas and Eddy, 2000) it was shown that ncRNAs do have more stable secondary structures (lower MFE) than randomized sequences preserving mononucleotide frequencies. However, Rivas and Eddy (2000) argue that secondary structure alone is not statistically significant for detection of ncRNA. Also (Clote et al., 2005) and paper I confirm that the MFE of functional RNAs is lower than random (preserving mono- and dinucleotides) sequences. (Bonnet et al., 2004) show that a special type of RNAs, the pre-miRNAs, have considerably lower MFE than shuffled sequences (preserving dinucleotide frequencies), an observation that we confirm in paper I. This is probably due to the very stable secondary structure of the pre-miRNAs, a long hairpin structure. In (Washietl et al., 2005) it is shown that sequence stability together with comparative information is an useful tool for general ncRNA gene identification.

Mono- and dinucleotide frequency conservation

In early studies comparing ncRNA to random sequences, the random sequences were generated so that they preserved the frequencies of nucleotides of the original sequence, i.e. the mononucleotide frequencies. In (Workman and Krogh, 1999) it is argued that since the free energy of an RNA structure depend on stacking base-pairs it is not enough to preserve mononucleotides, also dinucleotides, i.e. frequencies of neighboring nucleotides, need to be preserved. The explanation comes from the fact that the dinucleotide distribution in DNA sequences deviates from what would be expected based on the mononucleotide frequencies alone (Karlin and Mrázek, 1997). A program for shuffling a sequence preserving its mono- and dinucleotide frequencies called shuffle is available in Sean Eddy’s squid package (downloaded from http://selab.wustl.edu/cgi-bin/selab.pl?mode=software#squid).
Measure of the significance of the MFE

The MFE of an RNA sequence is a measure of the stability of the sequence, but the MFE alone does not say much. The MFE depends on the sequence length and the nucleotide compositions in the sequence, the mono- and the dinucleotide frequencies. A very simple measure of sequence stability is the normalized MFE value, i.e. the MFE value normalized by the sequence length. However, even the normalized MFE values will vary a lot for random sequences, depending on their nucleotide compositions (see paper I).

A more reliable measures of the significance of a MFE value are the Z-score, that compares the MFE of the native sequence and shuffled versions of the sequence. Here it is important that both the mono- and the dinucleotide frequencies are preserved.

The Z-score measures the number of standard deviations by which the MFE of the native sequence, s, deviates from the mean MFE of the shuffled sequences, \(X_{\text{shuffled}}\), i.e.

\[
Z(s) = \frac{MFE(s) - \mu}{\sigma},
\]

where \(\mu\) and \(\sigma\) are the mean and standard deviation, respectively, of the MFE of all the sequences in \(X_{\text{shuffled}}\).

4.1.2 Uniqueness of fold

In the previous section we discussed how the thermodynamic stability of a ncRNA compares with that of a random RNA, and whether the MFE of an RNA is sufficient to distinguish the ncRNA sequence from random sequences. However, not only is the stability measured in MFE important for an RNA, but also the uniqueness of the fold. Functional RNA structures tend to be both thermodynamically stable and uniquely folded (Draper, 1996). Schultes et al. (1999) show that ncRNA sequences are more ordered than random sequences. Well-defined structures have one deep valley in the folding energy landscape, whereas random sequences and not so well-defined sequences will have multiple valleys. Multiple valleys can also indicate that the RNA has alternative folds, i.e. possible conformational switches such as found in riboswitches. Another explanation for multiple valleys can be the folding process. That is to say maybe the RNA does not have a unique fold in vitro, but in vivo the folding path might be constrained by chaperones, other interacting molecules, or by the transcription process itself leading to a unique fold.

Kitagawa et al. (2003) present a study of human spliceosomal snRNAs, where the energy landscapes of the snRNAs are studied. The study shows that one of the snRNAs (U1) has a multi-valley energy landscape, whereas the other four investigated snRNAs have well-defined structures with only one steep valley in the energy landscape. Kitagawa et al. (2003) define a statistic, the so called valley index (VI), that measures the well-definedness of an RNA,
see paper I and (Kitagawa et al., 2003) for a definition. The study compares the snRNAs with shuffled versions of the snRNAs (the shuffled sequences preserve the mono- but not the dinucleotide frequencies in the snRNAs) and show that the shuffled sequences have different patterns in their energy landscapes than biological snRNAs do. However, in their study the shuffled sequences only preserve the mononucleotide frequencies of the biological snRNAs.

4.1.3 Measures of well-definedness

We study the well-definedness of an RNA structure by measuring the uncertainty in the fold. By computing the partition function of an RNA (McCaskill, 1990) (see section 3.1.4) the base-pair probabilities can be computed and presented in a dot plot (see Figure 3.1). The dot plot is useful for studying the base-pairing probabilities of a particular RNA, but for a more large scale analysis a single number that summarizes the information in the dot plot is desirable. Here, we will present measures that summarize a dot plot in a single number.

The Shannon entropy (Shannon, 1948) can be used to compute the uncertainty of the base-pairings (Huynen et al., 1997; Schultes et al., 1999). The two measures of uncertainty (both based on the Shannon entropy) defined in (Huynen et al., 1997) and (Schultes et al., 1999) differ slightly. However, their difference is small and only a matter of normalization. In paper I we use the following definition of normalized entropy (here denoted \( Q \) to be consistent with paper I although the Shannon entropy will be denoted \( H \) in section 4.2, see equation 4.5);

\[
Q = -\frac{1}{n} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} p_{ij} \log_2 p_{ij},
\]

(4.2)

where \( n \) is the sequence length and \( p_{ij} \) is the probability of a base-pair between nucleotides \( s_i \) and \( s_j \).

Another similar measure of well-definedness, also based on the base-pair probabilities, is the average base-pair distance or the ensemble diversity, as it is called in the RNAfold output. The average base-pair distance is computed as the average base-pair distance between all structures in the Boltzmann ensemble. The base-pair distance, \( d_{BP}(S, T) \), between two structures \( S \) and \( T \) compatible with \( s \), is defined as the number of base-pairs by which \( S \) and \( T \) differ, i.e.

\[
d_{BP}(S, T) = |S \cup T| - |S \cap T|.
\]

(4.3)

The average base-pair distance, normalized by sequence length, can be computed as follows (pers. comm. I. Hofacker);

\[
D = \frac{1}{n} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} p_{ij}^2 - p_{ij},
\]

(4.4)
for details of how this equation was derived see paper I.

The valley index defined in (Kitagawa *et al.*, 2003) is an average distance computed between a set of suboptimal structures. If this index is modified to compute the average base-pair distance instead of an average tree distance it is merely an approximation to $D$ (paper I). The two measures $Q$ and $D$ are strongly correlated (see paper I).

4.1.4 Software exploring the energy landscape

In this subsection we will present a couple of programs that can be useful for studying the energy landscape of an RNA molecule. The programs can identify RNAs that have one stable and unique configuration, RNAs that, at least *in silico*, do not have a stable structure at all, as well as conformational switches with two distinct stable secondary structures.

Two of the programs, *paRNAss* and *barriers*, are approximate methods in the sense that they do not investigate the entire Boltzmann ensemble of an input RNA. Instead, these two programs require an input set of secondary structures. For small RNAs this is not a problem as all structures in the ensemble (or sufficiently many) can be generated by a program such as RNAsubopt, but for larger RNA sequences this is not possible as the number of structures in the ensemble will be to high (and of course the programs would be very slow with billions of sequences in the input).

**paRNAss**

*paRNAss* (Giegerich *et al.*, 1999; Voss *et al.*, 2004) is a method for studying properties of the energy landscape, specially designed for predicting an RNA conformational switch (*paRNAss* is short for Prediction of Alternating RNA Secondary Structures). The *paRNAss* approach follows the following procedure; Secondary structures are sampled from the structure space using RNAsubopt (Wuchty *et al.*, 1999) or mfold (Zuker, 1994). Pairwise distances are calculated between the sampled structures using two different distance measures (e.g. energy barrier, morphological, tree alignment or string edit distance). Using a standard clustering method the structures are clustered into two clusters based on the distance measures. If the RNA is a conformational switch, then it has two stable structures and hence two separate clusters are expected (in a multi-switch, more than two stable structures are expected). As an additional test, the consensus structure of the clusters are computed and for each sample structure the distances to the two consensus structures are plotted against each other. If the RNA is really a conformational switch, then this *paRNAss* plot should display two clouds of points – one near the $x$-axis and one near the $y$-axis.

Note that since *paRNAss* calls RNAsubopt, it requires a user-defined energy bound, $E$, in order to generate all secondary structures within $E$ kcal/mol of the minimum free energy.
RNAshapes

RNAshapes (Giegerich et al., 2004; Steffen et al., 2006; Voss et al., 2006) introduce the concept of abstract RNA shapes and is by the authors suggested to replace paRNAss. An abstract RNA shape is a representation an RNA secondary structure. RNAshapes offers 5 different levels of abstraction, but common to all levels is that the length of stems and unpaired regions are not included. For example, at the highest level of abstraction the two structures

```
.(((.(((....))...((...))))))... 
.(((.......))...((...))...) 
```

will both be represented by the abstract shape `[[]][]`.

The RNAshapes package implements three functionalities;
1. It computes a set of abstract shapes (usually within an energy range from the MFE) and a shape representative (shrep) for each shape, where the shrep is the structure with the minimum free energy within the shape class (Giegerich et al., 2004).
2. It computes shape probabilities (Voss et al., 2006).
3. It computes a consensus structure for a set of RNA sequences as an abstract shape in common to all sequences.

Especially (2) can be very useful for identification of conformational switches. It is shown in (Voss et al., 2006) that example conformational switches show two shapes with probabilities 2/3 and 1/3, approximately.

barriers

The program barriers computes local minima and energy barriers in an energy landscape from an input set of structures. The input to barriers is a list of secondary structures and their free energies, sorted according to energy, such a list can be produced by e.g. the program RNAsubopt. barriers identifies local minima and energy barriers within the set of input structures.

Sfold

Sfold (Ding and Lawrence, 2003; Ding et al., 2005) implements statistical sampling or RNA secondary structures from the Boltzmann ensemble. The idea of Sfold is to sample 1000 sequences and then cluster the sampled structures according to base-pair distance. The accumulated probability of each cluster is computed. In (Ding et al., 2005) it is proposed that each cluster should be represented by a centroid structure instead of the MFE structure. The centroid structure of a set of structures is the structure with the minimum total base-pair distance to the structures in the set. Sfold can be useful for detecting conformational switches as two separate clusters that both have relatively high probability indicates that the RNA has alternative folds.
**RNAbor**

In paper III and IV we present a novel algorithm implemented in the program **RNAbor** for studying the folding energy landscape of an RNA. **RNAbor** enumerates and computes the accumulated probability of δ-neighbors of a given input structure. A δ-neighbor to the structure \( S \) is a structure, \( T \), that has a base-pair distance of δ to \( S \), i.e. \( \text{d}_{BP}(S, T) = \delta \). **RNAbor** does also compute a structure with the minimum free energy among all δ-neighbors for each value of δ.

The input structure to **RNAbor** can for example be the MFE structure. The **RNAbor** output can be displayed in a probability density plot showing accumulated probability as a function of δ. This density plot is a two-dimensional view of the energy landscape that can reveal the property of the input sequence (see Figure 4.1), in particular regarding the input secondary structure. A single high probability peak for small values of δ indicates that the RNA has a well-defined functional structure either with the exact structure as given in the input or a very similar structure. A density plot that instead has several low probability peaks suggests that the analyzed RNA has a multi-valley energy landscape and that it does probably not have a functional structure.

\[ p(\delta) = \frac{1}{1 + e^{-\beta \Delta G(\delta)}} \]

\( d_{BP}(S, T) = \delta \)

\( \beta = \frac{1}{kT} \)

\( \Delta G(\delta) = -RT \ln \left( \frac{p(\delta)}{p(0)} \right) \)

**Figure 4.1:** This figure shows three examples of the **RNAbor** probability output plot. (a) shows the output for a let-7 miRNA, an example of a well-defined structure, (b) shows an output with two separate probability peaks for the S-adenosylmethionine riboswitch, and (c) is the output for an unstructured random sequence.

**RNAbor** can also show other types of density plots, e.g. a density plot with two distinct high probability peaks, (indicating a conformational switch, or a single high probability peak not located around \( \delta = 0 \). In these cases the ability of **RNAbor** to output a structure of minimum free energy among δ-neighbors is useful, as it can give an optimal structure for the δ values corresponding to high probability peaks.
4.2 Analysis of function

In this section we turn to the function of RNAs. We will not discuss prediction of function, but rather analysis of function (which is of course useful for future function prediction). We will try to answer the question of what sequence and structure features that are important to function, as well as the question of what features give similar RNAs unique functions. For this analysis we use various visualization tools, so called *logos*.

A sequence logo (Schneider and Stephens, 1990) summarizes a sequence alignment in a stacked bar graph with alignment position on the x-axis and information content on the y-axis. For each alignment position a stack of letters shows the frequencies of bases as the relative height of the letters and the total stack height displays the information in bits, i.e. a measure of sequence conservation. The letters in a stack are sorted according to frequency with the most frequent base on top, so that the consensus sequence can easily be read from the logo (the original main purpose of the sequence logos was to display a generalization of a consensus sequence (Schneider and Stephens, 1990)). See Figure 4.2 for an example sequence logo.

![Sequence Logo Example](image)

*Figure 4.2*: An example sequence logo. 149 sequences of E. coli Ribosome binding sites summarized in a sequence logo (this is Figure 2 of (Schneider and Stephens, 1990)).

The logo concept can be extended to visualize more than just a sequence alignment. In structure logos (Gorodkin *et al.*, 1997a) and Mfold (paper V) the logos are extended to display conserved secondary structure in addition to sequence information. In paper VI and VII we present a logo that highlights underrepresented features instead of overrepresented features as in the original...
sequence logos as well as a set of logos that visualize differences between a set of coaligned alignments (where each alignment represent a specific function).

4.2.1 Information

The information content (Cover and Thomas, 1991), displayed in the sequence logos, also known as the “degree of surprise” is computed as the entropy difference between a background distribution of states and the distribution in a set of observations.

**Sequence information content**

The entropy, as defined by Shannon (Shannon, 1948), measures the “uncertainty” and is defined as follows; if $X$ is a random variable over the values $x \in \mathcal{X}$ and $p(x) = P(X = x)$ is the probability density distribution of $X$, then the entropy of $X$ is

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log_2 p(x).$$  \hfill (4.5)

The probabilities, $p(x)$, are generally estimated by the frequencies of the states, $f(x)$. Hence, for alignment $y$ the entropy of alignment column $i$ is computed as;

$$H(X_i|y) = -\sum_{x \in \mathcal{X}} f_i(x|y) \log_2 f_i(x|y),$$  \hfill (4.6)

where $X_i$ is a random variable corresponding to observations in position $i$ and $f_i(x|y)$ is the frequency of state $x$ in column $i$ in alignment $y$ ($0 \leq f_i(x|y) \leq 1$, $\sum_{x \in \mathcal{X}} f_i(x|y) = 1$).

The information gained by aligning sequences (gaining the knowledge $y$) is measured as the difference in entropy between aligned and unaligned sequences (when $y$ is known and when it is not) (Schneider et al., 1986);

$$I(X_i|y) = H(X) - H(X_i|y) = -\sum_{x \in \mathcal{X}} f(x) \log_2 f(x) + \sum_{x \in \mathcal{X}} f_i(x|y) \log_2 f_i(x|y),$$  \hfill (4.7)

where $f(x)$ is the genomic frequency of $x$ (can also be some other background frequency of state $x$) ($0 \leq f(x) \leq 1$, $\sum_{x \in \mathcal{X}} f(x) = 1$).

**Sample size correction**

When the alignment is small (few sequences) the observations behind the observed frequencies $f_i(x|y)$ for $x \in \mathcal{X}$ are few and hence $f_i(x|y)$ is not a good approximation of the probability $p_i(x|y) = P(X_i = x|y)$. An entropy $H(X_i|y)$ based on frequencies measured from few observations will be too low. To compensate for this, a small sample size error correction (Schneider et al., 1986) (paper VI), $e(n_i(y))$, is added to $H(X_i|y)$, where $n_i(y)$ is the number of observations in position $i$ of alignment $y$. Hence, the information content is;

$$I(X_i|y) = H(X) - (H(X_i|y) + e(n_i(y))).$$  \hfill (4.8)
The error correction is of course not known exactly, but the error can be estimated. Ultimately the error correction would be computed as the difference between the expected value of the entropy given the sample size and the probabilities of the different states in the given position and the entropy computed directly from the probabilities, but since the probabilities are unknown the error is instead computed using the prior probabilities of states (background frequencies), $f(x)$ for $x \in X$ (Schneider et al., 1986).

**Skewed genomes**

For skewed genomes (or an for some other reason uneven prior distribution) an alternative formula for computing the information content is described in (Schneider et al., 1986) and adopted in for example (Gorodkin et al., 1997a) and paper V. The alternative information definition is:

$$I_{\text{skew}}(X_i|y) = - \sum_{x \in X} f_i(x|y) \log_2 \frac{f_i(x|y)}{f(x)},$$

(4.9)

for equiprobable states this formula will give the same values as equation 4.7. The problem with equation 4.7 is that the computed information can be identical in very different situations, e.g. if A has a very high prior probability or if instead C has a very high prior probability will not influence the information content. The alternative information measure in equation 4.9 is actually computed as the Kullback-Leibler divergence (Cover and Thomas, 1991) (see equation 4.13) between the position and alignment specific distribution, $p_i(x|y)$, and the prior distribution, $p(x)$, i.e.

$$I_{\text{skew}}(X_i|y) = KL(p_i(x|y)||p(x)),$$

(4.10)

if the distributions were known. Here, we will instead use the observed frequencies for computing the information (Kullback-Leibler divergence) value.

In paper V we adopt the information measure defined in equation 4.9, since we want to be able to compare the logos created by MIfold with structure logos (Gorodkin et al., 1997a). In paper VI and VII we are using the former information definition including the sample size correction (equation 4.8).

4.2.2 Covariation measures

As discussed in section 3.2, various measures of covariation can be used for structure prediction, either alone or in combination with an energy model. The same measures that are used for structure prediction can also be used for visualizing the covariation, and hence potential base-pairings, in a sequence.

In MIfold (paper V), mutual information (equation 3.15), the (straight forward) covariance measure adopted in RNAalifold (Hofacker et al., 2002), or a Kullback-Leibler divergence between observed and expected base-pair distributions, can be used for highlighting regions with strong covariation.
Mutual information
For two alignment columns $i$ and $j$ the mutual information is computed based on observed frequencies as;

\[
I_{i,j}^{\text{mut}} = I^{\text{mut}}(X_i, Z_j) = \sum_{x \in X_{\text{RNA}}} \sum_{z \in X_{\text{RNA}}} f_{i,j}(xz) \log_2 \frac{f_{i,j}(xz)}{f_i(x)f_j(z)}, \tag{4.11}
\]

where $X$ and $Z$ are two random variables over the event space $\mathcal{X}$, i.e. over the states $x, z \in \mathcal{X}$ that can be observed in columns $i$ and $j$. See section 3.2.1 for details on the mutual information.

Covariance as in RNAalifold
As described in section 3.2 the covariance measure adopted in RNAalifold (Hofacker et al., 2002),

\[
C_{i,j} = C(X_i, Z_j) = \sum_{xz \in \mathcal{B}} \sum_{x'z' \in \mathcal{B}} f_{i,j}(xz)D(xz, x'z')f_{i,j}(x'z'), \tag{4.12}
\]

can be used to measure covariation between two alignment columns, see equation 3.17.

Kullback-Leibler divergence
The Kullback-Leibler divergence is a measure of the distance between two distributions $p(x)$ and $q(x)$, $x \in \mathcal{X}$, defined as;

\[
KL(p || q) = \sum_{x \in \mathcal{X}} p(x) \log_2 \frac{p(x)}{q(x)}. \tag{4.13}
\]

The “mutual information” measure adopted in the structure logos (Gorodkin et al., 1997a) is the Kullback-Leibler divergence between observed and expected frequencies of complementary bases. If $f_{i,j}^{\text{comp}}$ denotes the frequency of complementary bases in alignment columns $i$ and $j$, the covariance measure is computed as;

\[
M_{i,j} = f_{i,j}^{\text{comp}} \log_2 \frac{f_{i,j}^{\text{comp}}}{E[f_{i,j}^{\text{comp}}]} + (1 - f_{i,j}^{\text{comp}}) \log_2 \frac{1 - f_{i,j}^{\text{comp}}}{1 - E[f_{i,j}^{\text{comp}}]}, \tag{4.14}
\]

where $E[f_{i,j}^{\text{comp}}]$ is the expected base-pair frequency computed based on the frequencies in the columns $i$ and $j$

\[
E[f_{i,j}^{\text{comp}}] = \sum_{x_i \in X_{\text{RNA}}} \sum_{x_j \in X_{\text{RNA}}} \Delta(x_i, x_j)f(x_i|y)f(x_j|y), \tag{4.15}
\]

where \(\Delta(x_i, x_j)\) is defined as in equation 3.11, i.e. equal to 1 if $x_i$ and $x_j$ can form a base-pair and 0 otherwise.
Note that $M_{i,j}$ will only take into account covariance due to base-pairs, unlike the mutual information $I_{i,j}^{\text{mut}}$ (equation 4.11) and the covariance measure $C_{i,j}$ (equation 4.12). To reduce the noise an inconsistent sequences penalty, $q_{i,j}$, is introduced in RNAalifold (Hofacker et al., 2002).

$$q_{i,j} = 1 - f_{i,j}^{\text{comp}} - f_{i,j}^{\text{gap}},$$  \hspace{1cm} (4.16)

where $f_{i,j}^{\text{comp}}$ is the frequency of complementary bases in alignment columns $i$ and $j$ and $f_{i,j}^{\text{gap}}$ is the frequency of alignment sequences with gaps in both column $i$ and $j$. By subtracting $q_{i,j}$ from $I_{i,j}^{\text{mut}}$, or $C_{i,j}$ the noise can be reduced resulting in improved measures of covariation due to base-pairing (see paper V).

### 4.2.3 Sequence logos

The sequence logos described by Schneider and Stephens (1990) summarize a sequence alignment. The alignment can be an amino acid or nucleic acid sequence alignment. For every column in the alignment a stack of letters is displayed in the logo, where the total stack height displays the information in bits computed according to equation 4.8. In the original sequence logo implementation (Schneider et al., 1984; Schneider and Stephens, 1990) it is assumed that all states $x \in X$ are equiprobable a priori. This means that the information is computed as

$$I(X_i|y) = \log_2 |X| - (H(X_i|y) + e(n_i(y))).$$  \hspace{1cm} (4.17)

However, in many extensions to the sequence logos equation 4.9 (Gorodkin et al., 1997a) (paper V) or equation 4.8 (paper VI and VII) are adopted instead so as to include prior probabilities. The sum in the entropy equations is taken over the set of possible states, i.e. either $X_{\text{protein}} = \{A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y\}$, $X_{\text{DNA}} = \{A, C, G, T\}$, or $X_{\text{RNA}} = \{A, C, G, U\}$, depending on the type of alignment.

The letters in a stack are sorted according to their relative frequency, with the most frequent base on top, so that the ‘consensus’ sequence can be read from the topmost letters. The height of a letter is proportional to the frequency of the corresponding base, i.e. the height $h_i(x|y)$ of symbol $x \in X$ in alignment column $i$ in alignment $y$ is

$$h_i(x|y) = f_i(x|y)I(X_i|y).$$  \hspace{1cm} (4.18)

In Schneider’s original logos, gaps are ignored when the information content is computed, but the height of the stack is reduced to $a\%$ of the computed height if only $a\%$ of the sequences are ungapped.

In paper VI and VII we will use equation 4.8 to compute the information content, so as to be able to include a prior distribution of states. Furthermore,
we include gaps as a fifth state, i.e. the set of possible states is $\mathcal{X}_{\text{DNA}} = \{A, C, G, T, -\}$, or $\mathcal{X}_{\text{RNA}} = \{A, C, G, U, -\}$.

Sequence logos can be created using the Pascal program \texttt{MakeLogo} in the Delila package (Schneider \textit{et al.}, 1984), that creates a postscript figure like the one in Figure 4.2.

### 4.2.4 Structure logos

The program RNA structure logo (Gorodkin \textit{et al.}, 1997a) displays an alignment in a similar fashion as the sequence logos do (Schneider and Stephens, 1990), but prior frequencies are taken into account when computing the information content, gaps are not ignored and a covariance value is displayed in base-paired regions.

The input to structure logos is an RNA alignment with consensus structure information. The information content is computed using equation 4.9, so as to take the prior frequencies into account. In addition to this gaps are taken into account and are displayed as a fifth character. In equation 4.9 the sum is taken over the five states $\mathcal{X} = \{A, C, G, U, -\}$, where $-$ denotes the gap state. The prior distribution $f(x)$, $x \in \{A, C, G, U\}$ should be set by the user to the distribution of the bases in the entire genome or similarly, otherwise $f(x) = 0.25$ for the four bases. The prior for the gap state is set to $f(-) = 1$, since $f_i(x|y) \log_2 f_i(x|y) = 0$ if $f_i(x|y) = 0$ or 1.

A new (optional) ordering of the bases in a stack is introduced in the structure logos. The bases are ordered according to the symbol heights and the symbols heights are displayed proportional to the frequencies relative to the \textit{a priori} frequencies, i.e.

$$h_i(x|y) = \frac{f_i(x|y)/f(x)}{\sum_{x \in \mathcal{X}} f_i(x|y)/f(x)} I(X_i|y). \quad (4.19)$$

In base-paired regions of the alignment the contribution from a covariance measure, $M_{i,j}$ (see equation 4.14 for a definition) is added to the computed information value. In the logos the additional value is displayed with the letter $M$ on top of the ordinary stack. The height of the letter $M$ is $M_{i,j}/2$ in alignment columns $i$ and $j$. The covariance term is only calculated for pairs of positions that are base-paired according to the input structure.

### 4.2.5 MIfold

In paper V we describe the MATLAB toolbox MIfold. MIfold is a tool that is mainly useful for visualization of RNA alignments, but it can also be used for structure prediction. As a structure prediction tool MIfold requires a relatively large alignment (many sequences) or an alignment with large sequence diversity, as it bases its prediction on covariance signal only, no thermody-
namic information is taken into account. All of the three covariation measures described in section 4.2.2 are implemented in MIfold and can be used for structure prediction and visualization. Also the inconsistent sequences penalty (equation 4.16) is implemented and can be combined with any of the three covariation measures.

The secondary structure is predicted in a Nussinov (Nussinov and Jacobson, 1980) like algorithm where the base-pairs are weighted by their mutual information score. First level pseudoknots (see section 1.2) can be predicted by simply running the algorithm twice. See paper V for details.

The structure prediction abilities of MIfold (paper V) are described in in paper V. Here, we will describe the MIfold as an alternative visualization tool to the structure logos (Gorodkin et al., 1997a).

The information content is computed in the same way as for the structure logos, i.e. using equation 4.9. The prior distributions can be set by the user or else the prior distribution will be computed from the entire alignment. In MIfold the gap state is treated in the same way as the nucleotide states and hence also \( f(-) \) is by default computed from the alignment and the information content contribution from the gap state is displayed in the logo.

The MIfold information content visualization differ a bit from the sequence and structure logos. The information contribution from a single base \( x \) in a particular position \( i \) is computed as;

\[
I_{\text{skew}}(X_i = x|y) = - f_i(x|y) \log_2 \frac{f_i(x|y)}{f(x)}
\] (4.20)

The information contributions are displayed in a bar graph, where positive contributions (indicating overrepresented bases) are displayed above the x-axis and negative contributions (indicating underrepresented bases) are displayed below the x-axis. The gap state is also displayed in this view (see Figure 4.3 for an example).

In the sequence logo the 'consensus' sequence can be read from the logo as the topmost letters in every stack. This consensus is just the most frequent bases in each position. Instead of displaying a consensus sequence MIfold computes and prints the most informative sequence. The most informative sequence is a single sequence that for each column in the alignment contain the character in the IUPAC (International Union of Pure and Applied Chemistry) code (Cornish-Bowden, 1985) that correspond to the set of characters whose information contribution exceed a (user defined) threshold.

The secondary structure predicted based on the selected mutual information measure (one of the three; mutual information (equation 4.11), covariance score (equation 4.12) and the Kullback-Leibler distance between observed and expected base-pairing frequencies (equation 4.13)) is displayed in a mountain-plot (Hogeweg and Hesper, 1984) (see section 1.2.1) that is incremented by the mutual information score corresponding to the base-pair instead of just 1 as in the classical mountain plot. Pseudoknots are then added in the plot as
Figure 4.3: Screenshot of the mountainplot visualization in MIfold. The example is based on an alignment of 42 sequences of the prion protein mRNA pseudoknot (Barret et al., 2001). The 3’ helix is displayed as a mountain plot, whereas the base-pairs in the 5’ helix are represented by sloping lines. The reference structure is shown in red (only the 3’ helix is displayed). Below the mountain plot view, a bar graph shows the information content in the various alignment positions.

straight lines connecting the positions in the mountain plot corresponding to the interacting bases (see Figure 4.3).

4.2.6 tRNA logos
In paper VI and VII we describe four different types of sequence logo extensions that we apply on the tRNA identity problem for identification of potential new determinants and antideterminants. These logo extensions are not specific to the tRNA identity problem, although they were designed specifically to study tRNA identity.

tRNA identity
Specific amino acids are attached to specific tRNAs by enzymes called aminoacyl-tRNA synthetases (aaRS). The tRNA identity is defined as the amino acid 'charging' specificity of a tRNA and is crucial for a correct protein synthesis during translation. In general, there is one aaRS for each of the 20 canonical amino acids. This means that the tRNAs can be divided
into 20 functional classes based on their charging identity, but actually there are more than 20 classes, since there are a couple of special tRNAs like the initiator tRNA and non-canonical amino acids such as selenocysteine.

The sequence features in a tRNA that promote recognition of tRNA by a specific aaRS are called tRNA determinants. Sequence features that instead prevent interaction with other aaRSs are called tRNA identity antideterminants. Together we call the determinants and antideterminants tRNA identity elements.

Although the anticodon and acceptor stem (see Figure 1.6) are important for tRNA identity, the tRNA identity elements are distributed over the entire tRNA sequence in different places depending on identity class. The identity elements do also vary between the three domains of life, and even within Bacteria there is a variation (paper VII).

Here, we define a sequence feature as a nucleotide in a particular position in an aligned sequence, or the absence of a nucleotide (a gap), i.e. if a sequence in the alignment is denoted \( x = x_1 x_2 \ldots x_n \), where \( n \) is the alignment length, then \( x_i \in X \) is the sequence feature at alignment position \( i \), where \( X \) is the set of possible states, i.e. in an RNA alignment \( X = \{ A, C, G, U, – \} \).

The tRNAs are divided into subalignments according to their function (their tRNA identity). The set of possible functions is denoted \( Y \) and if each identity class is denoted with the IUPAC symbol (Cornish-Bowden, 1985) corresponding to the amino acid the tRNA is charged with, then \( Y = \{ A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, X, Y \} \), where \( X \) denotes the initiator tRNA (not a IUPAC symbol), to include non-canonical amino acids this set can be extended further.

**Function logos**

One sequence logo per functional class can be created to visualize overrepresented sequence features, but to visualize the differences between the function classes we introduce the function logos (paper VI). A sequence logo displays information of states \( X \) for a given function \( y \in Y \). A function logo does instead display information of functions \( Y \) for a given state \( x \in X \).

The information content displayed at position \( i \) in a function logo is the information that a state \( x \) in position \( i \) (i.e. a feature \( x_i \) ) confers about the frequencies of sequences belonging to to the functional classes in \( Y \), i.e.

\[
I(Y|x_i) = I(Y|X_i = x) = H(Y) - (H(Y|X_i = x) + e(n_i(x))) =
\]

\[
= - \sum_{y \in Y} f(y) \log_2 f(y) + \sum_{y \in Y} f_i(y|x) \log_2 f_i(y|x) - e(n_i(x)), \quad (4.21)
\]

where \( f(y) \) is the overall frequency of sequences belonging to the functional class \( y \) \( (0 \leq f(y) \leq 1, \sum_{y \in Y} f(y) = 1) \), \( f_i(y|x) \) is the frequency of sequences belonging to class \( y \) among all the sequences that have state \( x \) in position \( i \) \( (0 \leq f_i(y|x) \leq 1, \sum_{y \in Y} f_i(y|x) = 1) \), and \( n_i(x) \) is the number of sequences that have state \( x \) in position \( i \).
In the function logo the stack in position \( i \) consists of letters from \( \mathcal{Y} \) whose relative heights are proportional either to \( f_i(y|x) \) or to \( f_i(y|x)/f(y) \) (in analogy with equation 4.18 or 4.19 (Gorodkin et al., 1997a)).

**Inverse logos**

Both sequence logos and function logos visualize overrepresented features/functions, the inverse logos do instead visualize underrepresented features (or functions in the inverse function logos). The inverse logo can be either a sequence or a function logo, where the relevant frequency distributions have been transformed so as to make small frequencies large and large frequencies small.

In paper VI we define two types of inverse transformations; the reciprocal and the simplex inverses. They both have their advantages, but for relatively large biological data sets the reciprocal inverse is probably the most useful, as it is less sensitive to small variations among states observed at high frequencies. The reciprocal inverse of a frequency distribution \( f(x) \) for \( x \in \mathcal{X} \) based on \( n \) observations is defined as:

\[
f'(x) = \frac{1}{f(x)n + q},
\]

where \( q \) is a pseudocount value defined as:

\[
q = \begin{cases} 
1 & \text{if } \min_{x \in \mathcal{X}} f(x) = 0 \\
0 & \text{if } \min_{x \in \mathcal{X}} f(x) > 0.
\end{cases}
\]

**ID logos**

The function logos can be useful for visualizing tRNA identity elements within a taxon (or some other subset of tRNA sequences). Differences in the tRNA identity code between two taxa can be discovered by comparing the sets of function logos for the two taxa. To further isolate the differences we introduced two new data reduction visualizations in paper VII; the information difference (ID) logos and the Kullback-Leibler divergence difference (KLD) logos.

The ID logos display the functional information difference between two taxa, thus highlighting gains and losses of functional information associated to features in one lineage. The information difference \( \Delta I(Y|X_i = x) \) is simply computed as the difference in information of the foreground dataset (superscripted by \( F \)) and the background dataset (superscripted by \( B \));
\[ \Delta I(Y|X_i = x) = I^F(Y|X_i = x) - I^B(Y|X_i = x) = \\
= [H^F(Y) - H^F(Y|X_i = x) - e(n^F(x_i))] - \\
[H^B(Y) - H^B(Y|X_i = x) - e(n^B(x_i))]. \quad (4.24) \]

**KLD logos**

The ID logos can not display all sorts of difference between two distributions. For example, if the foreground and background datasets differ by a functionality shift, the ID logo will not show a signal since the information value for the two datasets are the same. An example of a functional shift is if A in position 51 is associated with tRNA\textsuperscript{Glu} in the foreground data set, but with tRNA\textsuperscript{Asp} in the background dataset, the information difference can still be zero. To be able to visualize such functional shifts we introduce the KLD logos.

The Kullback-Leibler divergence (see equation 4.13) between the probability distributions \( p^F(y|x_i) \) and \( p^B(y|x_i) \) is a measure of the divergence between the two data sets. Since we want to visualize site specific differences we subtract the Kullback-Leibler divergence for the prior probability distributions (class size differences). The Kullback-Leibler divergence difference (KLD) value is thus defined as

\[ \Delta KL^\text{func}(Y|X_i = x) = \\
= KL(p^F(y|x_i)||p^B(y|x_i)) - KL(p^F(y)||p^B(y)) - e_{KL}(n^F(x_i)) \quad (4.25) \]

\[ \approx \sum_{y \in Y} f^F_i(y|x) \log_2 \left( \frac{f^F_i(y|x)}{f^B_i(y|x)} \right) - \sum_{y \in Y} f^F(y) \log_2 \left( \frac{f^F(y)}{f^B(y)} \right) - e_{KL}(n^F(x_i)), \]

where \( e_{KL}(n^F(x_i)) \) is a small sample size error correction.

In paper VII we show how the ID and KLD logos can visualize different types of differences between Proteobacteria and Cyanobacteria.
5. Summary of papers

Paper I
In paper I we show that ncRNA, but not mRNA, has lower MFE than random sequences with the same mono- and dinucleotide composition. We compare three statistics, the $Z$-score, the $p$-value and the normalized energy $dG$, that all measure the stability of an RNA in terms of MFE, either compared to random sequences or just normalized by sequence length. These statistics are computed for a large set of RNA families showing that the miRNAs are the most stable, while the Hammerhead type I RNAs behave more like unstructured RNAs.

Also the uniqueness of a fold is investigated for the various RNA families in terms of three statistics that are all based on the partition function; the normalized Shannon entropy (Schultes et al., 1999), $Q$, the average base pair distance, $D$, and a modified version of the Valley Index first introduced in Kitagawa et al. (2003), $VI$. We show that many ncRNAs do not have a single unique fold, but rather several alternative folds, at least when folded in silico.

Paper II
Paper II describes a benchmark study evaluating the performance of homology search methods on ncRNA. Many of the available homology search methods are developed for searching for homologous protein sequences or protein coding sequences, but yet they are commonly used also for searching for ncRNAs. We investigate how appropriate these methods are for ncRNA searching and compare them with methods that are specially designed for the ncRNA homology search problem, including secondary structure information in the search criteria. We show that sequence based methods such as BLAST and FASTA are much faster, but less accurate, than profile HMM methods, such as HMMer and SAM, and the most accurate structure based methods, such as Infernal.

Paper III
Paper III describes an algorithm, implemented in our program RNAbor, for investigating the structural neighborhood of an input secondary structure $S$ compatible with an RNA sequence $s$. A secondary structure $T$ of $s$ is called a
$\delta$-neighbor of $S$, if $T$ and $S$ differ by exactly $\delta$ base pairs. RNAbor computes both the number and the partition function contributions of $\delta$-neighbors for various values of $\delta$.

We investigate the structural neighborhood of known ncRNAs, such as tRNA, and use RNAbor for detecting RNA conformational switches.

**Paper IV**

Paper IV describes a web server implementing RNAbor, the program described in paper III.

**Paper V**

Paper V describes a tool, MIfold, for visualizing an alignment of functional RNAs and predicting a consensus structure based on mutual information, or another covariation measure, only. As a structure prediction tool, MIfold has a relatively high performance for alignments of many diverse sequences, but as the number of sequences or the sequence diversity decreases the performance drops, simply because in a data set with little or no variation the covariation measured will be insignificant. However, as a visualization tool MIfold can be used on any alignment. It shows the information content contribution from each nucleotide in every alignment column as well as the predicted consensus structure. The predicted secondary structure may contain a first level pseudoknot.

**Paper VI**

Paper VI describes two extensions to the sequence logos (Schneider and Stephens, 1990), the functional logos and the inverse logos. The functional logos display subfunctions (such as tRNA charging identity) that are overrepresented among sequences that carry a specific feature. Inverse logos display underrepresented features or functions instead of overrepresented as in the standard sequence and function logos.

In this paper we apply these logos on a database of structurally aligned bacterial tDNA sequences, creating logos that highlight potential determinants and antideterminants that confer specific tRNA charging (or initiator) identities. Both known and potentially new identity elements are detected.
In paper VII, tRNA identity elements are further investigated. Here, we contrast the identity elements of tRNAs in Proteobacteria and Cyanobacteria using two new data reduction visualizations. One, called Information Difference logos (ID logos), shows the evolutionary gain or retention of functional information associated to features in one lineage. The other, Kullback-Leibler divergence Difference logos (KLD logos), shows recruitments or shifts in the functional associations of features, especially those informative in both lineages. Using these two logos differences between the Proteobacterial and Cyanobacterial tRNA determinants are visualized.
6. Sammanfattning på svenska

En studie inom RNA-bioinformatik

Identifiering, prediktion och analys


Funktionellt RNA, s.k. icke-kodande RNA (ncRNA), har i allmänhet en struktur som, precis som i proteinfallet, är avgörande för dess funktion. Många forskarlag världen över ägnar sig åt att bestämma strukturen hos protein, både experimentellt och med hjälp av olika beräkningsmetoder. Idag finns även ett antal tredimensionella RNA-strukturer tillgängliga efter att de blivit experimentellt bestämda, även om antalet lösta proteinstrukturer är betydligt högre.


I denna avhandling studerar vi tre olika områden inom RNA-bioinformatiken; identifiering av RNA-gener, prediktion av RNA-sekundärstruktur, och analys av RNA-sekvenser, dels genom att studera energilandskapet (samtliga möjliga sekundärstrukturer och dessas fria energier) av en RNA-molekyl, dels genom visualisering av en mängd sekvenser i syfte att hitta gemensamma och särskiljande egenskaper.

Nu när sekvenseringsmetoderna blir allt snabbare och fler fullständiga genom görs tillgängliga ställs större krav på annoteringen av dessa nya genom. Det finns idag en rad tillförlitliga program som identifierar potentiella proteingener enbart baserat på genomsekvensen. Speciellt hos prokaryota organismer kan proteingener identifieras med hög tillförlitlighet tack vare
bl.a. promotorsekvenser. RNA-gener saknar promotorsekvenser och inte heller kan man som i proteinfallet söka efter tillräckligt långa öppna läsramar (ORF). I jakt på RNA-gener kan man istället, under antagandet att det funktionella RNA-t har en struktur som är nödvändig för dess funktion (vilket ofta är fallet), söka efter regioner som kodar för ett RNA som skulle kunna bilda en stabil och unik sekundärstruktur.


I denna avhandling vidareutvecklas DP-algoritmen som predikterar MFE-struktur till en helt ny algoritm för att undersöka energilandskapet hos ett RNA närmare. I papper III och IV presenterar vi programmet RNAbor för att studera strukturer i närheten av en given RNA-struktur, S, kompatibel med en given sekvens, s. Den givna strukturanen kan vara MFE-strukturen eller någon annan struktur av intresse, t.ex. en struktur bestämd med hjälp av jämförande sekvensanalys eller röntgenkristallografi. RNAbor studerar δ-grannar, där en δ-granne är en struktur T, kompatibel med s, som har basparsavstånd δ till S (d_BP(S, T) = δ). Vårt program beräknar antal (N^δ), bidrag till fördelningsfunktionen (partition function) (Z^δ), samt MFE (MFE^δ) och motsvarande MFE-struktur bland samtliga δ-grannar till S. N^δ, Z^δ och MFE^δ beräknas för samtliga värden på δ. Fördelningsfunktionen Z^δ är speciellt användbar då den kan användas för att beräkna sannolikheten för strukturer på olika δ-avstånd från S. Sannolikheten för strukturer på basparsavstånd δ är

\[ p^δ = \frac{Z^δ}{Z}, \]  

där Z är den totala fördelningsfunktionen, \( Z = \sum_δ Z^δ \).

En graf över sannolikheten \( p^δ \) som funktion av δ ger en tvådimensionell bild av energilandskapet, som kan synliggöra vissa egenskaper hos sekvensen s, speciellt med avseende på strukturen S. Exempelvis så visar en graf med endast en hög sannolikhetstopp nära δ = 0 på att S är en väldefinierad stabil struktur, sannolikt den funktionellt aktiva strukturen. En graf med två tydligt separerade sannolikhetstoppar däremot antyder att s har alternativa strukturer, som båda kan vara funktionellt aktiva. Visar grafen däremot ingen topp med hög sannolikhet alls, utan många mindre toppar, så är det troligt att s inte har någon funktionell struktur, åtminstone ingen som kan bildas utan en hjälp-molekyl.

en sekvensinpassning. I papper V, VI och VII presenterar vi en rad utökningar av sekvenslogokonceptet.

I papper V presenterar vi MATLAB-programmet MiFold, som dels visualiserar en RNA-sekvensinpassning, dels visualiserar och predikterar en sekundärstruktur baserat på samvariationsvärden. I MiFold kan användaren välja mellan tre olika mått på samvariation, för samtliga mått gäller det att strukturprediktionen blir bättre med en sekvensinpassning med fler sekvenser och lägre sekvensidentitet. Även för inpassningar med låg variation kan MiFold emellertid användas för att visualisera den samvariation som ändå finns.

Papper VI och VII behandlar tRNA-identitetsproblemet, d.v.s. identifiering av sekvenselement som är viktiga för att en tRNA-sekvens ska kännas igen av rätt enzym (aminoacyl-tRNA synthetase (aaRS)) och därmed sammankopplas med rätt aminosyra. I denna avhandling studeras en stor mängd bakteriella tRNA, som delats in i klasser baserat på identitet (dvs vilken aminosyra de ansvarar för att transportera till ribosomen). För att identifiera de sekvenselement som särskiljer dessa funktionella grupper kan en sekvenslogo skapas för varje klass, varefter dessa jämförs. Eftersom detta innebär minst 20 olika sekvenslogor (en för varje aminosyra, kanske en extra för det tRNA som initierar proteinsyntesen och kanske dessutom något tRNA associerat med en speciell, modifierad aminosyra), så blir det inte en lätt uppgift att hitta särskiljande drag. För att göra denna jämförelse enklare introducerar vi funktionslogo (function logo) i paper VI. En funktionslogo är liksom en sekvenslogo ett stapeldiagram, med position på x-axeln och informationsvärde på y-axeln, men istället för en logo per funktionsklass gör vi en logo per nukleinsyra (dvs en logo per symbol i inpassningen, även en för lucka (gap)). En funktionslogo visar överrepresenterade funktionsklasser bland sekvenser med en given bas. Papper VI presenterar ytterligare en utökning, nämligen inverslogo (inverse logo), som istället för att visa överrepresenterade funktionsklasser eller särdrag visar underrepresenterade dito.

Papper VII går ett steg längre i jämförelsen av tRNA-identiteter och jämför tRNA-identitetselement mellan olika taxa, mer specifikt mellan proteo- och cyanobakterier. För att kunna göra denna jämförelse inför vi två nya logokoncept; ID logo (information difference logo) och KLD logo (Kullback-Leibler divergence difference logo).
7. Acknowledgements

I would like to thank everyone who in some way has contributed to this thesis. In particular, I would like to thank:

- David Ardell and Vincent Moulton. I would like to thank my former supervisor Vincent Moulton for many discussions, sharing his many RNA ideas, introducing me to the field of RNA bioinformatics and for introducing me to interesting people in this field, some of whom are my collaborators today, and for remaining a colleague even after his change of country. I would also like to thank my current supervisor David Ardell for many discussions and ideas, for introducing me to new fields and people in RNA bioinformatics, and for supporting me in my continued work with external collaborators.

- my former and current cosupervisors Sverker Edvardsson and Gerhart Wagner for their support and help.

- all my current and previous collaborators, especially Paul Gardner for sharing many ideas of which we have implemented a few and for hosting me in Bielefeld and many times in Copenhagen; Peter Clote for discussions on RNA algorithms, general advice and for inviting me to and hosting me in Boston; and Anthony Poole and Tore Samulesson for teaching me some biology and for their help in the first years of my PhD studies.

- all my coauthors; Karl Andersson, Casey Bergman, Jonathan Bollback, Yuanyuan Cui, Mats Gustafsson, Maris Lapinsh, Olle Nilsson, Peteris Prusis, and Jarl Wikberg.

- everyone at the LCB for a friendly and stimulating work environment. In particular, I would like to thank Alice Lesser, for answering my questions regarding English, Swedish, mathematics etc.

- Pontus Freyhult for trying to answer all my programming questions and for providing computer support in general.

- family and friends for their support.


A doctoral dissertation from the Faculty of Science and Technology, Uppsala University, is usually a summary of a number of papers. A few copies of the complete dissertation are kept at major Swedish research libraries, while the summary alone is distributed internationally through the series Digital Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology. (Prior to January, 2005, the series was published under the title “Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology”.)