universität wien

DISSERTATION

Titel der Dissertation

Variations on RNA folding -

Locally stable structures and RNA hybridization

angestrebter akademischer Grad

Doktor der Naturwissenschaften (Dr. rer.nat.)

Verfasser:	Stephan Bernhart
Matrikel-Nummer:	9503449
Dissertationsgebiet:	Chemie
Betreuer:	Univ.-Prof. Dr. Peter Schuster

Wien, am 28. Juni 2007

Abstract

In recent years, the importance of RNA molecules in living cells has been recognized, and consequently the interest in RNA related science has increased dramatically.
We introduce here new computational tools as well as modifications to established tools for thermodynamic RNA secondary structure prediction. These tools cover a variety of problems:

- Prediction of local RNA structures
- Prediction of RNA-RNA hybridization structures
- Extension of consensus structure prediction of RNA alignments
- Extension of RNA single structure prediction
- Prediction of local consensus structures of RNA alignments

The tools `RNAcofold` and `RNAplfold` cover the partition function and pair probability of the structure of RNA dimers as well as local partition functions and pair probability, respectively. `RNAcofold` can also predict equilibrium concentrations of dimers and monomers dependent on given initial concentrations of monomers. The existing algorithm `RNAfold` has been modified to include the partition function of canonical structures and a faster computation of interior-loop energy. The energy evaluation of the consensus structure prediction tool `RNAalifold` has been changed to a more realistic physical model. Furthermore, the possibility of sequence weighting has been added to `RNAalifold`, and a local version, `RNALalifold`, has been developed.

We show some applications of these new programs, which range from analysis of siRNA/mRNA interaction for `RNAcofold` and accessibility prediction of RNA interaction target sites for `RNAplfold`. We can, using other groups' experimental results, conclude that the reported lack of miRNA function when too many GU base pairs are present may be due to a change in the binding at

the 5' region of the miRNA. We also show that the new `RNAalifold` program outperforms the previous version at least when the alignments contain many gaps.

The programs developed are part of the freely available `ViennaRNA` software package, and can be downloaded from
`http://www.tbi.univie.ac.at/RNA/`

Zusammenfassung

In den vergangenen Jahren wurde die große Bedeutung der RNS für lebende Systeme erkannt, weshalb das Interesse an Forschung über RNS stark gestiegen ist.
In dieser Arbeit werden neue Algorithmen sowie Modifikationen von bekannten Algorithmen für die thermodynamische Sekundärstrukturvorhersage von RNS Molekülen vorgestellt.
Die neuen Programme `RNAcofold` und `RNAplfold` berechnen Zustandssummen und Basenpaarwahrscheinlichkeiten von RNS Dimeren beziehungsweise lokalen RNS Sekundärstrukturen. `RNAcofold` kann darüberhinaus verwendet werden, um die Gleichgewichtskonzentrationen von Dimeren und Monomeren als Funktion der Startkonzentration von Monomeren vorherzusagen.
Das bestehenden Programm `RNAfold` wurde um die Zustandssummenberechnung von kanonischen RNS-Strukturen erweitert, und eine schnellere Berechnung von Inneren Schleifen wurde integriert. Die Energieevaluierung des Programmes `RNAalifold`, das gemeinsame Strukturen von alignierten RNS Molekülen vorhersagt, wurde auf ein physikalisch betrachtet realistischeres Modell umgestellt. Darüberhinaus wurde die Möglichkeit, die einzelnen Sequenzen des Alignments für die Berechnung der gemeinsamen Struktur unterschiedlich zu gewichten, ergänzt.

Wir zeigen hier auch einige mögliche Anwendungen der erstellten Programme. Dazu gehört die Untersuchung der Interaktionen von mikro RNS mit Boten-RNS, die mit `RNAcofold` durchgeführt werden kann. Durch die Untersuchung von experimentellen Daten, die von verschiedenen Gruppen durch die Mutation von mikro RNS Bindungsstellen auf Boten-RNS erhalten wurden, können wir eine alternative Erklärung dafür geben, dass mehrere eingeführte G-U Basenpaare die Wirkung der mikro RNS beeinträchtigen. Wir können diesen Effekt einer Veränderung des Musters der zwischenmolekularen Bindung am 5' Ende der mikro RNS zuschreiben, das bekannterweise eine entscheidende Bedeutung für die Funktion der mikro RNS besitzt. Die

Vorhersage der Zugänglichkeit möglicher Bindungsstellen für RNS Interaktionen, für die das Programm `RNAplfold` herangezogen werden kann, ist sowohl für die Vorhersage möglicher mikro RNS Bindungsstellen als auch für die Evaluierung von Bindungsstellen für RNS Interferenz von großer Wichtigkeit. Wir zeigen auch, dass die Modifikationen des Programmes `RNAalifold` dazu führen, dass vor allem gemeinsame Strukturen von lückenhaft alignierten RNS Molekülen besser vorhergesagt werden können als zuvor.

Die entwickelten Programme sind Teil des frei zugänglichen `ViennaRNA` Softwarepakets, das auf
`http://www.tbi.univie.ac.at/RNA/`
heruntergeladen werden kann, sowie Teil der zu diesem Paket gehörigen Funktionsbibliothek, die Bioinformatikern weltweit die Verwendung der Algorithmen auch innerhalb eigener Programme ermöglicht.

Contents

1 Introduction

1.1 About this work

RNA, as an integral part of the living cell, has been living in the shadow of its well known sister molecule DNA for almost half a century. Only recently did most life-scientists realize that what they have called junk-DNA – that is non protein coding DNA – can indeed have a multitude of functions via RNA products. Today, RNA is accepted as a central part of many important cellular functions, and some scientists, like John Mattick, believe that it was the ability to use RNA for diverse regulatory functions that made complex modern life (i.e. multi-cellular organisms) possible.

This work is concerned with the *in silico* prediction of RNA structures, for local substructures as well as for intermolecular structures. With the enormous amount of biological data already accessible and the even bigger amounts of data which will be generated in the near future, fast computational methods for all areas of molecular biology are needed. Computational methods for RNA structure prediction is what the theoretical biochemistry group at the University of Vienna has been concerned with for many years. This thesis summarizes algorithmic developments generated in recent years.

This work mostly is a compilation of several papers that were published in recent years [7–9, 19, 92] as well as unpublished results and modifications of algorithms we recently introduced. This thesis focuses on the development of algorithms rather than on applications. However, there are some very promising, mostly unpublished, results which have been generated by our group.

The first chapters introduce RNA secondary structure prediction and motivate its necessity. In chapter 3, some additions to the existing `RNAfold` program are described. Chapter 4 is concerned with the prediction of joint

structures of two RNA molecules. Chapter 5 describes a program for local partition function computation, and the final chapter 6 is concerned with additions to the `RNAalifold` program for consensus structure prediction.
As the programs described in this work are diverse and have a variety of different applications, applications and results derived from using these algorithms are part of the respective chapters.

The algorithms and programs developed for this work have been implemented in ANSI C and are part of the `ViennaRNA` software package, which can be downloaded at
`http://www.tbi.univie.ac.at/RNA/`
While the programs described here have been thoroughly tested and contain no mistakes we are aware of, there is still room for improvement in all of them. All the programming work done for this thesis is strongly interconnected with the `ViennaRNA` package, and the corresponding software library, and the new programs are as easily maintainable as the established programs like `RNAfold`.

1.2 RNA

1.2.1 History of RNA

Nucleic acids were first described by Miescher in 1869 [87]. Allegedly, he already suspected a connection to heredity – but it may well be that anybody who discovered something new in the cell did so in those times. In 1909, Phoebus Levene found that some nucleic acids contain ribose (in 1929 he identified the sugar in the others as 2' deoxyribose) [18]. Of the five nucleobases, Guanine was described first, in 1844, when it was isolated from guano.

Soon after the description of the DNA double helix in 1953 [130], RNAs were suspected to play a role in information transfer from DNA to proteins [109]. The first hints in that direction came even earlier, when in the early forties it had been realized that the cellular sites of protein synthesis are rich in RNA [11, 16]. This was further confirmed by the discovery of tRNA (then called sRNA) [48, 49] and the deciphering of the genetic code [94]. These facts lead to the central dogma of molecular biology formulated by Francis Crick [20, 21]. The central dogma is a set of rules about the direction that information can flow in a cell. It basically states that there can not be an information transfer originating from protein. The central dogma is not concerned with the role of RNA (or DNA or proteins, for that matter) in the cell. However, interpretations of the scientific community at large changed the meaning of the central dogma, and RNA was reduced to a carrier of information.
The discovery of catalytic activity of RNA in modern cells, namely *E. coli* RNAse P by Sidney Altman 1983 [14] and a self-splicing intron in tetrahymena by Thomas Cech 1982 [66], won them a Nobel prize in 1989, but was not entirely sufficient to remove the "carrier of information" label from RNA. In the 1990ies, however, RNA interference in plants [93] , fungi [112] and animals [34] as well as the miRNA pathways in animals [71] and plants [108] were discovered. Together with the finding that the Ribosome is essentially a

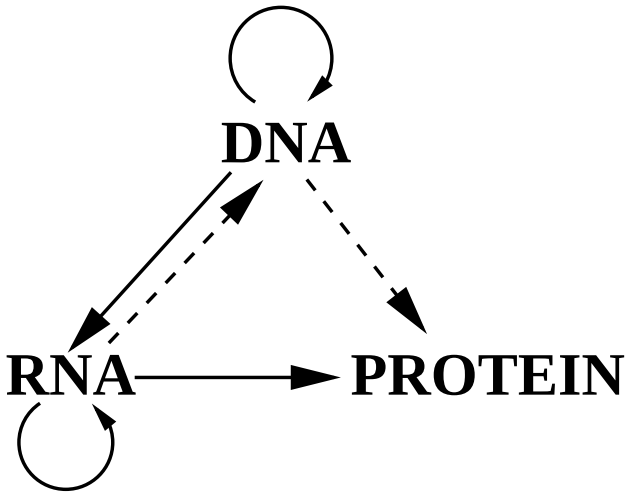


Figure 1: The central dogma as depicted by Crick in 1970 [21]. Solid arrows show general, dashed arrows special information transfers. Absent arrows are impossible according to the central dogma.

Ribozyme [5,95] (interestingly, in his Nobel lecture 1962, James Watson supposed that the ribosomal proteins are primarily structural proteins needed to get rRNA and reaction intermediates in the right positions [131]) this helped to shift the perception of RNA. Today, RNA is in the focus of attention of many scientists, be it as regulatory molecule, as catalyst or as very versatile silencing tool.

1.2.2 RNA chemistry

RNA, ribonucleic acid, Fig.2, is a hetero-polymer consisting of a backbone of ribose in D-β furanose ring structure, phosphate groups and nucleobases. In the RNA polymer, the ribose is linked by phospho-diester bonds between the 5' and the 3' carbon. Every phospho-diester group carries a negative charge, so counter ions are important and usually present in aqueous solutions of RNA (and DNA). Because of the 5' to 3' linkage of the sugars, RNA and DNA are directed polymers. By definition, the sequence of nucleobases, the variable part of the RNA molecule, is written from the 5' to the 3' end.

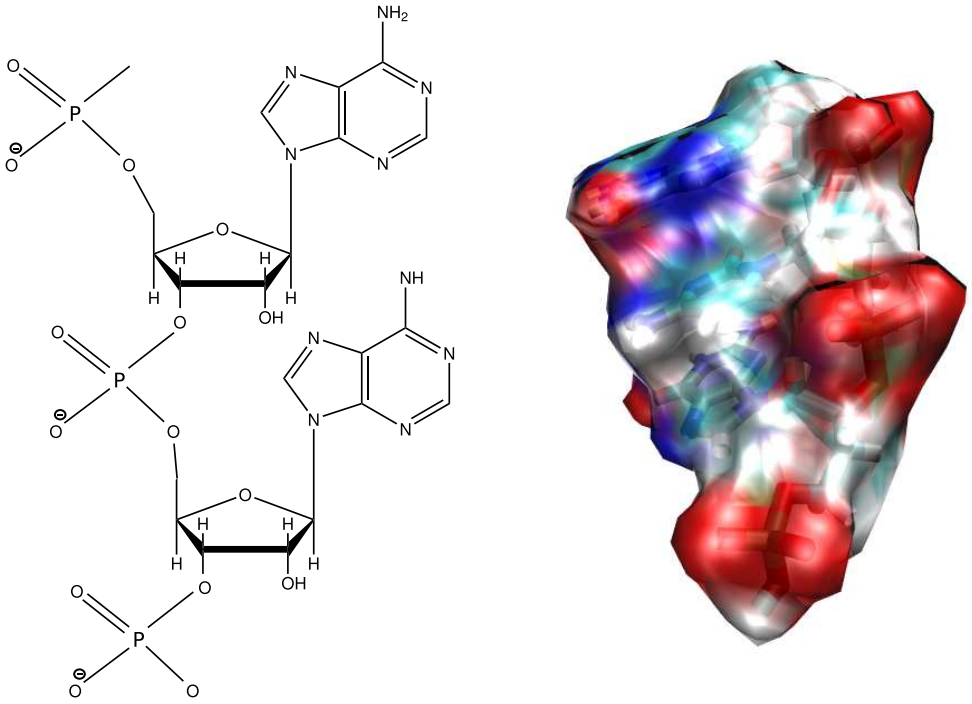


Figure 2: Structure of RNA.
Left: structural formula, 2 Adenine nucleotides. Replacement of the two OH groups would make this RNA a DNA.
Right: Space occupancy of 3 nucleotides

On the 1' Oxygen one of the four nucleobases, the purines Adenine and Guanine or the pyrimidines Cytosine and Uracil (3) is linked to the ribose predominantly in anti conformation.

The differences to the more well known DNA (deoxy ribonucleic acid) molecule are quite small. The first difference is the usage of Uracil instead of Thymine, the second one is that DNA lacks the 2' OH group of the ribose. These differences have, however, some distinct physicochemical effects: first of all, RNA is more catalytically active, due to the existence of the free OH group on the Ribose ring. Secondly, the preferred form of RNA dimers is the A-helix (see Fig. 4), as opposed to DNA, which mostly takes the B-helix, dependent on its sequence and on environmental conditions [6]. These chemical differences lead to different functions of RNA and DNA in biological systems. Because of these functions, RNA molecules contain fewer monomers

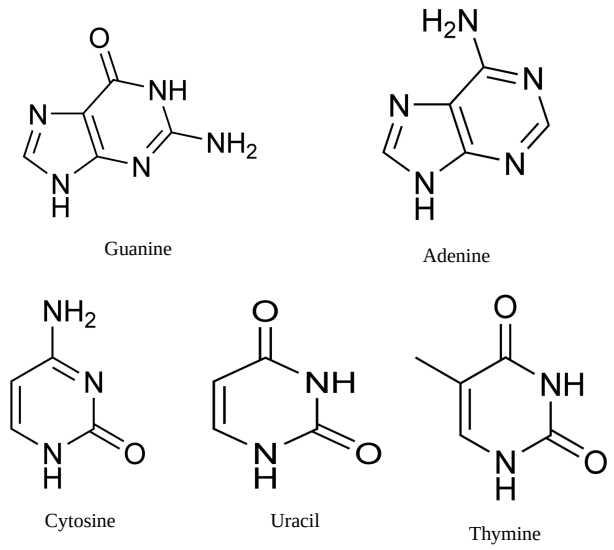


Figure 3: The five nucleobases. Top: purines, Bottom: pyrimidines

by several orders of magnitude, i.e. the average RNA is much shorter than the average DNA. Furthermore, RNA usually acts as a monomer, as opposed to the famous DNA dimers building the DNA double helix. RNA dimers are used rarely, mostly as "substitute" for genomic DNA in certain RNA viruses. The RNA monomers, lacking a complementary counterpart, are prone to fold back on themselves, forming structures that are fundamental for their function in living systems.

1.2.3 RNA biology

As RNA has long been regarded as a rather uninteresting molecule, we probably do not know all functions that are mediated via RNA molecules in life. However, many vital cellular functions are either catalyzed or controlled by RNA molecules. The most prominent part of these functions belong to transcription. There, RNA has catalytic as well as regulatory functions. For the catalytic part, the Ribosome itself is a ribozyme, and of course there are tRNAs and mRNAs. These three core elements of protein synthesis can be found in all domains of life, and their maturation is also mediated mostly

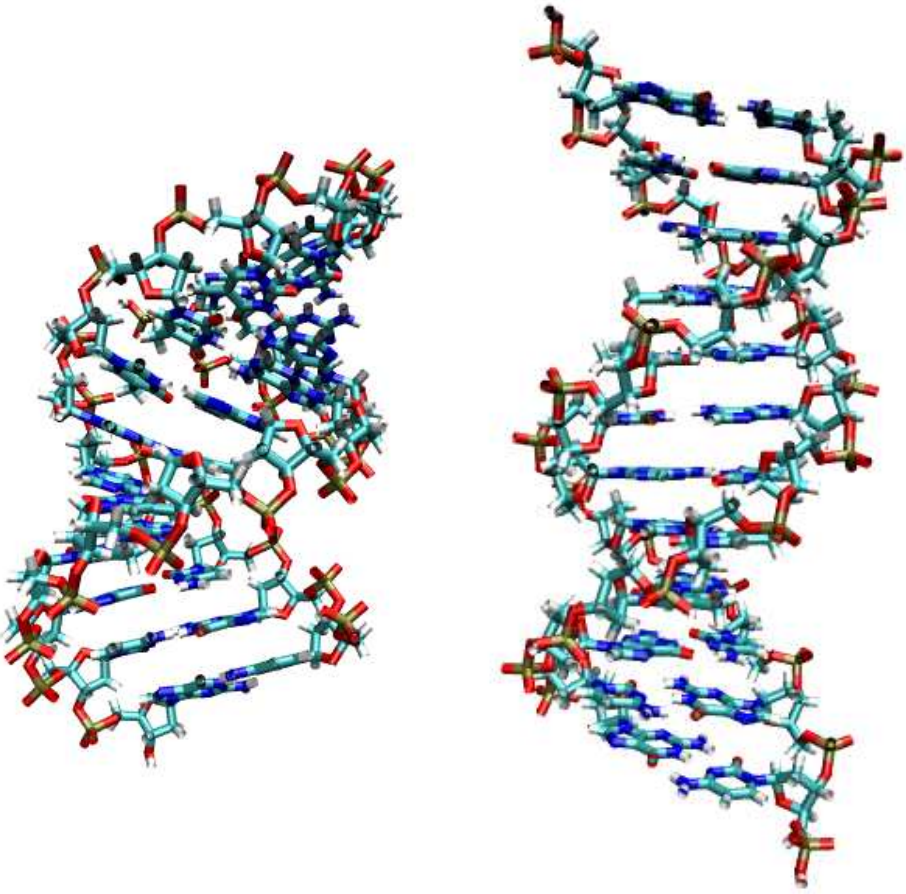


Figure 4: The two main forms of nucleic acid helices. To the left the A type helix, which is mostly adopted by RNA, as well as by DNA in environments devoid of water, e. g, in most crystals. The right picture is the B-type helix, which is the natural conformation for DNA in the cell. In both pictures, twelve base pairs are shown. It is easy to see that the B-form is more elongated, with better access to the bases possible from the outside.

with aid of RNA molecules: RNAse P, which cleaves the 5' part of pre tRNA has an important RNA part. The editing of rRNA, tRNA and mRNA bases is guided by so called guide RNAs. For the rRNAs, snoRNAs, namely H/ACA and C/D Box, are used for pseudo-uridinylation and 2'O-methylation of ribose, respectively. Recently, snoRNAs targeting spliceosomal RNAs and tRNAs have been found [123, 126].

In mRNA maturation, important parts of the major-spliceosome, which splices at the "canonical" splice site GT-AG, are RNAs: U1, U2, U4, U5, U6. The same is true for the minor-spliceosome, splicing at other sites [101], where RNAs U11, U12, U4atac, U5 and U6atac take part. 7SK RNA has recently also been identified as a part of the splicing apparatus in vertebrates [67].

The 3' end of replication dependent histone pre-mRNAs is not poly-adenylated, but cleaved by a ribonucleoprotein complex consisting of U7 RNA and protein components [117] . In addition, RNA is also used for transcriptional control – the miRNA pathway is a very versatile tool for transcriptional attenuation.

There are also functional motifs present in mRNAs, e.g. the iron response element IRE [47], the internal Ribosome entry site IRES [59] and the selenocystein insertion sequence SECIS [64]. Finally, the signal recognition particle, SRP, which guides proteins to the ER during transcription, also has an important RNA component [60].

Y-RNAs, which are part of the Ro ribonucleoprotein complex, whose function is yet to be determined, but who are allegedly involved in quality control of 5S rRNA [68]. There are also a number of prokaryotic RNAs known (around 60 for *E. coli*), of which at least a third are known or supposed to be repressors or activators of protein translation [40]. Another example of functional prokaryotic RNAs tmRNA, which is used for rescuing stalled ribosomes [45].

The number of identified RNAs which are allegedly not really involved in protein synthesis is small by comparison. An example could be vault RNAs,

which are part of the vertebrate vault ribonucleoparticles [91]. Better known is the function of e.g. telomerase RNAs, which are a part of the telomerase ribonucleoparticle which caps chromosomes [33], and RNAse MRP [90], which cleaves primers for DNA replication in eukarya.
An extensive review of known non-coding RNA and their function is e.g. given in [10].

1.2.4 RNA bioinformatics

Due to the fact that predicting RNA secondary structures is relatively easy, the first algorithms for doing so date back into the late 1970ies. In 1975, Pipas and McMahon [104] published a set of programs to predict RNA secondary structure using thermodynamic data, which however was inefficient, taking $\mathcal{O}(2^n)$ in processor time. The first efficient $\mathcal{O}(n^3)$ algorithm was developed by Nussinov in 1978 [97], and improved in 1980 [96]. In 1981, Zuker and Stiegele [141] developed the dynamic programming algorithm which is in principle used in this work. The computation of the partition function was first published by John McCaskill in 1990 [84]. In 1994, the `ViennaRNA` Package was created [52].

A new field of RNA bioinformatics is the identification of novel RNA genes. Many non-coding RNA molecules found to be transcribed via high throughput screens have no known function. As a consequence, there is a great need for computational tools for identification and classification of RNA genes. Identification of RNAs by sequence homology, essentially done by blasting, is a widely distributed ansatz, but it obviously only works for known RNA families. Additionally, this approach neglects the importance of RNA structure. Successful programs that take structure into account are e.g. tRNAscan-SE by Lowe and Eddy [75] or the ERPIN web server [69].
Recently, algorithms to identify unknown RNA genes out of genomic data have been published by Washietl [129] and Pedersen [102]. These programs have already been successfully applied to search for non-coding RNAs in ver-

tebrates [128], nematodes [89] and urochordata [88].

In this work, we provide enhanced versions of basic tools from which many of these methods can profit. The RNA gene finding done by the `RNAz` program [128], for example, is based on the consensus structure prediction program `RNAalifold`, of which we created an improved version as well as a local version.

2 RNA structure

An RNA molecule can be characterized by the sequence of its bases. This, also called the primary structure, is sufficient to distinguish RNAs. If the sequence of the nucleotides is the same, so are two or more molecules. However, this information alone is not sufficient. As Neil Campbell wrote in his textbook Biology [15]:

> *How a device works is correlated with its structure: Form fits function. Applied to Biology, this theme is a guide to the anatomy of life at its many structural levels, from molecule to organisms. Analyzing a biological structure gives us clues about what it does and how it works. Conversely, knowing the function of a structure provides insight about its construction.*

Accordingly, if we are interested in the function of RNA molecules within living organisms, their structure is a good starting point. It must be said that as it is comparatively easy to sequence RNA (or DNA) molecules, there is a lot of RNA sequence data available. On the other hand, determining the tertiary structure with X-ray crystallography or NMR is much harder, and therefore there will always be more known sequences than structures [72]. We are settling here for a kind of compromise by working with the secondary structure of RNA, which is easier to obtain than 3D structure but more informative than sequence information alone.

2.1 RNA secondary structure

An RNA secondary structure is defined as a list of either Watson-Crick (G-C, C-G, A-U, U-A) or wobble (G-U, U-G) base pairs with the following properties: (for convenience, we use the following nomenclature: we will write (i, j) to signify a base pair between positions i and j, and we will always assume $i < j$ if not stated otherwise.)

GCGAGCCUAGCGAAGUCAUAAGCUAGGGCAGUCUUUAGAGGCUGACGGCAGGAAAAAAGCCUACGUCUUCGGAUAUGGCUGAGUAUCCUUGAAAGUGCCACAGUGACGAAGUCUCACUAGAAAUGGUGAGAGUGGAACGCGGUAAACCCCUCGCA

Figure 5: Different Levels of RNA structure description. At the top, the nucleotide sequence of the S Domain of RNAse P of *B. subtilis* is shown. To the left, its secondary structure is plotted
To the right the 3D structure is shown, where the surface of the molecule is transparent, and the bonds within the molecule are shown in stick representation.
Both structures have been determined by NMR [63].

- The base pair has to be either a wobble or Watson-Crick base pair
- Any base can take part in at most one base pair.
- If bases i, j form a base pair, there have to be at least 3 bases between i and j
- Base pairs do not cross. (Knot-free condition)

Note that the knot-free condition also assures that there are only anti-parallel stacks.
In a more formal definition, a secondary structure S is a set of base pairs on sequence s. With x_i the base on sequence position i, S satisfies the following rules:

i $(i,j) \in S \Rightarrow (x_i x_j) \in \{AU, UA, GC, CG, GU, UG\}$

ii if $((i,j) \wedge (k,l)) \in S \wedge i = k \Rightarrow j = l$

iii if $(i,j) \in S \Rightarrow i < j - 3$

iv if $((i,j) \wedge (k,l)) \in S \wedge i < k \Rightarrow i < k < l < j \vee i < j < k < l$

It must be emphasized that violating these conditions is not physically impossible. There are all possible kinds of non-standard base pairs in nature [39,73], as well as pseudo-knots violating the knot-free condition and bases taking part in more than one base pair [38, 86, 106]. However, somewhat arbitrarily these things are not defined as secondary, but as tertiary structure. There are several reasons for doing this:
The 6 base pairs included in the definition of secondary structure are the only ones which mostly give energetically favored pi-system interactions. Most of the free energy of an RNA molecule is contributed by these base pairs. This is mostly due to the fact that they are almost exactly isosteric, which means that at least the Watson-Crick base pairs can be exchanged without distorting the three dimensional structure of the helix in any way. For GU base pairs, the distortions are only minor [79].

Furthermore, there is a kinetic hierarchy within the formation of the RNA structure: usually, the secondary structure forms, and when these helices are built, they interact to form the tertiary structure [13, 125].
These two facts show that the building of a tertiary structure is strongly influenced by the secondary structure. As a consequence, changes in secondary structure as a result of the formation of a tertiary structure are rare. However, tertiary structures can show substantial similarities even though the secondary structures are different [62, 132]. This shows that while secondary structure prediction is a powerful tool, it cannot entirely replace tertiary structure prediction or experimental structure determination.
Pseudo-knots do occur frequently in biologically important RNA molecules like tmRNA, RNAse P, telomerase and others. About 20% of the known biologically relevant RNA structures contain pseudo-knots. Not including them thus limits the accuracy that can possibly be achieved by secondary structure prediction algorithms.

The other class of reasons are for convenience of the computation. It has been shown that not using the knot-free condition will render the problem NP-complete [76] if the more sophisticated energy evaluations are used. However, several special classes of pseudo-knots can be computed in polynomial time, with the fastest (and most restricted) algorithms running in $\mathcal{O}(n^4)$ [107]. This is still quite slow and makes computation infeasible for long RNA molecules. Another inconvenient fact for dealing with pseudo-knots is that the energy parameters for the Turner energy model (section 2.3) are not known and are hard to quantify experimentally. The entropic corrections used by Isambert and Siggia [57], on the other hand, are dependent on the length of the helices and therefore not suited for dynamic programming.

2.2 Visualization of RNA secondary structures

There are several approaches to the visualization of RNA secondary structures. A widely used approach is to draw the molecule. The created drawings are very informative as long as the molecules do not get to large. When comparing RNA molecules, it is easy for humans to find common patterns or differences by visual inspection. However, for long RNA molecules, the picture contains too much information, and different solutions are to be preferred. One such solution is the so called mountain-plot. Here, the sequence is drawn at the abscissa. When a base is paired upstream (i.e. it is i in a (i, j) pair), the value on the ordinate is increased by one (compared to $i - 1$). For j, it is decreased by one. Still another representation draws the sequence along a circle and then connects base pairs by secants. More formal representations also exist: one is the so-called dot-plot, where in a matrix a dot is drawn at point i, j if a base pair exists in the structure. This can be expanded to show the mfe structure and the base pair probabilities at the same time (as i, j and j, i correspond to the same base pair).
A very reduced version is the so-called "Vienna" or bracket dot representation. In this, unpaired bases get a ".", bases paired upstream a "(" and downstream a ")". Because of the knot-free condition, the decomposition using standard mathematical rules will be unique and paired brackets will always correspond to a base pair. (See Fig. 6).

Yet another possibility is to represent RNA secondary structures as ordered trees (i.e. the children of a node cannot be exchanged). There are different levels of abstraction that can be used to draw these trees. In the most detailed case, every loop is represented by a node, and the bases are children (or labels) of the respective node. In other, more abstract tree representations, the stacks are contracted into one node, stacks and bulges are contracted, and so on, until only the branching topology is represented (Fig. 7).

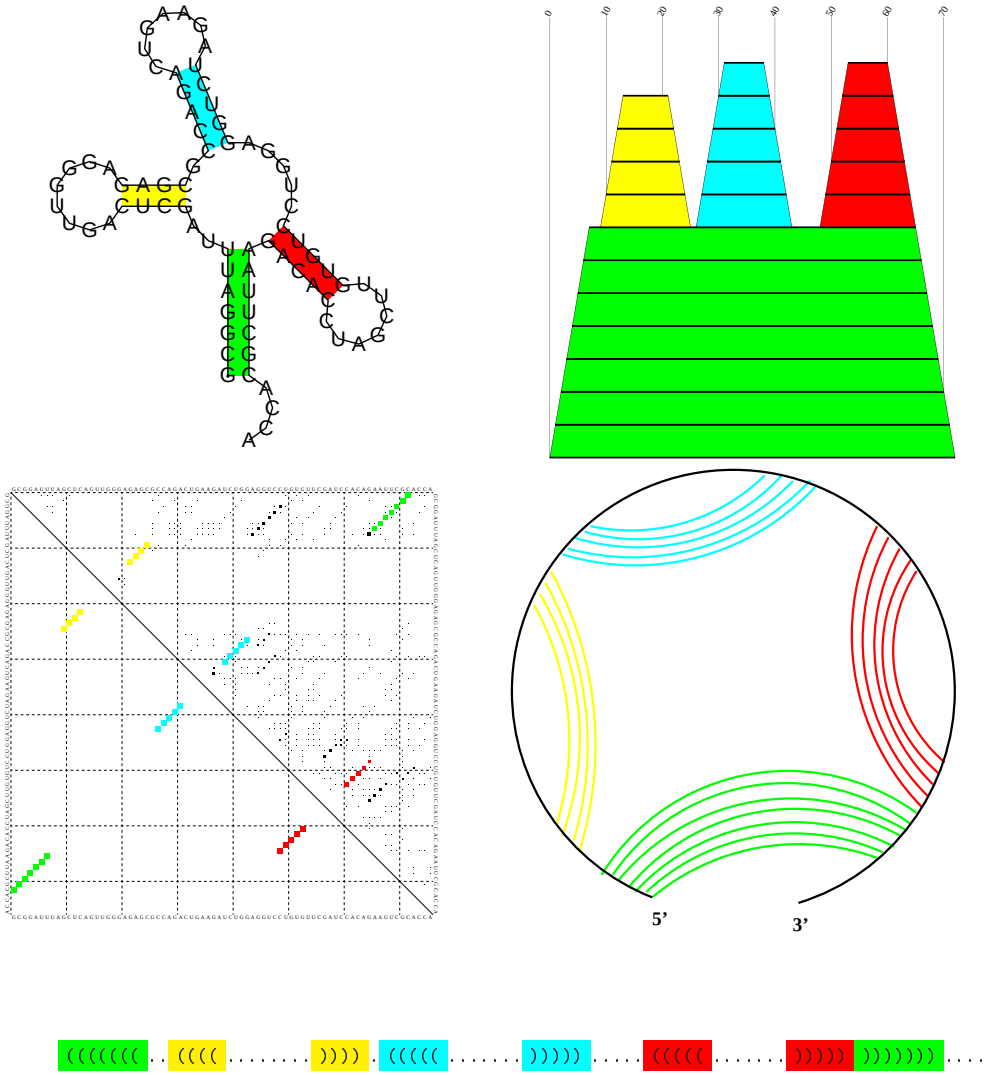


Figure 6: Representations of RNA secondary structure. A yeast (*Saccharomyces cerevisiae*) tRNA-Phe in different representations. The stems are colored to make comparison easier.

Top: left: graph representation of secondary structure. right:Mountain plot.

Middle: left: dot plot right: circular graph representation.

Bottom: Dot bracket representation.

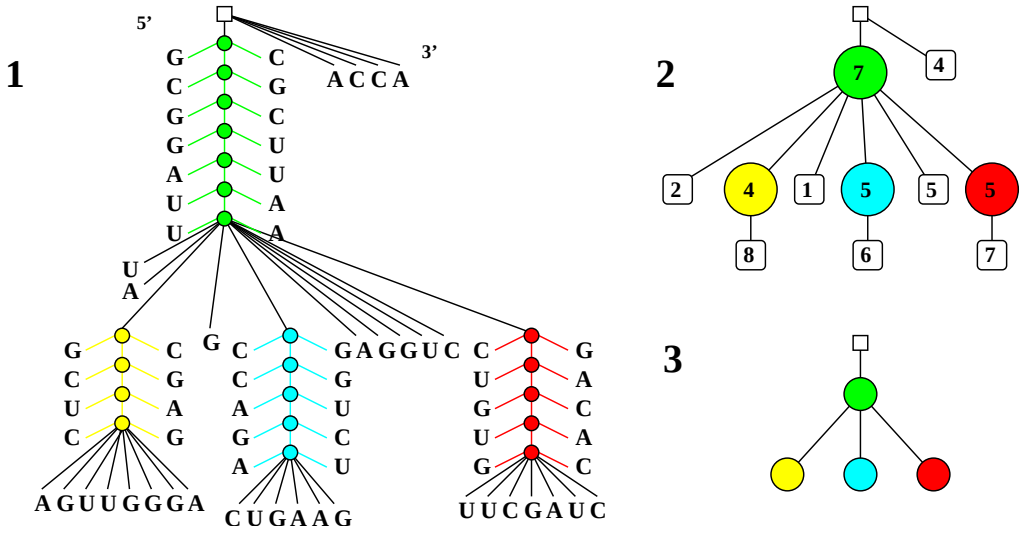


Figure 7: Tree representations of RNA secondary structure. (the same tRNA molecule and coloring as in Fig. 6) is used. **1** shows a representation containing all secondary structure information. **2** Is an abstraction where the nodes, representing stacks (cycles) or unpaired bases are labeled with the number of base pairs or free bases, resp. **3** is the most abstract representation, where only the most basic topology can be seen.

2.3 The free energy of RNA secondary structures

An important assumption for computing the free energy of RNA secondary structures is additivity. The structure can be decomposed into building blocks, called loops. The assumption now is that the free energies of these loops are additive. This means that the contributions of the different loops are not dependent on other parts of the molecule. This enables us to use e.g. dynamic programming approaches for free energy prediction. Naturally, this simplification may lead to problems when trying to predict biologically relevant RNA structures. However, it is a prerequisite for the usage of dynamic programming or SCFG algorithms (see below). As molecules adopting the same secondary structure can have many different atomic configurations (e.g. backbone atoms), the energy of an RNA secondary structure will always contain entropic contributions. It is therefore always a free energy. However, we will use the term "energy" here in the same sense, as it is cumbersome to always use the correct term.

2.3.1 Loop decomposition

The maybe easiest way to describe the loop decomposition of RNA secondary structure is considering the secondary structure as an outer planar graph. When doing this, the set of bases correspond to the vertices of the graph. The phospho-diester bonds are in the set of edges E. They link base (vertex) i with base $i+1$ (we will call them "outer edges"). Base pairs are also in E, they connect vertices while fulfilling the secondary structure rules of 2.1, we will call them "inner edges". These rules lead to the following properties of the graph:

- If $((i,j) \in E \Rightarrow i = j-1 \vee (x_i, x_j) \in AU, UA, GC, CG, GU, UG$
- The maximal degree of a vertex is 3, all vertices except 1 and n have at least degree 2.
- If $((i,j) \in E \Rightarrow i = j-1 \vee i < j-3$
- Edges do not cross, i.e. the graph is outer planar.

The loops are now the faces of the graph, i.e. the areas that are enclosed by edges. These faces are differentiated by the number of "inner edges" (base pairs) that enclose them. A face containing one base pair is called a hairpin-loop. Loops containing two base pairs are interior-loops. These fall into three different subclasses, depending on the number and distribution of the outer edges. Interior-loops with 4 edges (two outer and two inner edges) are called stacks, interior-loops where the shortest path between two base pairs contains one outer edge (and that are not stacks) are called bulges, and all others are actual interior-loops. Finally, faces containing more than two base pairs are called multi-loops or branch loops (see Fig. 8). Sometimes, the outer area of the graph, which is not surrounded by edges, is called the exterior-loop. While this is contradictory to the definition of loops given

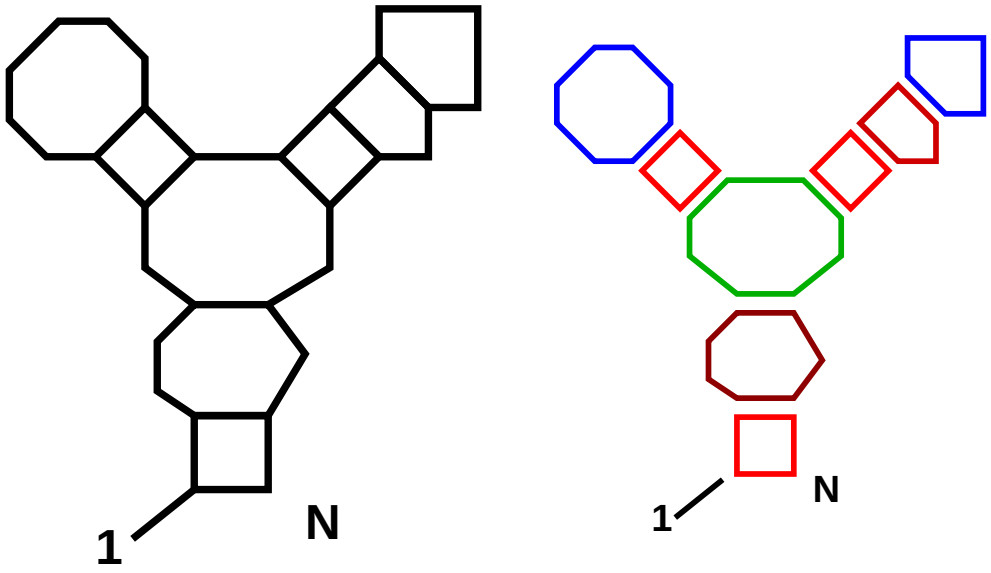


Figure 8: An RNA secondary structure (left) and its loop decomposition (right). The exterior-loop is shown black, the hairpin-loops blue, the interior-loops in shades of red (stacks: bright red, bulge darker, interior-loop almost brown) and the multi-loop in green.

just in this paragraph, it enables us to decompose the whole RNA secondary structure into loops.

2.4 Free energy of loops

The contribution of a single Loop to the free energy of a RNA secondary structure can either be experimentally derived or computed from experimental data.

To get to the energy values experimentally, Turner and others use melting experiments [81]. In these experiments, the "melting" (i. e. the phase transition between single and double stranded RNA) of small RNA molecules is monitored against the Temperature. The Gibb's free energy can then be computed out of Absorbance/Temperature curves using the following formulas:

$$1 - f = \frac{A(T) - A_d(T)}{A_s(T) - A_d(T)}$$

where f is the fraction of molecules in double stranded state, $A(T)$ is the Absorbance at temperature T and $A_s(T)$ and $A_d(T)$ are the absorbances for single strand (upper baseline) and double strand (lower baseline) molecules resp. at temperature T. with C_T the total concentration of all single strands and the equilibrium constant

$$K = \frac{2f}{(1-f)^2 C_T} = \exp(\frac{-\Delta G^\circ}{RT}) = \exp(\frac{-\Delta H^\circ}{RT} + \frac{\Delta S^\circ}{R})$$

the Enthalpy $-\Delta H^\circ$ and the Entropy ΔS° can be computed for different molecules.

For dimers, ΔH_N, the enthalpy of the formation of a bi-molecular complex, can be computed using [127]:

$$\frac{1}{T_m} = \frac{1}{T'_m} + \frac{R}{\Delta H_N} \log \frac{c}{c'}$$

where T_m is the melting temperature at concentration c and T'_m at c'. By variation of the sequence and comparison of their energies, energy values can be assigned to different loops. The reference point of the energy is always the random coil of the open chain (i. e. a molecule without base pairs). However, obtaining energies of bigger loops is difficult. This has two reasons: Firstly, the combinatorial explosion one gets because the number of different sequences is 4^n. Secondly it must be ensured that no other base pair patterns are possible for the molecule, or at least that these are negligible, because otherwise the reference point (base line) will be shifted and the phase transitions are blurred. This makes the measurement of multi-loop energies very hard for this technique. However, Mathews et al. did measure a few 3 way and 4 way branched loops [83].

Even with the small structural motifs used, the error is about 10% [81, 138], where stacking parameters are less prone to errors than the different other loop parameters.

A different, recent, approach to measure the contributions of different loop types is using single molecule techniques [139]. While using melting curves

is of course an ensemble approach, these techniques can monitor structural changes at a single molecule level. Furthermore, they can use optical tweezers to actively pull at molecules, measuring the force necessary to break base pairs or change structural motives. While these techniques are in principle suitable to refine the energy parameters, up until now this has not happened.

Whatever technique is used for measuring the energy parameters, remembering the big dependency on counter ions is crucially important. As mentioned above, every phospho-diester bond carries a negative charge. In biological environments, there is a multitude of different combinations of ions, and divalent cations, especially the abundant Mg^{2+}, are known to change structural preferences of RNA molecules. Usually, measurements are taken in 1 M NaCl solution, which is buffered to pH 7.
Computation of loop energies uses the polymer ring closure works of Jacobson and Stockmayer [58]. These state that the entropy loss (which is the only contribution used) rises logarithmically with the length of the loop. The energy of a hairpin i, j is taken to be dependent only on the base pair i, j, the so-called mismatched bases $i+1$ and $j-1$ and on the length of the unpaired stretch $l_u^h = j - i - 1$. The energy of a long interior-loop is dependent on the type of the closing base pairs i, j and k, l, on the mismatches $i+1, j-1, k-1$ and $l+1$, on the total length of the interior-loop $k + j - i - l - 2$ and on the asymmetry of the loop $|(k - i) - (j - l)|$. Finally, in multi-loops the energy is dependent on the number of stacks and the number of unpaired bases of the multi-loop. The number of unpaired bases, however contributes linearly to the energy, in contrast to the logarithmic rise computed by Stockmayer and Jacobson. This is necessary to compute the multi-loop contributions in $\mathcal{O}(n^3)$ within dynamic programming algorithms.

2.5 Predicting RNA secondary structures

2.5.1 Dynamic programming

Dynamic Programming is the reduction of a big problem to small subproblems, all of which are further reduced up to some sort of smallest possible problem, which is then solved and solutions of the bigger problems are built up out of solutions to smaller problems. In the Dynamic Programming Method of RNA folding, which is used in the `ViennaRNA` Package and thus in all the programs developed in the course of this work, an RNA secondary structure is decomposed like this: A stretch X of sequence s and length l is built of a stretch X_1 of length $l-1$ and a base i. The base i can be

- unpaired, so the minimum free energy $E_X = E_{X_1}$
- or paired with a base between $1..l-1$.

The dynamic programming approach uses the definitions of secondary structure as well as the assumption that the energies of parts of the molecule are strictly additive and not context sensitive. For the maximum matching "Nussinov" algorithm, which will shortly be explained, this is clearly the case. It also is true for the more complicated "stacking" energy models. However, this assumptions fails when looking at the actual data. The energy of say an interior loop does change slightly depending on the surrounding loops. This is probably one of the main reasons that even the best algorithms are not perfect, see section 2.6 for details. The Nussinov algorithm [98] basically maximizes the number of base pairs an RNA molecule can achieve, or the number of hydrogen bonds between the bases. To maximize the number of hydrogen bonds, a scoring is introduced where GC or CG pairs get score 3, AU,UA score 2 and GU,UG score 1. The minimum free energy recursions then look like this:

$$N_{i,j} = \max \begin{cases} N_{i-1,j} \\ \max_{i<k\leq j}\{N_{i+1,k-1} + N_{k+1,j} + \delta(i,k)\} \end{cases} \quad (1)$$

with the score $\delta(i,k)$ the score of the base pair i,k. For a sequence of length l, the minimum free Energy of the Nussinov model is simply $N_{1,l}$. For ease of understanding, we use $N(i,i) = N(i+1,i) = 0$. As can be easily seen, this computation needs $\mathcal{O}(n^3)$ processor time and $\mathcal{O}(n^2)$ memory.
If the loop based energy model is used, the computations have to differentiate between the different loop types. However, the energy gain of the secondary structure is due to the stacking of the π-electron-systems of the bases. To model this, the so-called Turner or nearest neighbor energy model is used. It takes into account the stacking of base pairs – and single bases on base pairs, see chapter 2.5.2.
If base pairs stack directly onto each other, the energy is stabilizing. All possible WC and wobble base pair stack interaction energies have been tabulated. If we have a bulge-loop, the free energy is dependent on the length of the bulge as well as on the bases of the bulge lying next to the closing base pairs – the so called mismatches. Usually the energy of a bulge-loop is destabilizing. Only in bulges of size one, i.e. if there is only one base between the base pairs, the stacking energy is used together with a penalty, so they can be stabilizing, also.
For interior-loops, the energy is modeled as being dependent on the size of the loop, i.e. how many bases are there between the two base pairs, the asymmetry of the loop and the bases adjacent to the closing base pairs, the mismatches. In the case of a hairpin-loop, the energy is either tabulated, as is the case for most Tetraloops, i.e. loops with exactly 4 bases between the closing base pairs, or it is modeled as dependent on the size of the loop and the mismatches. For a multi-loop, a linear approximation is used, where the energy is split in contributions of the unpaired bases in the loop, the number of stacks in the loop, and a penalty for closing the multi-loop. All these contributions are destabilizing.

In course of the recursions for the computation of the minimum free energy, we will fill and use the following arrays:

$F_{i,j}$ The minimum free energy on a stretch between i and j

$B_{i,j}$ The minimum free energy subject to the constraint that i and j form a base pair (i, j)

$M_{i,j}$ A multi-loop contribution where there is at least one stack somewhere between i and j

$M^1_{i,j}$ A multi-loop contribution containing exactly one stack and base i is paired.

Using these arrays, the recursion for the minimum free energy looks like this:

$$\begin{aligned} F_{i,j} &= \min\left\{F_{i+1,j}, \min_{i<k\leq j} B_{i,k} + F_{k+1,j}\right\} \\ B_{i,j} &= \min\left\{\mathcal{H}(i,j), \min_{i<p<q<j} \mathcal{I}(ij,pq)B_{p,q}, \min_{i<k<j} \alpha + \beta + M_{i,k} + M^1_{k+1,j-1}\right\} \\ M_{i,j} &= \min\left\{M_{i+1,j} + \gamma, \min_{k} B_{i,k} + M_{k+1,j} + \beta(i,j), M^1_{i,j}\right\} \\ M^1_{i,j} &= \min\left\{M^1_{i,j-1} + \gamma, B(i,j) + \beta(i,j)\right\} \end{aligned} \tag{2}$$

With $\alpha, \beta(i,j), \gamma$ the parameters for closing a multi-loop, extending it by one base pair and by one unpaired base, resp. $\mathcal{H}(i,j)$ is the energy of a hairpin-loop between sequence positions i and j, and $\mathcal{I}(ij,pq)$ the energy of an interior-loop between the base pairs i,j and p,q. Remember that the values of both $\mathcal{H}$ and $\mathcal{I}$ are dependent on the unpaired bases inside the loops as well as on the closing base pairs. Again, we use $F(i+1,i) = F(i,i) = 0$ as a kind of initialization to keep the formulas simple.

The seemingly complex multi-loop computation is done to ensure the unambiguousness of the decomposition, which means that there is only one way to decompose every structure. This is of no import to the minimum free energy computation, but is vital for the computation of the partition function we will attempt shortly. The only problem of this approach lies in the computation of $B_{i,j}$, namely in the interior-loop part thereof. The computation as written down here is obviously $\mathcal{O}(n^4)$. This drawback is circumvented by forbidding

interior-loops with a length longer than a constant X. The recursion then looks like this:

$$B_{i,j} = \min \begin{cases} \mathcal{H}(i,j) \\ \min_{i<p<q<j;p+j-i-q-1<\mathrm{X}} \mathcal{I}(ij,pq) B_{p,q} \\ \min_{i<k<j} \alpha + \beta + M_{i,k} + M^1_{k+1,j-1} \end{cases}$$

There are, however algorithms that compute full length interior-loops in $\mathcal{O}(n^3)$. See section 3 for a description.

2.5.2 Dangling ends

While these four loop types are all that is needed to decompose any secondary structure following the definition in 2.1, the success of such an energy model is fairly poor. In the early 1970ies [78, 127], it was realized that unpaired bases adjacent to base pairs play a role in stabilizing secondary structures. Recently, this effect was attributed to the exclusion of water from the hydrogen bonds of the closing pairs of a stack [56]. Alternatively, it can also be due to a stacking of single bases onto a base pair and an interaction of the π-electron-systems of the bases.
Whatever the physical principles behind the stabilizing effect, using this so-called **Dangling Bases** with their measured energy contributions increases the predictive power of the loop based energy model. The energy contributions are dependent on the type of base dangling and on what base(pair) it dangles onto. It has to be noted that in general, stacking onto the 3' end of a base pair is more stabilizing than on the 5' end for RNA molecules (average energy contribution of a 3' dangle=$\overline{\Delta\Delta G}^o_{3'} = -0.82$ kcal/mol and $\overline{\Delta\Delta G}^o_{5'} = -0.22$ kcal/mol for the 5' dangle). For DNA molecules, the directional difference is less pronounced, and the 5' dangle is the more stabilizing ($\overline{\Delta\Delta G}^o_{3'} = -0.30$ kcal/mol and $\overline{\Delta\Delta G}^o_{5'} = -0.42$ kcal/mol).

While we allow dangling energies only for the bases directly adjacent to a base pair, recent results indicate that there is a stabilizing effect for stretches

of up to 4 bases, with the difference of up to $-\Delta\Delta\bar{G}^o_{3'} = 1.0$ kcal/mol between 1 and 4 dangling bases [100]. However, it is not straightforward to incorporate this long range effect into the folding algorithms.

There are different ways to treat dangling end contributions. We will explain them in chapter 2.7 below.

2.6 Accuracy of dynamic programming RNA secondary structure prediction

The accuracy of predicting RNA secondary structures by physics-based dynamic programming as described above is dependent on the size of the molecules. The bigger a molecule is, the worse these production gets. This has several reasons, which we already shortly discussed:

- The different assumptions the model has to make
- The inaccuracy of the energy parameters
- The existence of pseudo-knots

The effect of these problems seems to pile up for longer molecules. Obviously, the probability to get at least one pseudo-knot is proportional to the sequence length. Of the different assumptions, the linearity of the multi-loop size parameters may be affected strongest, because the probability to get big multi-loops (in number of components as well as in number of free bases) also rises with the size of the molecule. For the additivity of loop energies as well as for the inaccuracy of the experimentally derived energy parameters, one could hope that deviations cancel out each other, but evidently this is not the case for most molecules. Finally, the length dependency of the performance can also be due to the kinetics of the folding.
As reported in the paper of Mathews et al. [81] the mean percentage of correctly predicted base pairs for the most advanced prediction methods is:

RNA	% base pairs correct
16S RNA	51.1
23S RNA	56.7
5S	77.7
Group I intron	69.9
Group II intron	88.2
RNAse P	54.5
SRP RNA	73.0
tRNA	83.0
Total	72.9

As can be seen, the algorithms perform best at short RNAs like tRNA (approx. 80 nt) or Group II introns (approx. 80 nt), and worst on 16S and 23S RNA, with lengths of 1600nt and 2900 nt, respectively. The fact that RNAse P is also badly predicted is probably due to the large number of pseudoknotted base pairs (14%, as opposed to e.g 0.2% at the 23S or 1.4% at the 16S RNA).

It must be emphasized that using a Nussinov style algorithm with hydrogen bond maximization, the average percentage of correctly predicted base-pairs using the same test set is 20.5%.

However, in a more recent study, Doshi et al. [29] used a different, more stringent definition of what correctly predicted means – while Mathews et al. considered a base pair to be correctly predicted also if it was shifted by one, i.e. if instead of the base pair (i, j) in the consensus structure the prediction was e.g. $(i + 1, j)$, Doshi et al. counted only base pairs as correct if both i and j coincided in prediction and consensus structure. Doshi et al. got correctly predicted values of 71%, 45%, and 43% for 5S, 16S and 23 S rRNAs, respectively. They also showed that the longer the contact distance of a base pair, i.e. the higher $j - i$, the worse the predictions of the algorithms.

2.7 Partition function

RNA molecules in physiological conditions are far away from being caged into a rigid secondary structure. Because the stabilizing energies of base pair formation are in the same energy range as the thermal energy, base pairs open and close all the time. Of course, tertiary structure interactions as well as interacting molecules can stabilize a certain secondary structure, but on the whole, an RNA molecule will adopt many possible secondary structures. Several approaches to use this to get more information about an RNA secondary structure exist. A possibility is the computation of a number of suboptimal structures in addition to the mfe structure. Manfred Zuker's algorithm [140] computes the best secondary structure that contains the base pair (i, j) for every base pair (i, j), while Stefan Wuchty's algorithm [136] predicts all secondary structures within a given energy band of the minimum free energy. We follow McCaskill's idea to compute the partition function of the secondary structures of an RNA molecule [84].

We define an ensemble of secondary structures $S(x)$ on a sequence x as the set of all secondary structures x can form, following the rules of secondary structure building 2.1. For every single one of these structures, there is an energy $E(s, x)$ of structure $s \in S$ being adopted by sequence x. If we now want to get the probability to find sequence x in structure s at a given time, this is proportional to the Boltzmann coefficient:

$$p(s|x) \propto e^{\frac{-E(s,x)}{kT}} \tag{3}$$

where k is Boltzmann's constant and T is the absolute Temperature. If we use molar units for the energies, this will change to

$$p(s|x) \propto e^{\frac{-E(s,x)}{RT}} \tag{4}$$

with R the Gas constant.
Every sequence x has to take on some structure, so:

$$\sum_{s \in S} p(s|x) = 1 \tag{5}$$

we can now define the exact probability of $p(s|x)$:

$$p(s|x) = \frac{e^{\frac{-E(s,x)}{RT}}}{\sum_{t \in S} e^{\frac{-E(t,x)}{RT}}} \tag{6}$$

$\sum_{t \in S} e^{\frac{-E(t,x)}{RT}} = Q$ is called the **partition function** of x. This partition function has lots of applications in thermodynamics. $-RT \ln Q$ is the Gibb's free energy of the ensemble, the probability of a certain the mfe structure being adopted is its Boltzmann weight divided by Q.
Generally, one can get the probability of any given feature by summing the Boltzmann factors of all structures showing this feature and then dividing by Q. How do these properties of the partition function help us with the prediction of RNA secondary structures?
First, one can look at the probability of the mfe structure. If it is high, then so is the confidence that it is at least close to the "real" structure. However, if RNA molecules get long enough, the probability of the mfe structure will always be very low. This is because the number of secondary structures a sequence can adopt rises exponentially with the length of the sequence($\psi(S_n) \approx 1.855^n$) [54]. Note that the total number of possible secondary structures of all sequences of length n also rises exponentially ($\psi(n) \approx 2.289^n$) [54], but much slower than the number of possible sequences (4^n), hence there are always fewer structures than sequences.

It has to be remembered that the measured energy values deviate around 10% from the true values [81, 138]. Partition function folding gives us the possibility to look at a bigger set of possible structures. What we will examine are the probabilities of base pairs. If a RNA molecule is randomly chosen out of the ensemble in thermodynamic equilibrium, what is the probability to find a certain base pair i, j formed in this molecule.
As stated above, one can compute that by simply summing up all Boltzmann weights for the structures containing i, j paired and dividing by Q. With this information, one can get a nice overview over all probable structures in the ensemble without generating them exhaustively. These pair probabilities are

not independent. Trivially, the probability to get a certain base pair is much higher if it can stack to an existing base pair. It is therefore not possible to compute the probability of a structural feature using the pair probabilities only.

Dangling ends

As explained above, dangling ends are unpaired bases that stack onto base pairs. Similarly, helices can stack onto each other also, bringing the π-electron-systems of their closing base pairs into interaction. This happens e.g. in the t-RNA clover leaf, where two extended "helices" are formed by stacking of one arm onto a neighboring one, respectively, thus creating two extended stacks which then form the L-shaped tertiary fold. While this "coaxial stacking" of helices can be viewed as a tertiary motif, it is still quite easy to incorporate it into the mfe prediction algorithms. In the `ViennaRNA` package, the dangling end energies are considered in four different ways, as specified by the user.
The default for the mfe case is that bases stack onto at most one base pair with minimum energy, i.e. if there are two stacks to choose, a base will dangle onto the stack where the most energy is to be gained. Another possibility is to let a base dangle onto every neighboring helix, regardless of what happens on the other side of this base. This could also be viewed as trying to mimic coaxial stacking. A full consideration of coaxial stacking is also implemented in the mfe case. Finally there is the possibility to neglect dangling ends altogether.

When computing the partition function, we use either no dangling ends or the version where a dangling base dangles to every neighboring stack. In case of "minimum energy" dangling or coaxial stacking, we would have to differentiate between e.g. the structure where base i dangles 3' and the structure where it dangles 5'. This would complicate matters and would also lack elegance. As the definition of secondary structure does not include dangling

ends, we would treat two (or more) identical secondary structures as different ones with slightly different energies, according to their dangles. We feel it is better to neglect this more complicated dangling end contributions.

2.7.1 Computation of the partition function

As we have seen, the partition function is a quite powerful tool for the analysis of the RNA secondary structure ensemble. It is very fortunate that it is quite easy to implement its computation in dynamic programming algorithms.
Let us first take a look at the Nussinov algorithm. In the simplest case, without any energy function, computing the partition function boils down to counting the number of possible secondary structures. Again, we can postulate that if we elongate a sequence by one base, this base is either paired or not. Furthermore,a new base pair i, j splits the sequence into two parts – an inner part and an outer part. With the energy contribution $\exp(i, j)$ for a base pair i, j, we get:

$$Q(i, j) = Q(i - 1, j) + \sum_k Q(i + 1, k - 1) \exp(i, k) Q(k + 1, j) \qquad (7)$$

As in the case of the mfe computation, $Q(i + 1, i)$ and $Q(i, i)$ are used as initialization, only in the partition function case they are set to 1.
As can easily be seen by comparison, the only difference between the mfe and the partition function algorithms is the usage of a sum whenever in the mfe there is a min and a multiplication instead of a summation. Thus, computing the partition function is computationally not more demanding than computing the minimum free energy.
We have to stress that this is also the case for the loop based energy model. However, the ease of the transition between the mfe and the partition function computation there is due to the fact that we made a unique multi-loop decomposition. This has not been necessary for the mfe case, but is of vital importance for getting the right partition function.

Equation 10 shows the computation of the partition function as it is implemented in the Vienna RNA package, including dangling end energies. We use $Q(i,j)$ as the partition function on the stretch i to j, $Q^B(i,j)$ as the partition function given i and j form a base pair, and Q^M and Q^{M1} are the partition function equivalents of the multi-loop contributions also used in equ. 2.

$$\begin{aligned}
Q(i,j) &= Q(i+1,j) + \sum_k Q^B(i,k) e^{d(i-1,i;k+1,k)} Q(k+1,j) \\
Q^B(i,j) &= \mathcal{H}(i,j) + \sum_{i<k<l<j; j+i-l-k>X} \mathcal{I}(ij,kl) Q^B(k,l) e^{d(i,i+1;j,j-1)} + \\
&+ \sum_{i+5\le k\le j-5} Q^M(i,k) Q^{M1}(k+1,j) e^{a(i,j)} \qquad (8) \\
Q^M(i,j) &= Q^M(i+1,j) e^c + \qquad (9) \\
&+ \sum_{i<k<l} Q^B(i,k) e^{d(i-1,i;k+1,k)} e^{b(i,k)} Q^M(k+1,j) + Q^{M1}(i,j) \\
Q^{M1}(i,j) &= Q^{M1}(i,j-1) e^c + Q^B(i,j) e^{b(i,j)} e^{d(i-1,i;j+1,j)}
\end{aligned}$$

with $\mathcal{I}(uv,xy)$ the exponent of the interior-loop energy between base pairs uv and xy, $\mathcal{H}(i,j)$ the exponent of the hairpin energy of base pair i,j, $d(u,v;x,y)$ the energy of base u dangling on v and x dangling on y of base pair vy, a the entropy penalty to close a multi-loop, b the penalty to extend a multi-loop by one stack, and c the penalty to extend a multi-loop by an unpaired base. Again, to keep the computation $\mathcal{O}(n^3)$, the length of interior-loops is restricted to $\le X$. The initializations $Q(i,i) = 1$ (corresponding to the open chain) and $Q^M(i,i) = Q^B(i,i) = Q^{M1}(i,i) = 0$ are used.

As partition functions can get very large, precautions have to be taken not to generate overflows. For that reason, a scaling is introduced. Every partition function $Q(i,k)$, and every $Q^B(i,k)$, $Q^M(i,k)$, $Q^{M1}(i,k)$ etc. is multiplied with a scaling factor $f < 1$ to the power of $k - i + 1$, the number of bases that generate it. This ensures that when adding up or multiplying partition functions, the result will be scaled correctly. At the end of the computation, when the free energy is computed, the scaling is removed again:

$$F = -RT(\ln Q + n \ln f) \qquad (10)$$

The scaling factor f is chosen so that $1 \approx Qf^n$. An estimate for f can be derived for the minimum free energy. As the Gibb's free energy is usually dominated by this term, this approach is sufficient most of the times. However, checks within the `ViennaRNA` package ensure that f is small enough and will put out a warning if it is not. It is then advisable to change f to a smaller value, which can be conveniently done using a command line option.

2.7.2 Base pair probabilities

The probability to get a base pair i, j is a very good representation of the partition function, which is fiendishly difficult to visualize otherwise. To compute the pair probabilities, we have to sum up over all structures that contain the base pair i, j and divide by the partition function. This can also be formulated thus:

$$p(i,j) = \frac{Q(i,j)}{Q}$$

with $Q(i,j)$ the partition function of all structures containing the base pair i, j. We can decompose this to

$$p(i,j) = \frac{Q^B(i,j)\hat{Q}^B(i,j)}{Q},$$

with $Q^B(i,j)$ and $\hat{Q}^B(i,j)$ the partition functions over all structures inside and outside i, j given that i, j pair, respectively. As $Q^B(i,j)$ is already computed in the course of the partition function algorithm, we are now concerned only with $\hat{Q}^B(i,j)$. To get this "outer" partition function, we again distinguish two main possibilities – either i, j is enclosed by another base pair k, l with $k < i < j < l$ or it is not. For the enclosed case, k, l can either close a multi-loop or an interior-loop. The multi-loop contributions can be written as parts where there is no base pair between k and i, a part where there is no base pair between j and l, and a part where there are base pairs on both

sides. Omitting dangles, this leads to

$$\begin{aligned}\hat{Q}^B(i,j) &= Q(1,i-1)Q(j+1,n) + \\ &+ \sum_{k<i,j<l} \hat{Q}^B(k,l)\mathcal{I}(kl,ij) + \\ &\quad +\hat{Q}^B(k,l))e^{a(k,l)}e^{b(i,j)}\left(Q^M(k+1,i-1)+Q^M(j+1,l-1)\right) \\ &\quad +\hat{Q}^B(k,l))e^{a(k,l)}e^{b(i,j)}Q^M(k+1,i-1)Q^M(j+1,l-1))\end{aligned}$$

with the usual boundaries concerning the interior-loops. The multi-loop part can be split to achieve $\mathcal{O}(n^3)$ computation time:

$$Q_t^M(i,l) = \sum_{k<i} e^{a(k,l)}(1+Q^M(k+1,i-1)\hat{Q}^B(k,l))$$

$$\hat{Q}^M(i,j) = e^{b(i,j)}\sum_{l>j} Q_t^M(i,l)(1+Q^M(j+1,l-1)+Q^M(j+1,l-1)$$

To visualize the base pair probabilities, dot plots are used. The size of the dot is proportional to the square root of the probabilities of the base pairs. In the lower left triangle, the minimum free energy structure is depicted (Fig. 9).

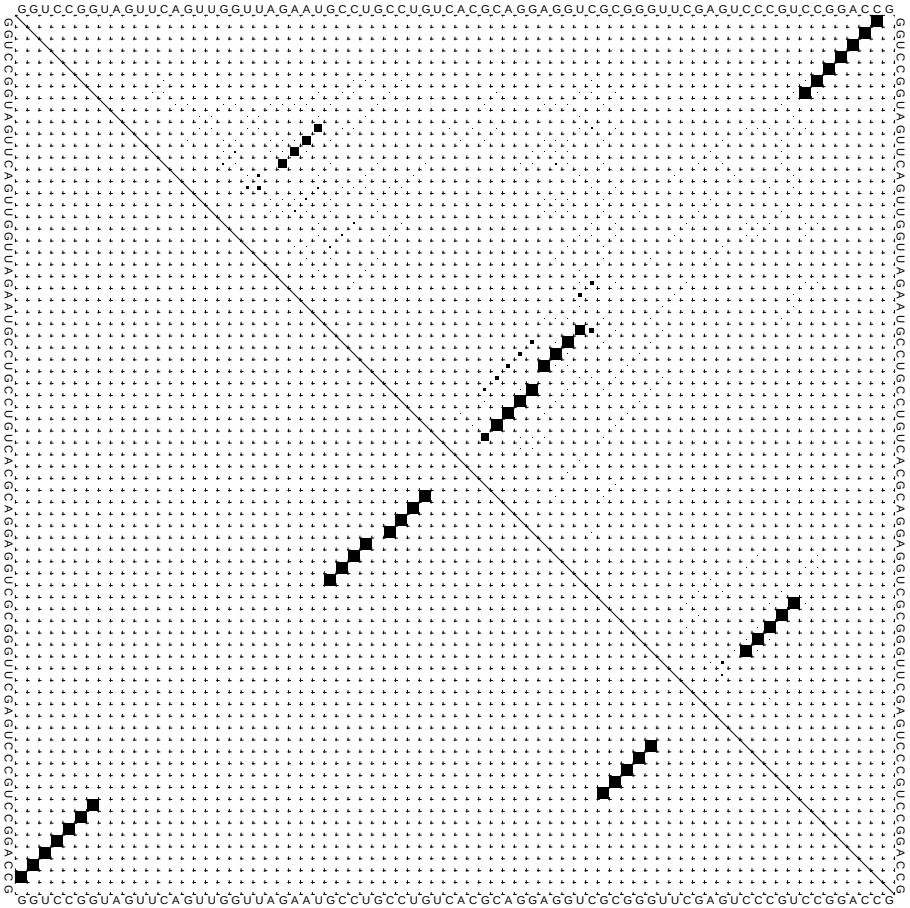


Figure 9: Dot plot showing the pair probabilities of a *B. subtilis* tRNA-Asp sequence. In this example, the mfe structure (lower left triangle) does not show the typical clover leaf shape, while the base pair probability plot (upper right triangle) shows that the stem missing in the mfe structure is quite probable in the ensemble.

2.8 Stochastic Context Free Grammars

In 1959, Noam Chomsky started to describe a formalization of grammars to construct correct sentences in languages [17] . He developed a hierarchy of increasingly complex grammars which can create increasingly complex sentences. After this hierarchy was adopted by computer scientists to describe programming languages and formalisms, in 1994 Eddy and Durbin [32] and Sakakibara et al. [113, 114] began using one of these grammars types to describe RNA secondary structure. All of Chomsky's grammar can be described by three classes of objects [31]:

- Non-Terminals
- Terminals
- Production rules

A production rule takes one or more Non-Terminals and rewrites them into a string of terminals (usually written as lower case letters) and Non-Terminals (usually upper case letters). A production rule now would look like this:

$$S \rightarrow aTb$$

The hierarchy of grammars is defined by the things that a production rule can do. The more complex the production rule, the higher up the grammar. In the simplest case, the so called "regular grammars", only rules that take one Non-Terminal and produce either one terminal or one terminal and one Non-Terminal are allowed:

$$\begin{aligned} S &\rightarrow a \\ S &\rightarrow aT \end{aligned}$$

for simplicity, this is usually abbreviated to:

$$S \rightarrow a \,|\, aT$$

with this grammars, one could generate all possible RNA sequences e.g. with these rules:

$$\begin{aligned} S &\rightarrow T \\ T &\rightarrow aT \,|\, cT \,|\, gT \,|\, uT \,|\, \epsilon \end{aligned}$$

Where the terminals a, c, g and u stand for the bases, ϵ is the terminator and S is used as a start Non-Terminal.
While you can construct every RNA sequence using a regular grammar, constructing RNA secondary structures is not possible with this simple grammar - there is more information than a regular grammar can deal with. The next level of complexity are context free grammars. Here, all productions rules that take one Non-Terminal and produce a string consisting of any number of Non-Terminals and terminals are allowed. They are called context free because (in contrast to more complex grammars), the production rules are independent of the neighbors of the Non-Terminal.
With this class of grammars, it is possible to create all RNA secondary structures. A very compact grammar is e. g. this one developed by Ivo Hofacker [30]:

$$S \rightarrow aS \,|\, aSa'S \,|\, \epsilon$$

where a, a' denote two pairing bases. This grammar can create all hairpin -, interior- and multi-loops possible, but also allows too short hairpins. Additionally, taking stacking of bases into account is not possible with this easy grammar. To include stacking, a more complex grammar like this one is needed:

$$\begin{aligned} S &\rightarrow aP^{aa'}a' \,|\, cS \,|\, Sc \,|\, SS \,|\, \epsilon \\ P^{aa'} &\rightarrow aP^{bb'}a' \,|\, S\,, \end{aligned}$$

where a, a' and b, b' are WC or wobble base pairs and c are unpaired bases.
While the production in the case of regular grammars follows a left to right direction, the context free grammars built up secondary structures from the inside to the outside. With one of these grammars, it is easily found out

whether a secondary structure can be realized by a certain sequence – if it is possible to create the structure by only using production rules of this grammar, the sequence matches the secondary structure.

2.9 Stochastic grammars

The grammars described above can only distinguish between strings that can be constructed using their rules and strings that can't. When dealing with RNA secondary structures, that means that they can tell whether a structure can be formed by a sequence, but all structures are equally plausible. Obviously, it is not possible to compute minimum free energies using these grammars. However, it is quite easy to create grammars that can distinguish between solutions. Stochastic grammars assign a probability to every production rule. For example, if stacking effects are to be taken into account, the production of a GC-CG stack should be more probable than the production of a GC-AU stack, because it has better energy.
There are several dynamic programming algorithms that deal with 3 different problems for stochastic grammars [31]:

- Calculate the optimal alignment of a sequence to a parametrized stochastic grammar. (Predict the mfe secondary structure)
- Calculate the probability of a sequence given a parametrized stochastic grammar. (Predict probability of a structure in the ensemble).
- Given a set of example sequences/structures, estimate the optimal probability parameters for an unparameterized stochastic grammar. (Assign "energies" to structural features)

These algorithms do only work on the *Chomsky normal form* of the grammars. In Chomsky normal form, there is a restriction on the production

rules. Only productions of the form:

$$\begin{aligned} S_1 &\rightarrow S_2 S_3 \text{ or} \\ S_1 &\rightarrow a \end{aligned}$$

are allowed. Every context free grammar can be cast into Chomsky normal form. However, the production rules are more numerous, therefore the Chomsky normal form is not the usual way to write down a grammar.

2.9.1 Inside

The inside algorithm [70] calculates the probability or score of a sequence/-structure pair with an SCFG (stochastic context free grammar). Let productions of the form $\mathcal{S}_x \rightarrow \mathcal{S}_y \mathcal{S}_z$ have probabilities $t_x(y,z)$ and $\mathcal{S}_x \rightarrow a\ e_x(a)$, and the probability of a string $i \ldots j$ to be generated by S_x is called $\alpha(i,j,x)$. For initialization:

$$\alpha(i,i,x) = e_x(x_i)$$

where x_i is the base on position i. With M different non-terminals, the iteration steps are:

$$\alpha(i,j,x) = \sum_{y=1}^{M} \sum_{z=1}^{M} \sum_{k=i}^{j-1} \alpha(i,k,y)\alpha(k+1,j,z)t_x(y,z)$$

The probability of a sequence x given an SCFG θ, is then

$$P(x,\theta) = \alpha(1,L,1)$$

The algorithm is $\mathcal{O}(L^3M^3)$ in runtime and $\mathcal{O}(L^2M)$ in memory.

2.9.2 Outside

While the inside algorithm computes the probability $\alpha(i,j,x)$ that the SCFG was used for a string $i...j$ of sequence x, the outside algorithm [70] computes the probability $\beta(i,j,x)$ of all possibilities outside i and j, i.e. it sums up all

probabilities on the sequence $1...i-1, j+1...L$. The outside algorithm uses the inside probabilities α computed by the inside algorithm. As initialization

$$\beta(1, L, v) = \begin{cases} 1 \text{ if } v = 1 \\ 0 \text{ if } 2 < v < M \end{cases}$$

as iterations, i is increased from 1 to L and j is decreased from L to j. The calculations of $\beta(i, j, v)$ are then performed like this:

$$\begin{aligned} \beta(i,j,v) &= \sum_{y,z} \sum_{k=1}^{i-1} \alpha(k, i-1, z)\beta(k, j, y)t_y(z, v) + \\ &+ \sum_{y,z} \sum_{k=j+1}^{L} \alpha(j+1, k, z)\beta(i, k, y)t_y(v, z) \end{aligned}$$

Termination for any i is:

$$P(x, \theta) = \sum_{v=1}^{M} \beta(i, i, v)e_v x_i$$

Using the βs and αs of these two algorithms, probability parameters can be estimated by expectation maximization.

2.9.3 CYK

The CYK (Cocke-Younger-Kasami) algorithm computes the most likely series of productions that generated a sequence, i.e. it can find the optimal free energy structure of an RNA sequence. In principle, it is an inside algorithm where the sums are replaced by max operations. In addition, a so-called "traceback" variable is kept, which is used to find the optimal alignment (or secondary structure).

2.9.4 Applications of SCFGs

There are several RNA-related applications of SCFGs. Besides alignment algorithms, Rivas and Eddy developed QRNA [111], an early ncRNA finder, and pknots [110], which can predict pseudo knotted structures. Bjarne Knudsens pfold [61] computes common structures for an alignment of RNA molecules.

2.9.5 Disadvantages and Advantages of SCFGs

SCFGs have certain drawbacks that are intrinsic to the approach. The logarithmic length dependence of a hairpin or interior-loop cannot easily be modeled, as that would need production rules for every possible kind and length of interior or hairpin-loop. Even if the length of the loops was restricted, this would lead to so many rules that overfitting would make the training of the SCFGs impossible. An advantage of SCFG approaches is the fact that they do not rely on experimentally derived parameters. All the parameters used can be fitted using molecular structure data.

2.10 CONTRAfold

While having been around since the early 90s, SCFG based algorithms for RNA secondary structure related tasks have almost always been outperformed by physics-based algorithms using the Turner energy model. In 2006, Do et al. [27] put forward a program, CONTRAfold, that augments the SCFG approach by using CLLMs, conditional log-linear models. CLLMs generalize upon SCFGs by removing some built in assumptions.

CLLM training employs the conditional maximum likelihood principle, which focuses on finding parameters that give good predictive performance. As they are a generalization of SCFGs, CLLMs can easily incorporate sophisticated scoring schemes of thermodynamic methods, and can replace hard to measure scores by training. They e.g. score not only Watson-Crick and wobble, but all possible base pairs.

Furthermore, they use the maximum expected accuracy approach to balance between loss of specificity and gain of selectivity. These characteristics lets CONTRAfold outperform even the thermodynamic methods, but the maximum expected accuracy approach can also lead to structures that, while sharing a maximum of base pairs with the ?consensus structures, are definitely not the minimum free energy structure. (See Fig. 10).

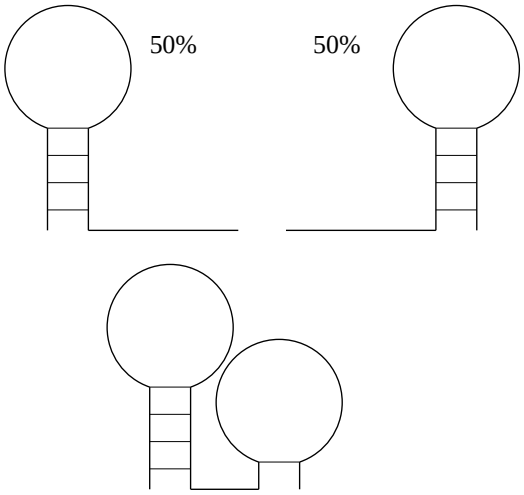


Figure 10: Possible caveat of the maximum expected accuracy approach. At the top, a hypothetical riboswitch with two equally probable states is shown.
At the bottom, the possible prediction of a mea algorithm captures as many base pairs as possible, but the structure is very improbable.

3 Additions to RNAfold

In this section, refinements of the RNAfold algorithm which were done in recent years and to which this work contributed to are introduced. The computation of the partition function of canonical structures is a part of A. F. Bompfünewerer's most recent work [9].

3.1 Canonical structures

3.1.1 computing canonical structures

In the context of RNA secondary structure, a canonical structure is a structure where every base pair is stacked. While they do exist, see e.g. the *B. subtilis* RNAse P structure in fig 5, these so called "lonely", i.e., unstacked base pairs are not common in nature. As there is no stacking energy, they are stabilized by dangling bases exclusively.
We expanded the existing option for RNAfold to generate only canonical structures to partition function folding. This had to be done with minimal changes to the existing code, to keep the ViennaRNA package easily maintainable and the functions of the ViennaRNA library backward compatible. In principle, we followed the approach taken in the mfe computation. The recursion has to be changed as can be seen in the Feynman diagrams in Fig.12

The changes to the code are kept minimal. Instead of directly computing the Q^B entries, we temporarily save the respective entries for $i+1$ in Q^{B*}. $Q^{B*}(i,j)$ is a structure that is made canonical by a base pair $(i-1, j+1)$. Q^B is then filled with canonical structures only. In the case of stacking base pairs, $Q^B(i,j)$ is filled using $Q^{B*}(i+1, j-1)$, while the canonical Q^Bs are used for all other cases. For the computation of the base pair probabilities, the same approach is used. The recursions, omitting dangles, now look like

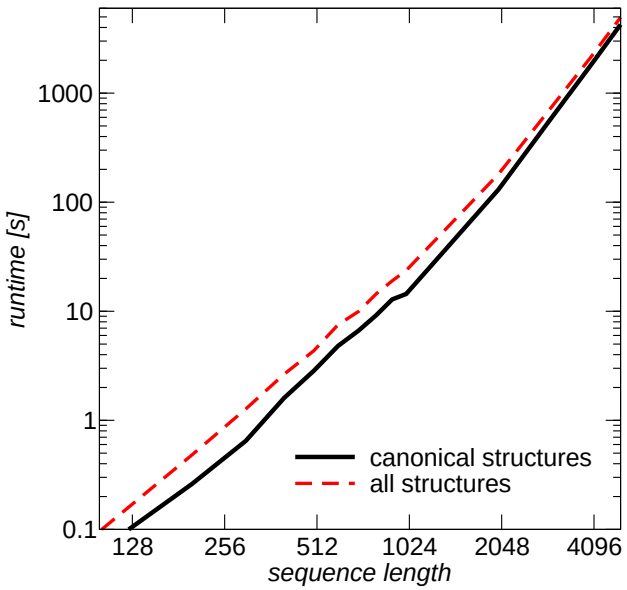


Figure 11: Performance of the canonical partition function computation compared to the standard version.

this:

$$
\begin{aligned}
Q(i,j) &= Q(i+1,j) + \sum_k Q^B(i,k)Q(k+1,j) \\
Q^B(i,j) &= Q^{B*}(i+1,j-1)\mathcal{I}(ij,i+1j-1) \\
Q^{B*}(i,j) &= \mathcal{H}(i,j) + \sum_{i<k<l<j;j+i-l-k>X} \mathcal{I}(ij,kl)Q^B(k,l) + \\
&+ \sum_{i+5\leq k\leq j-5} M(i,k)M^1(k+1,j)e^{a(i,j)} \qquad (11) \\
M(i,j) &= M(i+1,j)e^c + \\
&+ \sum_{i<k<l} Q^B(i,k)e^{b(i,k)}M(k+1,j) + M^1(i,j) \\
M^1(i,j) &= M^1(i,j-1)e^c + Q^B(i,j)e^{b(i,j)}
\end{aligned}
$$

This assures that all structures are canonical, as the building blocks used either already are canonical or are made canonical by the stacking base-pair.

When computing $\hat{Q}^B(i,j)$, the same approach is used. Computing the base

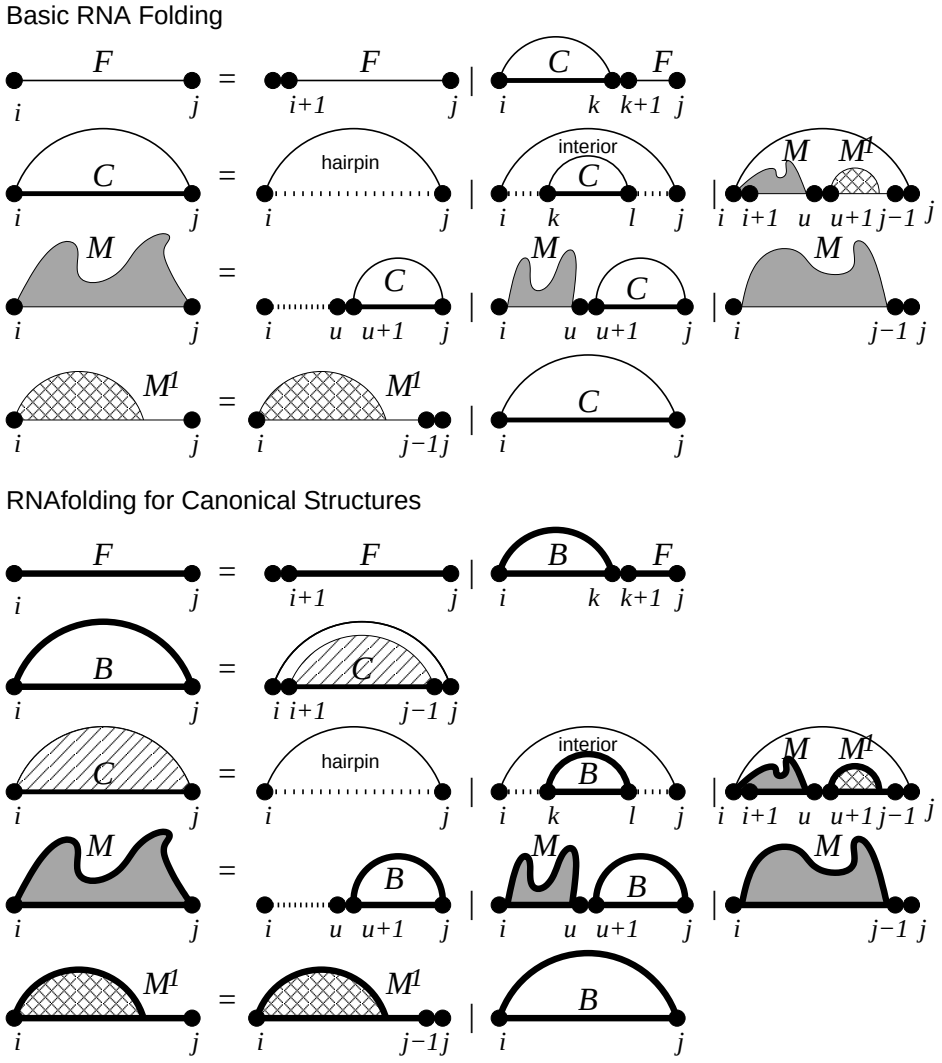


Figure 12: Feynman diagrams showing the difference between the computation of all (top) and canonical secondary structures only. [9].

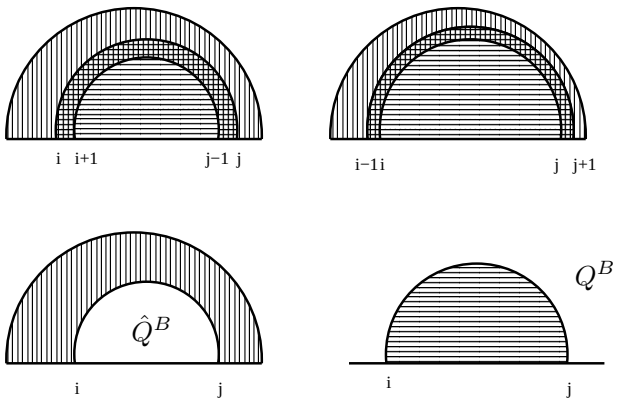


Figure 13: Illustration of the overlapping computations of the $p(i,j)$s. At the top, $\hat{Q}^B(i,j)Q^B(i-1,j+1)$ and $\hat{Q}^B(i+1,j-1)Q^B(i,j)$ are shown. The overlapping interior-loop contributions can be seen. The bottom just explains how $\hat{Q}^B$ and Q^B are depicted here.

pair probabilities is a little more tricky, as there are base pairs i,j, made canonical by a neighboring base pair, where $Q^B(i,j)=0$ or $\hat{Q}^B(i,j)=0$. We therefore have to use a post-processing step, where we construct the probability $p(i,j)$ of a base pair to appear in the ensemble out of:

$$\begin{aligned} p(i,j)Q &= \frac{\hat{Q}^B(i,j)Q^B(i-1,j+1)}{\mathcal{I}(i-1j+1,ij)} + \\ &+ \frac{\hat{Q}^B(i+1,j-1)Q^B(i,j)}{\mathcal{I}(ij,i+1j-1)} - \hat{Q}^B(i,j)Q^B(i,j) \end{aligned}$$

That is, by using an overlap (see Fig. 13),

$$\begin{aligned} &\hat{Q}^B(i,j)Q^B(i-1,j+1) \text{ and} \\ &\hat{Q}^B(i+1,j-1)Q^B(i,j), \end{aligned}$$

we assure that if there is a possibility to get a canonical base pair, we will count it. As this strategy will add the stacking energy of i,j two times, we have to divide by this energy again, e.g. by $\mathcal{I}(i-1j+1,ij)$. Finally, structures in which (i,j) stacks both onto $(i-1,j+1)$ and $(i+1,j-1)$ are counted twice. Therefore, we subtract $\hat{Q}^B(i,j)Q^B(i,j)$.

3.1.2 Performance

Some base pairs, namely the ones that can not stack in either direction (i.e. where neither $i-1, j+1$ nor $i+1, j-1$ can pair) can never occur in a canonical structure. These can be neglected in the computations, and thus, the computation is a little faster than for the non-canonical structures (Fig 11). In some cases, using canonical structures can lead to better predictions (see Fig 14). Alas, this is not generally the case. The statistics for a subset of Rfam [42] structures provided by CONTRAfold [27] show no significant difference between canonical and "normal" secondary structures (Wilcoxon p-values of $\approx$0.9 to be of the same distribution).
Here, we use the comparison scheme developed by Gardner et al. [36] for their BRALIBase I benchmark of structure prediction tools for the mfe structures. This basically states that base pairs which do not contradict the consensus structure (i.e. which can be formed in addition to all base pairs of the consensus structure, but are not part of the consensus structure), are not counted as false positives. For partition function and pair probability, we simply added the probabilities of the base pairs which are in the reference structures for the true positive rate. For the false positive rate, we added up all pair probabilities for base pairs contradicting the reference structure.

In the case of the tRNA seed subset of Rfam, the results are a little bit more promising (see table 1). When we compare the results for tRNAs to the partition function versions, we can see an improvement of the MCC results as well as for the partition function results. (all Wilcoxon p-values were computed by using the R software package). This essentially means that it is very probable that the partition function of canonical structures gives better results than usual thermodynamic folding, at least for tRNA structures, which are known not to contain lonely base pairs.

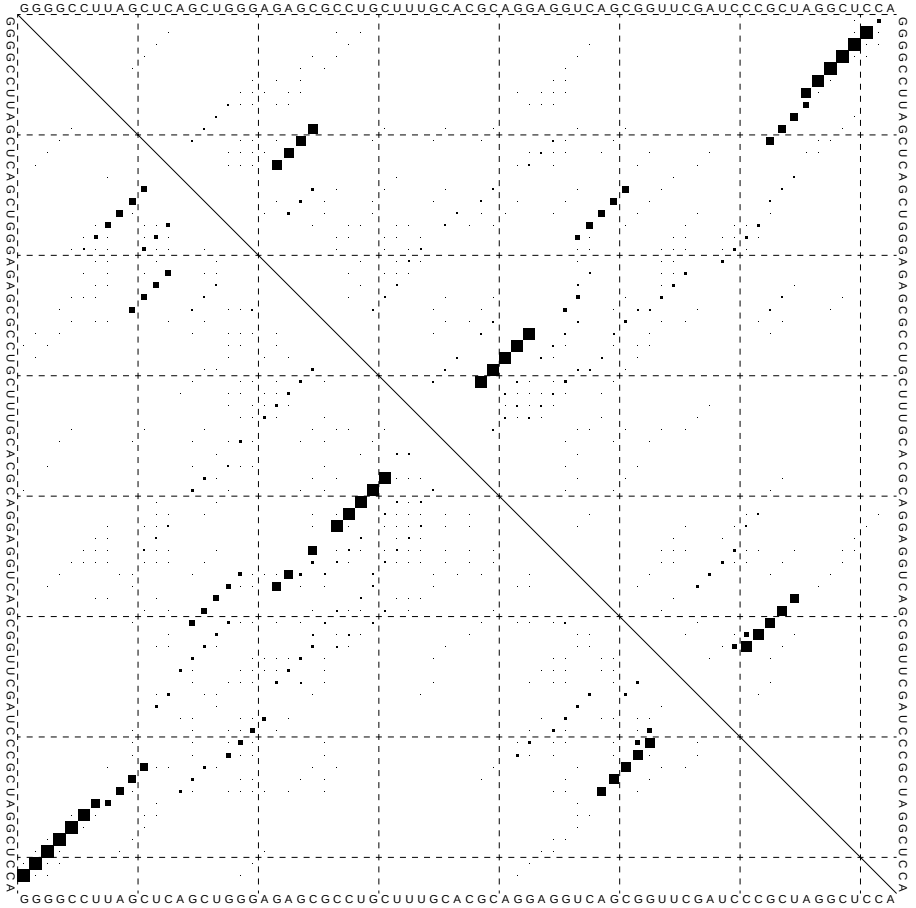


Figure 14: Comparison of pair probability dot plots of a *Streptococcus suis* tRNA-Ala with canonical (upper right triangle) and normal (lower left triangle) computation. As can be seen, the clover leaf structure is better represented when using canonical structures [9].

data	method	mean LP	mean no LP	Wilcoxon p-values
CONTRAfold	mfe	68.81%	68.62%	0.90
CONTRAfold	p-func TP	66.36%	66.87%	0.80
CONTRAfold	p-func FP	115.72	115.54	0.93
tRNA	mfe	61.8%	62.8%	0.36
tRNA	p-func, TP	62.84%	63.78%	0.20
tRNA	p-func, FP	41.08	39.35	0.07

Table 1: Performance of the canonical structure prediction compared to the usual prediction of `RNAfold`. For the mfes, the values compared where the MCCs, for the partition functions, see text for comparison strategy. TP = true positive, FP = false positive. no LP = canonical structures. .

3.2 Interior-loops

3.2.1 Introducing a real $\mathcal{O}(n^3)$ algorithm of interior-loop free energy computation

In the Turner energy model, the energy contributions of large interior-loops are dependent on the size of the interior-loop, its asymmetry as well as on the closing base-pairs, i.e.

$$\begin{aligned}\mathcal{I}(i,j;k,l) &= m(i,j;i+1,j-1)+m(l,k;l+1,k-1)+ \\ &\quad L(k-i-1+j-l-1)+A(|(k-i-1)-(j-l-1)|),\end{aligned}$$

where $A(l)$ is the penalty for a loop with asymmetry l, $L(l)$ is the length dependent contribution and $m(i,j;i+1,j-1)$ are mismatch energies for the unpaired bases $i+1, j-1$ adjacent to the pair (i,j). Rune Lyngsø et al. [77] used this fact to develop an algorithm for computation of interior-loop contributions in $\mathcal{O}(n^3)$. Essentially, they exploited that the asymmetry of an interior-loop does not change if 1 nucleotide is added on each side of the loop. For our partition function version, we introduce the quantity $I(i,j,l)$ as the partition function over all structures on the region $i \ldots j$ that end in an interior loop of length l closed by (i,j), excluding the mismatch energy

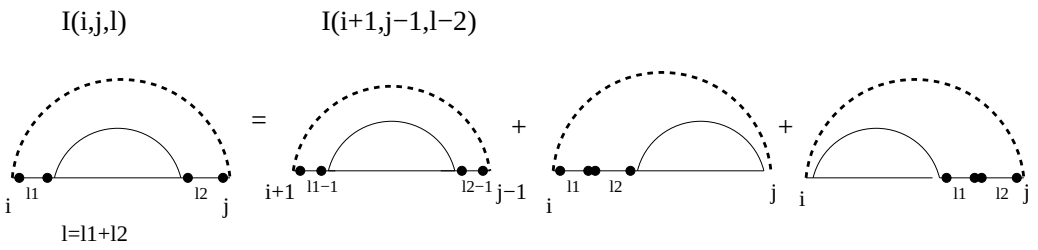


Figure 15: Illustration of the decomposition of an interior loop of size l into two bulges and an interior loop of size $l-2$.

$m(i,j;i+1,j-1)$ and the length dependent term $L(l)$. In other words

$$I(i,j,l) = \sum_{k,l} \exp(-\beta(m(l,k;l+1,k-1) + A(|(k-i-1)-(j-l-1)|))),$$

with $\beta = \frac{1}{RT}$. The partition function over all inner interior-loops by (i,j) can now be computed as

$$I(i,j) = e^{-\beta m(i,j,i+1,j-1)} \sum_{l} I(i,j,l) e^{-\beta L(l)} \tag{12}$$

The $I(i,j,l)$ can be computed more efficiently by noting that an interior loop of length l is either the extension of an interior loop of size $l-2$ or a bulge loop of size l (on either the left or right side), see also Fig. 15. We can therefore compute the I values as

$$\begin{aligned} I(i,j,l) &= I(i+1,j-1,l-2) + \\ &\quad + e^{-\beta A(l)} e^{-\beta m(i+1,j-l-1;i+2,j-l-2)} Q^B(i+1,j-l-1) + \\ &\quad + e^{-\beta A(l)} e^{-\beta m(i+l+1,j-1;i+l+2,j-2)} Q^B(i+l+1,j-1) \end{aligned}$$

This leads to a folding algorithm that runs in $\mathcal{O}(n^3)$ time without restricting interior loop size.

As most interior-loops of sizes $l \leq 5$ have tabulated experimentally derived energies, and the energy functions $A(l)$ and $L(l)$ are not the same for interior-

loops and bulges, the full implementation is even more complicated:

$$
\begin{aligned}
I(i,j) &= I^{\mathrm{t}}(i,j) + I^{L}(i,j) \\
I^{\mathrm{t}}(i,j) &= \sum_{(k-i-1)+(j-l-1)\leq 5} \mathcal{I}(i,j;k,l) Q^{B}(k,l) \\
I^{L}(i,j) &= e^{-\beta m(i,j,i+1,j-1)} \sum_{5<l<j-i-5} I^{l}(i,j,l) e^{-\beta L(l)} + \\
&+ e^{-\beta m(i,j,i+1,j-1)} e^{-\beta L_B(l)} e^{-\beta A_B(l)} I^{B}(i,j) \\
I^{l}(i,j,l) &= I^{l}(i+1,j-1,l-2) + \\
&+ e^{-\beta A(l-2)} e^{-\beta m(i+2,j-l-2,i+3,j-l-3)} Q^{B}(i+2,j-l-2) + \\
&+ e^{-\beta A(l-2)} e^{-\beta m(i+l+2,j-2,i+l+3,j-3)} Q^{B}(i+l+2,j-2) \\
I^{B}(i,j) &= Q^{B}(i+1,j-l-1) e^{-\beta m(i+1,j-l-1,i+2,j-l-2)} + \\
&+ Q^{B}(i+l+1,j-1) e^{-\beta m(i+l+1,j-1,i+l+2,j-2)}
\end{aligned}
$$

where $L_B(l)$ and $A_B(l)$ are the length and asymmetry energy functions for bulges, $I^{\mathrm{t}}(i,j)$ contains the tabulated entries, and $I^{L}(i,j)$ all other interior loop energies.

3.2.2 Performance

In practice, the $\mathcal{O}(n^3)$ algorithm is still slower than the restricted version used currently in the `ViennaRNA` package. However, one can combine the technique above with an upper bound for l in order gain performance, see Fig. 16. Therefore, even when using the new $\mathcal{O}(n^3)$ algorithm, we restrict the size of interior-loops to 30 by default. As can be seen in Fig. 17, the difference between a maximum loop size of 1000 and 30 is very small. This indicates that for thermodynamic computations, it is not really necessary to compute all interior loops – as their free energy rises fast with their length. While large loops are rare in equilibrium or ground state, they may however occur as intermediates in folding pathways (Christoph Flamm, personal communication). Recently, even faster algorithms for computing interior-loop contributions have been developed [99]. These algorithms can compute all possible interior-loops in $\mathcal{O}(M \log^2 n)$, where $M < n^2$ is the number of

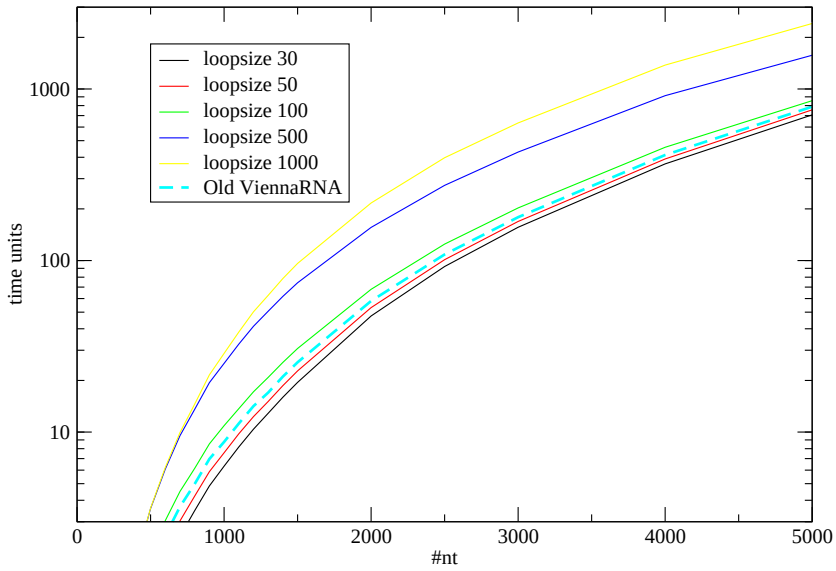


Figure 16: Time requirements comparison between the original `ViennaRNA` implementation (dashed) and the $\mathcal{O}(n^3)$ implementation following Lyngsø et al. with different maximum loop sizes.

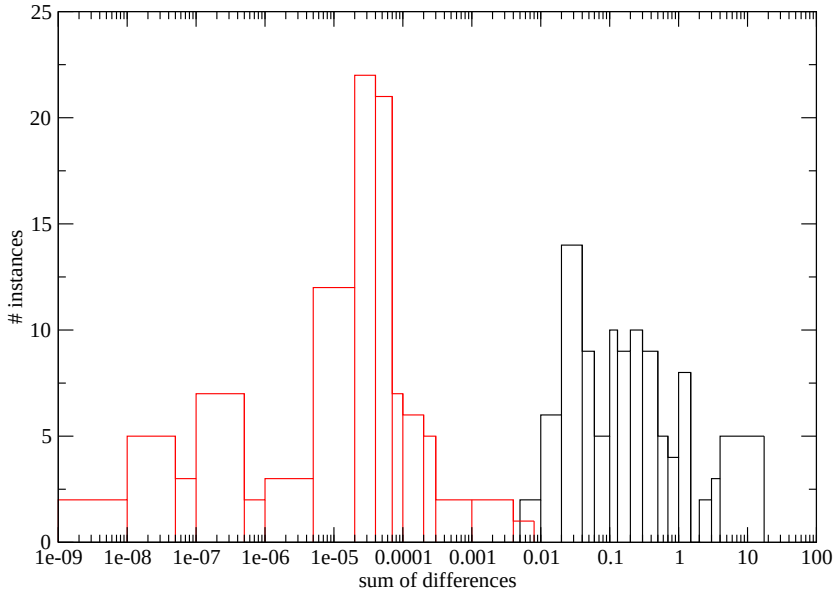


Figure 17: Histogram of the sum of the differences of all pair probabilities when computed with maximum loop size 15 and 30 (black) and 30 and 1000 (red). 50 random sequences of length 2000 and 50 random sequences of length 3000 were folded. As can be seen, the differences between loop sizes 15 and 30 are significantly larger than in the 30 to 1000 example.

possible base pairs. However, as they use branch and bound approaches, they are only applicable for mfe computations. Furthermore, while this would obviously lead to a speed-up, it must be emphasized that the folding algorithm itself still stays $\mathcal{O}(n^3)$ due to the multi-loop computations.

4 Joint secondary structures of more than one RNA molecule

4.1 Biology

Many functions of RNA in biological systems are governed by interactions between several RNA molecules. In RNA interference and the miRNA and piRNA pathway, a small RNA forms base pairs to a target mRNA. In at least some RNA editing pathways, a small RNA is used to guide the editing machinery to its targets. H/ACA and C/D box sno-RNAs interact with their target rRNA. This seems to be an often used mechanism in more complex cells. The protein machinery which catalyzes a certain function is guided to its targets by small RNAs, which use base pairing to recognize these targets. By using this mechanisms, cells can avoid an explosion of the number of proteins necessary for controlling gene expression, which appears to grow quadratically with genome size in prokaryotes [22].

RNAs using this type of interactions also help to rectify the so-called G-value paradox, the fact that the number of protein coding genes does not coincide with biological complexity [122]. The number of transcripts which do not code for protein in a eukariotic cell is very high. Up to 85% of the human genome are transcribed, with a recently published reliable number of about 15% [105], while only about 3% code for proteins. Many of the described transcripts can as of yet not be assigned to any known RNA family and it seems possible that at least some of these act via RNA interactions.

In prokaryotes, there is also a number of RNA-interaction governed regulations. Many of their regulatory RNAs (termed small RNAs – sRNAs or regulatory RNAs – rRNAs), like e.g. RyhB, OxyS, DsrA or SgrS, base pair with their target mRNAs with aid of the protein Hfq [40]. Roughly a third of the known non coding RNAs of *E. coli* act in this way [40].

4.2 Chemistry

When the building of a dimer or multimer out of two or more molecules is to be computed, several physico-chemical properties of the dimerization/-multimerization reaction have to be taken into account. First of all, the binding of n molecules will result in a loss of degrees of freedom – instead of the $3n$ degrees of freedom for translation, there are only 3, instead of the $3n$ degrees of freedom for rotation we have only 3. It is easy to see that this results in a loss of entropy. For RNA molecules, this loss can be considered to be independent of the nucleotide sequence of the molecules. In the Turner energy model, its value, which we will call duplex initiation penalty Θ^I, is 4.1 kcal/mol/binding. Another important feature of this type of reaction is that it is obviously concentration dependent. The number of multimers that can form is a function of the number of monomers – in a very dilute solution, you do not expect many multimers. In the limit of very high concentrations, many duplexes will form, and the bigger the concentrations the more multimers will form, also. It will be seen that taking the computations beyond trimer formation is, while possible, increasingly complicated. Therefor, we will first stick to dimer formation, then describe how to compute trimers and how to expand this to n-mers. The concentration of dimers (or multimers) and monomers is connected by the equilibrium constant K via the mass action law.

$$[AB] = [A][B]K_{AB}$$

In other respects, there is little difference between intermolecular and intramolecular base pairing. A base does not "know" whether its counterpart in a base pair is linked to it via the backbone or not. The only differences are of entropic and sterical nature.

4.3 Previous work

The idea to predict dimer secondary structures using dynamic programming algorithms is not new. The first approaches to do this were to link the two

sequences using "virtual" nucleotides and use mfold or `RNAfold` on the linked sequence. As linkers, either short sequences that form very stable hairpins (from here on called "hairpin linkers") or nucleotides which are not allowed to form base pairs (called "poly-N linkers") were used. This idea has certain merits, but it runs into some problems because of the energy model: In the mfe case, the "hairpin linker" causes problems because

i The energy of the hairpin is added to the energy of the two molecules – this will lead to an energy too low.

ii The base pairs of the hairpin's stack can build stacks or interior-loops and multi-loops with the rest of the molecules – this can lead to wrongly predicted structures.

iii The bases of the hairpin can base pair with the rest of the molecule, which leads to wrongly predicted structures.

While the energy contribution can easily be subtracted again, there is no easy way to correct for the effect in ii. Nevertheless, this approach was used by many researchers, e.g. [120]. For the partition function case, the effects are even more pronounced. The bases of the linker can and will take part in the suboptimal structures of the molecule. Even if you restrict the linker bases to the hairpin structure of the linker – getting rid of problem iii – you still get unwanted energy contributions which will distort your energy landscape. For partition function calculations, this approach simply will not do. For the "poly-N linkers", the situation is a little better, it was used e.g. in the Oligowalk program by Mathews et al. [80]. Here, the hairpin will lead to a wrong energy of the exterior-loop, which will also lead to wrongly predicted structures. The link between two bases which are not part of the same molecule will also lead to exterior-loops being treated as interior-loops. However, Mathews et al. [80] went around these problems by treating the loop containing the linker in a different way, applying an initiation penalty instead of a loop penalty, which is in essence what the mfe version of `RNAcofold` does.
Recognizing these problems, algorithms not using linkers were put forward.

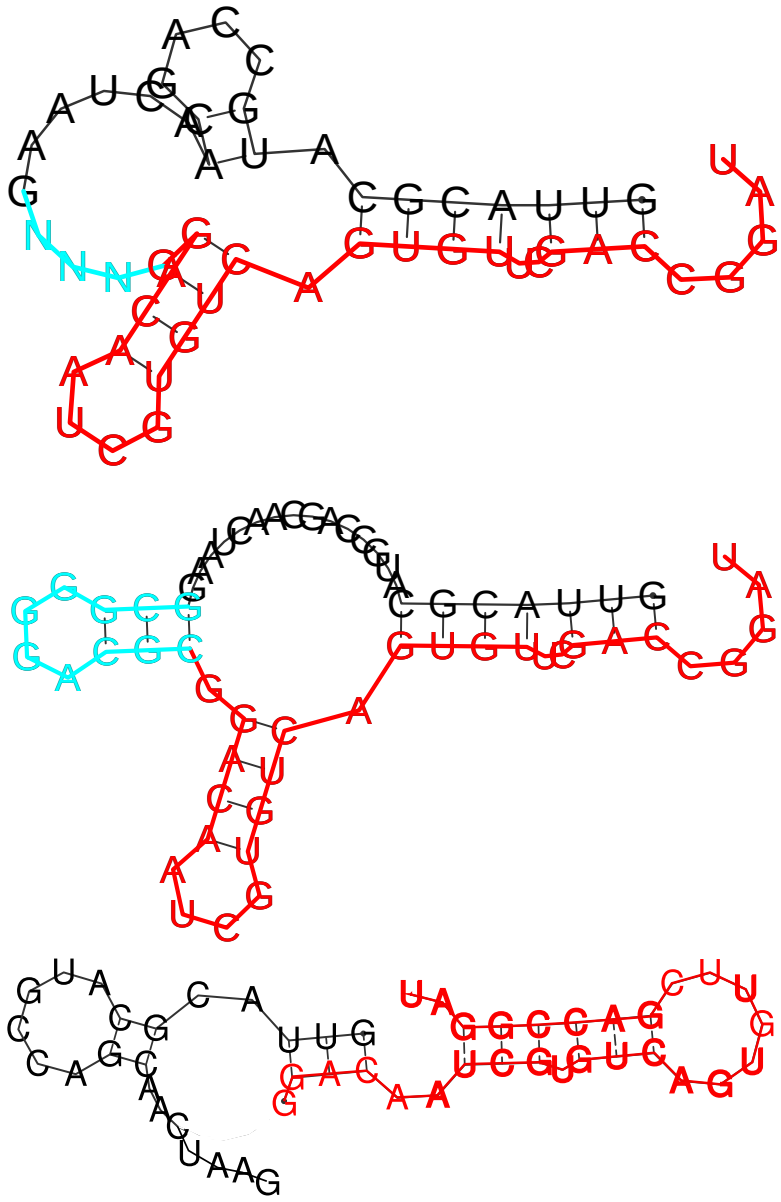


Figure 18: Comparison of dimer minimum free energy structures predicted by different means. The first molecule is drawn black, the second red. Linkers are drawn in cyan.
Top: Structure when using a "poly-N" linker.
Middle: Structure when using a typical linker sequence taken from [120], with constraints forcing the linker to take a hairpin structure
Bottom: Structure from `RNAcofold` (described below).
While the two linked structures mostly show the same base pairs, `RNAcofold` gives a different picture. The energy difference of the two structures is 0.1 kcal/mol, which is only slightly above 1 % (-7.4 to -7.3 kcal/mol). This figure only illustrates that wrong structures can arise.

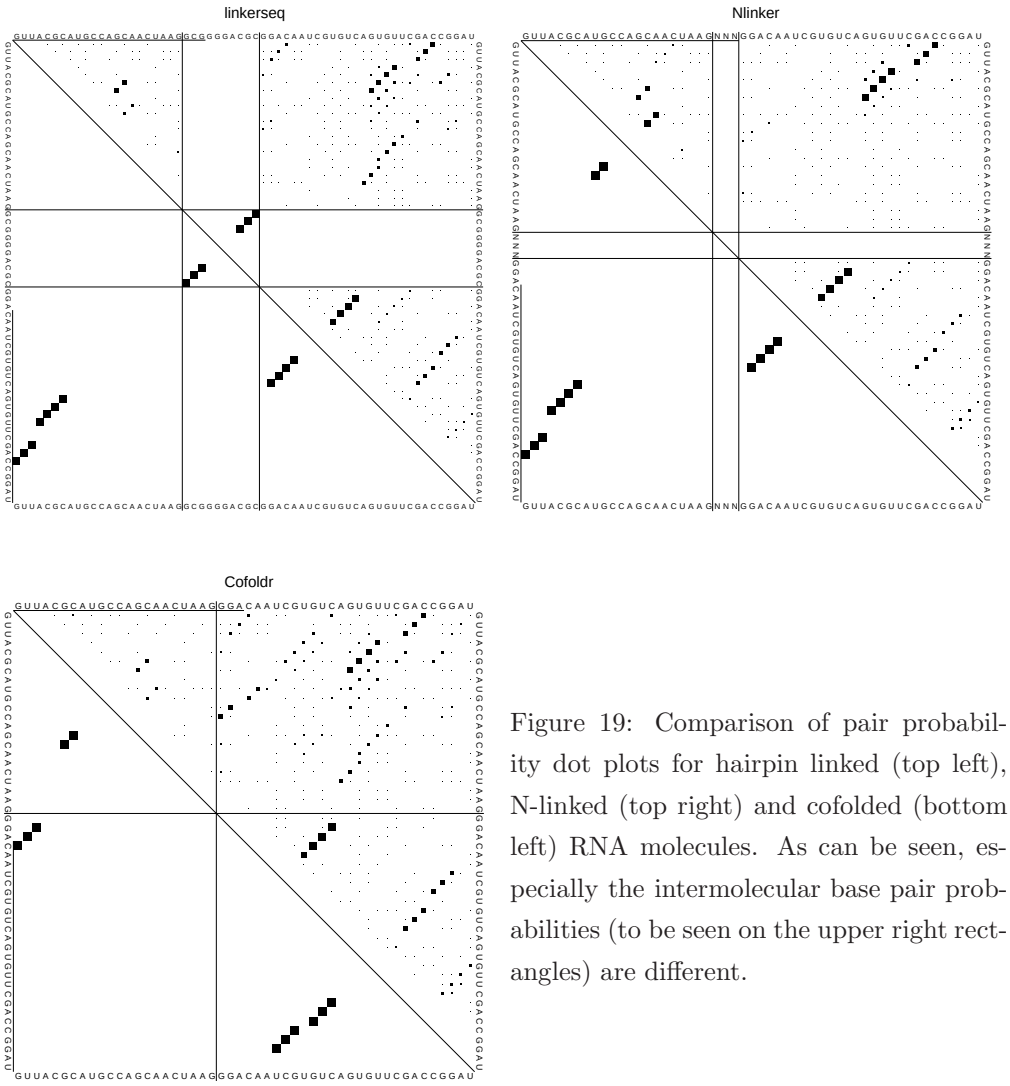


Figure 19: Comparison of pair probability dot plots for hairpin linked (top left), N-linked (top right) and cofolded (bottom left) RNA molecules. As can be seen, especially the intermolecular base pair probabilities (to be seen on the upper right rectangles) are different.

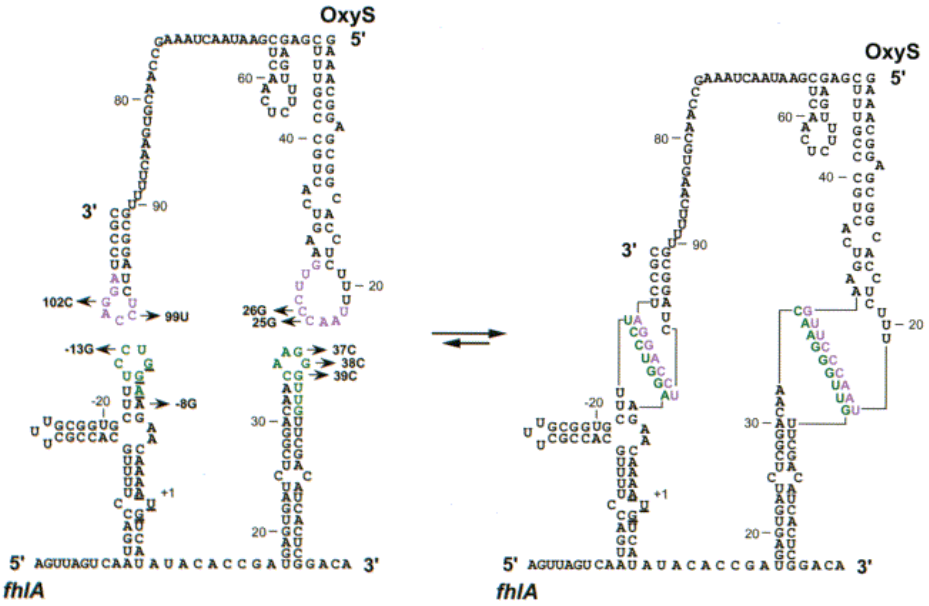


Figure 20: Secondary structure of the fhlA and OxyS molecules (left) and their dimer (right). There are two kissing hairpins formed, which can not be detected by sequence concatenation, as they are intermolecular pseudoknots. Figure taken from [4]

Dimitrov and Zuker suggested an algorithm where the molecules could not form intramolecular base pairs [23], and Andronescu et al. [3] wrote an algorithm that in essence does the same as the mfe version of `RNAcofold`, but can handle more than two molecules. They also designed a two molecule version of Wuchty's [137] `RNAsubopt` algorithm.

The concatenation of two molecules, however, is limited by the fact that complicated structures, e.g. structures that have intermolecular pseudo knots, can not be predicted. These structures do however occur in nature, e.g. in snoRNA/rRNA duplexes or in the OxyS/mRNA duplex in bacteria (see Fig. 20). Because of this problem, other, more complicated approaches were introduced. Alkan et al. proved that the RNA cofolding problem is NP-hard even if there are no pseudoknots allowed intramolecularly [1] – the reason for this is a structure called an entangler (Fig.21. They also proposed an algorithm which covers many classes of possible structures, and uses heuristics to reduce the considerable processor requirements of $\mathcal{O}(n^3m^3)$ and memory

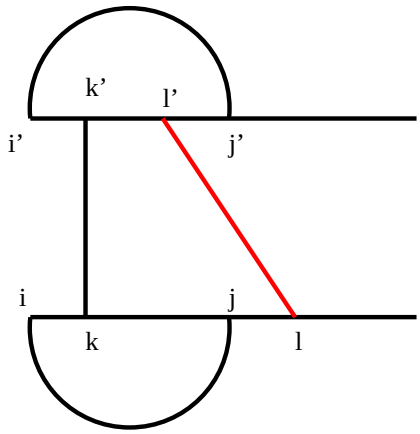


Figure 21: A minimal entangler structure. Intermolecular base pairs are arcs, intermolecular are lines connecting the molecules. Both the red line (l, l') and the black line (k, k') make this structure an entangler. If one of them was missing, this would be a structure which could be predicted in polynomial time.

requirements of $\mathcal{O}(n^2m^2)$. Pervouchine developed IRIS [103], a $\mathcal{O}(n^6)$ algorithm which can deal with all types of intermolecular interactions except the entanglers. He also takes the geometry of the backbone into account, i.e. he uses the triangle inequality to eliminate some – but not all – sterically impossible structures.

However, these programs are computationally very expansive, so we decided on the one hand to enhance the concatenation techniques, and on the other hand, we used the accessibility of possible binding sites to get intermolecular structures [92]. This approach will be discussed in more detail in section 4.14.

4.4 Computation

In principle, computing a secondary structure for an RNA dimer is easy. The concatenation of the two molecules is a simple way to use the well-established folding algorithms. Two things have to be taken into account: Any loop spanning the concatenation site is an exterior-loop, and if, and only if, there exists at least one intermolecular base pair, the duplex initiation penalty E^I

has to be added to the free energy of the structure. For minimum free energy folding, `RNAcofold` has been a part of the `ViennaRNA` package for quite some time. However, the necessity of counting every possible structure exactly once has made partition function calculation somewhat more difficult.
The solution of this problem is keeping track of the concatenation point – we use a value we call "cut_point" c_p, which is the first nucleotide of the second sequence – and check at all loop computations and dangling ends contributions whether they span the concatenation point:

- Hairpins i, j: exterior-loop if $i < c_p \leq j$
- Interior-loops $i, j; k, l$: exterior-loop if $i < c_p \leq k$ or $l < c_p \leq j$
- Multi-loops: If there is no base pair spanning $c_p - 1, c$
- Dangles: 5': dangling base $\neq c_p - 1$, 3': dangling base $\neq c_p$

All these Loop cases are replaced by one exterior-loop case: when computing the hairpins, we simply add $Q(i+1, c_p-1)Q(c_p, j-1)$ to $Q^B(i,j)$, which are all possible structures between $i+1$ and $j-1$ without intermolecular base pair. The full recursion of the partition function calculation then looks like this:

$$
\begin{aligned}
Q_{ij} =& Q_{i+1,j} + \sum_{i<k\leq j} Q^P_{i,k} \hat{d}^E_{ik} Q_{k+1,j} \\
Q^P_{ij} =& \begin{cases} \hat{\mathcal{H}}(i,j) \\ Q_{i+1,n_1} Q_{n_1+1,j-1} \hat{d}^I_{ij} \end{cases} \\
&+ \begin{cases} \sum_{i<k<l<j} Q^P_{kl}\, \hat{\mathcal{I}}(i,j;k,l) \\ 0 \end{cases} \\
&+ \hat{d}^I_{i,j} \hat{a} \sum_{i<u<j} Q^M_{i+1,u} Q^{M1}_{u+1,j-1} \\
Q^M_{ij} =& \begin{cases} Q^M_{i+1,j}\hat{c} + Q^{M1}_{i,j} + \sum_{i<k<j} Q^P_{i,k} \hat{d}^E_{ik} Q^M_{k+1,j} \hat{b} \\ 0 \qquad\qquad \text{if} \quad i = n_1 + 1 \vee j = n_1 \end{cases} \\
Q^{M1}_{ij} =& \begin{cases} Q^P_{ij} \hat{d}^E_{ij} + Q^{M1}_{i,j-1} \hat{c} \\ 0 \qquad\qquad \text{if} \quad i = n_1 + 1 \vee j = n_1 \end{cases}
\end{aligned}
\tag{13}
$$

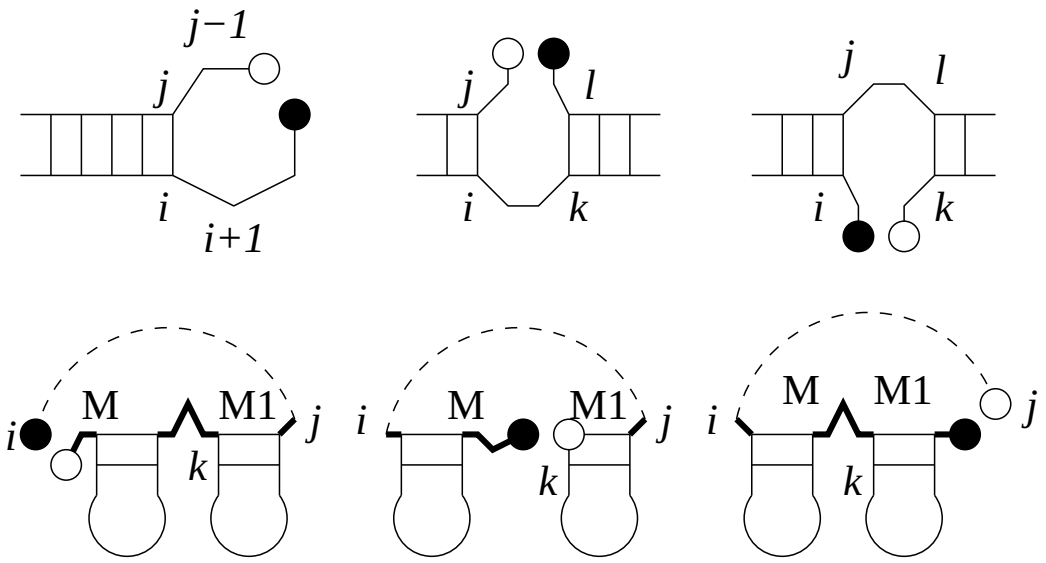


Figure 22: Loops with cuts have to be scored differently. Top row: hairpins and interior-loops containing the cut between n_1 (black ball) and $n_1 + 1$ (white ball).
Below: multi-loops containing the cut. Neither $M1$ nor M components must start at $n_1 + 1$ or stop at n_1. Note that the construction of Z^M out of Z^M and Z^{M1} ensures that the cut is not inside the loop part of Z^M either.

Upper alternatives refer to regular loops, lower alternatives to the loop containing the cutpoint. We have used the abbreviations
$\hat{\mathcal{H}}(i,j) = \exp(-\mathcal{H}(i,j)/RT)$,
and equivalently $\hat{\mathcal{I}}, \hat{d}, \hat{a}, \hat{b}, \hat{c}$, for the Boltzmann factors of the energy contributions.

Now we must apply E^I to all really dimeric structures, i.e. structures with intermolecular base pairs. Instead of doing that during the recursion, which would lead to the necessity to compute connected structures only, we can add this term in a post-processing step. The problem is to find all the intermolecular paired structures. This can be done by finding all structures that do **not** form intermolecular base pairs, and then subtract their contribution from the total partition function $Q(1,n)$. The structures that do not form intermolecular base pairs are combinations of structures of the two monomers. Their partition functions are $Q(1, c_p - 1)$ and $Q(c_p, n)$, so

$Q(1,n) - Q(1,c_p-1) * Q(c_p,n)$ is the partition function of all "true" dimers, i.e. of all structures having at least one intermolecular base pair. Thus, applying the initiation penalty boils down to:

$$Q^* = (Q(1,n) - Q(1,c_p-1) * Q(c_p,n)) \exp^{\frac{-E_I}{kT}} + Q(1,c_p-1) * Q(c_p,n) \quad (14)$$

4.5 Minimum free energy computation

The algorithm to compute the minimum free energy structure of two RNA molecules has been developed by Ivo Hofacker in [52]. However, the initiation penalty was not included. We changed the algorithm to incorporate this penalty, by using the approach described above – we added the penalty if there was at least one intermolecular base pair in the minimum free energy structure. This makes it possible that the "minimum" free energy structure is actually higher than the energy of the two disconnected minimum free energy structures, so we have to compare the result with this energy to get the real minimum.

Another problem may occur when computing homo-dimeric structures. If the "mfe" structure is symmetric, a penalty of $kT \log 2$ has to be added (see below). To ensure that no asymmetric structure has lower energy than this corrected structure, one possibility is to compute all suboptimal structures with $\epsilon = kT \log 2$ using `RNAsubopt` [137] and use the best asymmetric structure if it is within this energy band.

4.6 Suboptimal structures

Wuchty et al. [137] developed an algorithm to construct all possible secondary structures within a given energy band of the minimum free energy structure. We extended this algorithm to be able to create all possible secondary structures for two molecules. As we did not want to keep track about whether a substructure already has an intermolecular base pair or not, which is necessary to include the duplex initiation penalty, we again used a post processing step to add it. Basically, whenever a structure/energy pair is committed to

memory or put out, a small routine which adds the duplex initiation penalty if intermolecular base pairs are present is invoked.

4.7 Kinetics

In a biological system, it is reasonable to assume that the two molecules we are cofolding are not built simultaneously and/or at the same location. It is highly likely that they will have folded into intramolecular secondary structures before they get into contact with each other. Hence, the probability for them to get trapped in local intramolecular folding minima is big. On the other hand, if we consider kissing hairpin duplexes, these structures can act as "docking" structures, where the initial contact leads to a substantial refolding of the two molecules as the hairpins unzip and and intermolecular base pairs are built. The kissing hairpin duplexes are thus, while essential for the folding, not necessarily visible in the thermodynamic ensemble.
If we consider a basic move set containing only opening and closing of base pairs, as it is implemented e.g. in Kinfold [35], it would be reasonable to use the two mfe structures as starting structures for dimer kinetics. Thus, we need suboptimal structures with energies a little higher than the sum of the two mfes (the first intermolecular base pair can not take part in a stack, and the duplex initiation penalty has to be paid). These structures can in theory be computed using RNAsubopt. While this has been done for complexes of two small molecules (Christoph Flamm, personal communication), bigger molecules are difficult to deal with. There are simply too many suboptimal structures to compute. Thus, a different approach, limiting the number of computed structures while containing all necessary structures, is needed.

4.8 Testing the algorithms

To ensure that our algorithm produces the right results within our model, we used a two-fold approach. We used Wuchty's RNAsubopt to generate all possible secondary structures of the two molecules connected with a ploy-N

linker. The energy of these structures, after cutting out the linker, were then reevaluated according to our energy rules (including the initiation penalty). The result of this can be directly compared to the suboptimal cofold algorithm.

For small molecules ($\approx n \leq 30$), all possible substructures can be created using Wuchty's algorithm. We added up their Boltzmann factors to get the partition functions of these molecules, and compared them to the partition function version of `RNAcofold`. To check the pair probabilities, the Boltzmann weights for all structures containing base pair i, j were added up, divided by the partition function and the compared to the RNAcofold pair probabilities. It must be emphasized that while this approach does only work for small molecules, these molecules are big enough to be able to form all possible secondary structure elements (Loops) of our model, as the smallest possible molecule that can build a multi-loop is only 12 nucleotides long:
{ ((...)(...)) }.

For longer molecules, the only thing that can be done is using the equality $Q(A, B) = Q(B, A)$, so we swapped the molecules and compared the results of the Gibbs' free energy computation as well as the base pair probabilities.

4.9 Base pairing probabilities

McCaskill's algorithm [84] computes the base pairing probabilities from the partition functions of sub-sequences. It seems easier to first perform the backtracking recursions on the "raw" partition functions that do not take into account the initiation contribution and then use post processing steps for this contribution. This yields pairing probabilities P_{kl} for an ensemble of structures that does not distinguish between true dimers and isolated structures for A and B and ignores the initiation energy. McCaskill's backwards recursions are formally almost identical to the case of folding a single linear sequence. We only have to exclude multi-loop contributions in which the cut-point u between components coincides with the cut point c. All other

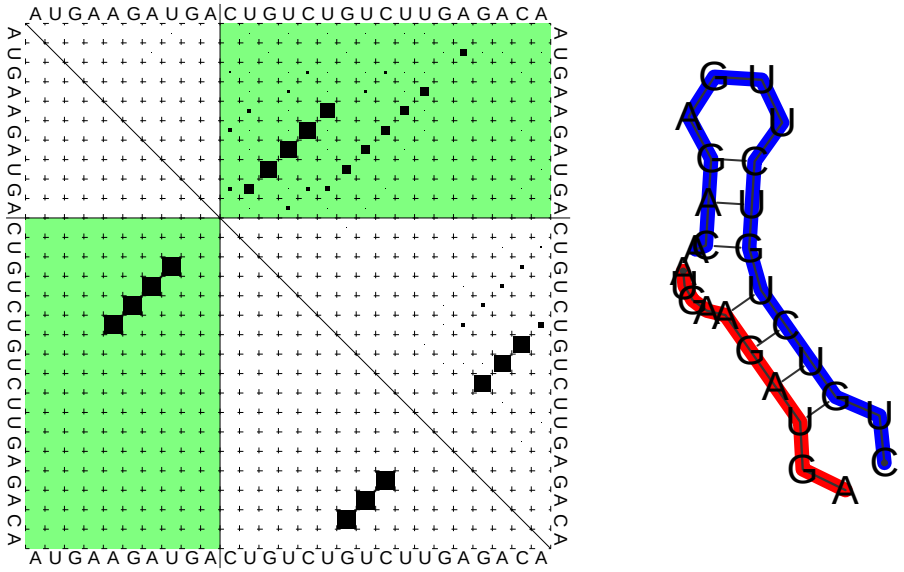


Figure 23: Dot plot (left) and mfe structure representation (right) of the cofolding structure of the two RNA molecules `AUGAAGAUGA` (red) and `CUGUCUGUCUUGAGACA` (blue).
Dot Plot: Upper right: Partition function. The area of the squares is proportional to the corresponding pair probabilities. Lower left: Minimum free energy structure. The two lines forming a cross indicate the cut point, intermolecular base pairs are depicted in the green upper right (partition function) and lower left (mfe) rectangle.

cases are already taken care of in the forward recursion. Thus:

$$\begin{aligned} P_{kl} =& \frac{Q_{1,k-1}Q^P_{k,l}Q^{l+1,n}}{Q_{1,n}} + \sum_{p<k;q>l} P_{pq}\frac{Q^P_{k,l}}{Q^P_{p,q}} \Bigg\{ \\ & \hat{\mathcal{I}}(p,q,k,l) \\ & + Q^M_{p+1,k-1}\hat{a}\hat{c}^{q-l-1} \\ & + Q^M_{l+1,q-1}\hat{a}\hat{c}^{k-p-1} \\ & + Q^M_{p+1,k-1}Q^M_{l+1,q-1}\hat{a} \qquad \Bigg\} \end{aligned} \tag{15}$$

The "raw" values of P_{ij}, which are computed without the initiation term, have to be corrected for this effect now. To this end, we separately run the backward recursion starting from $Q_{1,n}$ and from Q_{n_1+1,n_1+n_2} to obtain the base pairing probability matrices P^A_{ij} and $P^B_{n_1+i,n_1+j}$ for the isolated molecules. Note that equivalently, we could compute P^A_{ij} and P^B_{ij} directly using the partition function version of `RNAfold`.

In solution, the probability of an intermolecular base pair is proportional to the (concentration dependent) probability that a dimer is formed at all. Thus, it makes sense to consider the conditional pair probabilities given that a dimer is formed, or not. The fraction of structures without intermolecular pairs in our partition function Q (i.e. in the cofold model without initiation contributions) is Q^AQ^B/Q, and hence the fraction of true dimers is

$$p^* = 1 - \frac{Q^AQ^B}{Q}\,. \tag{16}$$

If we now consider a base pair (i,j), it can either arise from the dimeric or the monomeric state. If $i \in A$ and $j \in B$, it must arise from the dimeric state. If $i,j \in A$ or $i,j \in B$, however, it arises from the dimeric state with probability p^* and from the monomeric state with probability $1-p^*$. Thus the conditional pairing probabilities in the dimeric complexes can be

computed as

$$P'_{ij} = \frac{1}{p^*} \begin{cases} P_{ij} - (1-p^*)P^A_{ij} & \text{if } i,j \in A \\ P_{ij} - (1-p^*)P^B_{ij} & \text{if } i,j \in B \\ P_{ij} & \text{otherwise} \end{cases} \tag{17}$$

The fraction of monomeric and dimeric structures, however, cannot be directly computed from the above model. As we shall see below, the solution of this problem requires that we explicitly take the concentrations of RNAs into account.

4.10 Concentration dependence of RNA-RNA hybridization

As pointed out by Dimitrov and Zuker [23], the concentration of the two interacting RNAs as well as the possibility to form homo-dimers plays an important role and cannot be neglected when quantitative predictions on RNA-RNA binding are required. In our implementation of `RNAcofold` we therefore follow their approach and explicitly compute the concentration dependencies of the equilibrium ensemble in a mixture of two partially hybridizing RNA species.

If we consider a (dilute) solution of two nucleic acid sequences A and B with concentrations a and b, respectively, hybridization will yield a distribution of five molecular species: the two monomers A and B, the two homo-dimers AA and BB, and the heterodimer AB. In principle, of course, more complex oligomers might also arise, we will, however, neglect them in this approach. Considering higher order oligomers will complicate this computation significantly, and we will shortly get into it later on.
The presentation in this section closely follows a recent paper by Dimitrov [23], albeit slightly different definitions of the partitions functions are used here. The partition functions of the secondary structures of the monomeric states are Q^A and Q^B, respectively, as introduced in the previous section.

In contrast to [23], we include the unfolded states in these partition functions. The partition functions Q^{AA}, Q^{BB}, and Q^{AB}, which are the output of the `RNAcofold` algorithm (denoted Q in the previous section), include those states in which each monomer forms base-pairs only within itself as well as the unfolded monomers. We can now define

$$\begin{aligned} Q^{=}_{AA} &= Q^{AA} - (Q^{A})^2 \,, \\ Q^{=}_{BB} &= Q^{BB} - (Q^{B})^2 \,, \\ Q^{=}_{AB} &= Q^{AB} - (Q^{A}Q^{B}) \end{aligned} \tag{18}$$

as the partition functions restricted to the true dimer states, but neglecting the initiation energies Θ_I. An additional symmetry correction is needed in the case of the homo-dimers: A structure of a homo-dimer is symmetric if for any base pair (i, j) there exists a pair (i', j'), where i' (j') denotes the equivalent of position i in the other copy of the molecule. Such symmetric structures have a two-fold rotational symmetry that reduces their conformation space by a factor of 2, resulting in an entropic penalty of $\Delta G_{sym} = RT \ln 2$. On the other hand, since the recursion for the partition functions equ. 13 assumes two distinguishable molecules A and B, any asymmetric structures of a homo-dimer are in fact counted twice by the recursion. Leading to the same correction as for symmetric structures.
Since both the initiation energy Θ_I and the symmetry correction ΔG_{sym} are independent of the sequence length and composition, the thermodynamically correct partition functions for the three dimer species are given by

$$\begin{aligned} Q'_{AA} &= Q^{=}_{AA} \exp(-\Theta_I/RT)/2 \,, \\ Q'_{BB} &= Q^{=}_{BB} \exp(-\Theta_I/RT)/2 \,, \\ Q'_{AB} &= Q^{=}_{AB} \exp(-\Theta_I/RT) \,. \end{aligned} \tag{19}$$

From the partition functions we get the free energies of the dimer species, such as $F^{AB} = -RT \ln Q'_{AB}$, and the free energy of binding $\Delta F = F^{AB} - F^{A} - F^{B}$. We assume that pressure and volume are constant and that the solution is sufficiently dilute so that excluded volume effects can be neglected. The

many particle partition function for this system is therefore [23]

$$\begin{aligned}\mathcal{Q} = V^n \frac{a!b!}{n_A! n_B!\, 2\, n_{AA}!\, 2\, n_{BB}! n_{AB}!} \times \\ (Q'^A)^{n_A} (Q'^{AA})^{n_{AA}} (Q'^{AB})^{n_{AB}} (Q'^{BB})^{n_{BB}} (Q'^B)^{n_B}\end{aligned} \tag{20}$$

where $a = n_A + 2n_{AA} + n_{AB}$ is the total number of molecules of type A put into the solution (equivalently for b); n_A, n_B, n_{AA}, n_{BB}, n_{AB} are the particle numbers for the five different monomer and dimer species, V is the volume and n is the sum of the particle numbers. Eq 20 sums over all possibilities to arrange our A- and B-type molecules (of which we have a and b many, respectively) in our five molecular species and, within these, in all possible secondary structure states with Boltzmann weights.

The system now minimizes the free energy $-kT \ln \mathcal{Q}$, i.e., it maximizes $\mathcal{Q}$, by choosing the particle numbers optimally.

Following the discussion in [23], we use a saddle point approximation to evaluate $\mathcal{Q}$, i.e., we replace the sum by its largest term. Taking logarithms and Sterling's approximation $\ln n! = n \ln n - n$ for the factorials we obtain

$$\begin{aligned}\ln \mathcal{Q} \approx & a \ln a - a + b \ln b - b \\ & - n_A \ln n_A + n_A + n_A \ln Z'^A \\ & - n_B \ln n_B + n_B + n_B \ln Z'^B \\ & - n_{AB} \ln n_{AB} + n_{AB} + n_{AB} \ln Z'^{AB} \\ & - n_{AA} \ln n_{AA} + n_{AA} + n_{AA} \ln Z'^{AA} \\ & - n_{BB} \ln n_{BB} + n_{BB} + n_{BB} \ln Z'^{BB} \\ & + L(n_A + 2n_{AA} + n_{AB} - a) \\ & + M(n_B + 2n_{BB} + n_{AB} - b)\end{aligned} \tag{21}$$

after introducing the two Lagrange multipliers L and M to represent the two constraints on the particle numbers. The saddle point equations for this

system read

$$\begin{aligned} 0 &= \ln(Z'^{A}/n_A) + L \\ 0 &= \ln(Z'^{AA}/n_{AA}) + 2L \\ 0 &= \ln(Z'^{AB}/n_{AB}) + L + M \\ 0 &= \ln(Z'^{BB}/n_{BB}) + 2M \end{aligned} \tag{22}$$

and reduce to three balance equations

$$\begin{aligned} (Z'^{A})^2/n_A^2 &= Z'^{AA}/n_{AA} \\ (Z'^{B})^2/n_B^2 &= Z'^{BB}/n_{BB} \\ Q'^{A}/n_A \times Q'^{B}/n_B &= Q'^{AB}/n_{AB} \end{aligned} \tag{23}$$

by eliminating the Lagrange multipliers L and M. Introducing the equilibrium constants

As in [23], the dimer concentrations are therefore determined by the mass action equilibria:

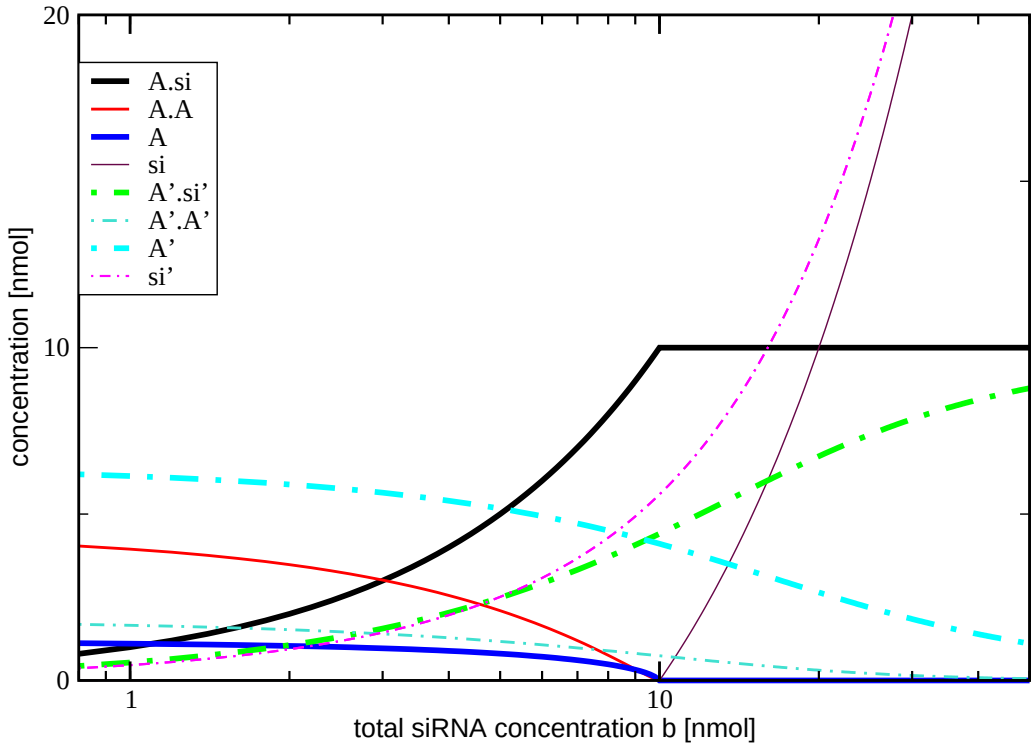
$$\begin{aligned} [AA] &= K_{AA}\,[A]^2 \\ [BB] &= K_{BB}\,[B]^2 \\ [AB] &= K_{AB}\,[A]\,[B] \end{aligned} \tag{24}$$

with

$$\begin{aligned} K_{AA} &= \frac{Z'^{AA}}{(Q^A)^2} = \frac{(Q^{AA} - (Q^A)^2)e^{-\Theta_I/RT}/2}{(Z_A)^2} \\ &= \frac{1}{2}\,e^{-\Theta_I/RT}\left(\frac{Q^{AA}}{(Q^A)^2} - 1\right) \\ K_{BB} &= \frac{1}{2}\,e^{-\Theta_I/RT}\left(\frac{Q^{BB}}{(Q^B)^2} - 1\right) \\ K_{AB} &= e^{-\Theta_I/RT}\left(\frac{Q^{AB}}{Q^A Q^B} - 1\right) \end{aligned} \tag{25}$$

Concentrations in equ.(24) are in mol/l.

Note, however, that the equilibrium constants in equ.(25) are computed from a different microscopic model than in [23], which in particular also includes internal base pairs within the dimers.



Binding energies: $\Delta F(A) = -24.53\text{kcal/mol}$
$\Delta F(A') = -11.76\text{kcal/mol}.$

Binding energies: $\Delta F(A) = -24.53\text{kcal/mol}$
$\Delta F(A') = -11.76\text{kcal/mol}.$

Figure 24: Example for the concentration dependency for two mRNA-siRNA binding experiments. In [116], Schubert *et al.* designed several mRNAs with identical target sites for an siRNA *si*, which are located in different secondary structures. In variant A, the *VR1 straight* mRNA, the binding site is unpaired, while in the mutant mRNA *VR1 HP5-11*, A', only 11 bases remain unpaired. We assume an mRNA concentration of $a = 10$ nmol/l for both experiments. Despite the similar binding pattern, the binding energies ($\Delta F = F^{AB} - F^A - F^B$) differ dramatically. In [116], the authors observed 10% expression for *VR1 straight*, and 30% expression for the HP5-11 mutant. Our calculation shows that even if siRNA is added in excess, a large fraction of the *VR1 HP5-11* mRNA remains unbound.

Together with the constraints on particle numbers, equ.(24) forms a complete set of equations to determine $x = [A]$ and $y = [B]$ from a and b by solving the resulting quadratic equation in two variables:

$$\begin{aligned} 0 &= f(x,y) := x + K_{AB}xy + 2K_{AA}x^2 - a \\ 0 &= g(x,y) := y + K_{AB}xy + 2K_{BB}y^2 - b \end{aligned} \tag{26}$$

The Jacobian

$$\begin{aligned} \mathbf{J}(x,y) &= \begin{pmatrix} \partial f/\partial x & \partial f/\partial y \\ \partial g/\partial x & \partial g/\partial y \end{pmatrix} \\ &= \begin{pmatrix} 1 + K_{AB}y + 4K_{AA}x & K_{AB}x \\ K_{AB}y & 1 + K_{AB}x + 4K_{BB}y \end{pmatrix} \end{aligned} \tag{27}$$

of this system is strictly positive and diagonally dominated, and hence invertible on $\mathbb{R}^+ \times \mathbb{R}^+$. Furthermore f and g are thrice continuously differentiable on $\mathbb{B} = [0,a] \times [0,b]$ and we know (because of mass conservation and the finiteness of the equilibrium constants) that the solution $(\hat{x}, \hat{y})$ is contained in the interior of the rectangle $\mathbb{B}$. Newton's iteration method

$$\begin{aligned} x' &= x + \frac{g(x,y)\partial_y f(x,y) - f(x,y)\partial_y g(x,y)}{\Delta} \\ y' &= y + \frac{f(x,y)\partial_x g(x,y) - g(x,y)\partial_x f(x,y)}{\Delta} \\ \Delta &= \partial_x f(x,y)\partial_y g(x,y) - \partial_y f(x,y)\partial_x g(x,y) \\ &= \det \mathbf{J} \end{aligned} \tag{28}$$

thus converges (at least) quadratically [118, 5.4.2]. We use (a,b) as initial values for the iteration.

4.11 Implementation and performance

The algorithm is implemented in ANSI C, and is distributed as part of the of the *Vienna RNA* package. The resource requirements of `RNAcofold` and `RNAfold` are theoretically the same: both require $\mathcal{O}(n^3)$ CPU time and $\mathcal{O}(n^2)$ memory. In practice, however, keeping track of the cut makes the evaluation of the loop energies much more expensive and increases the CPU time

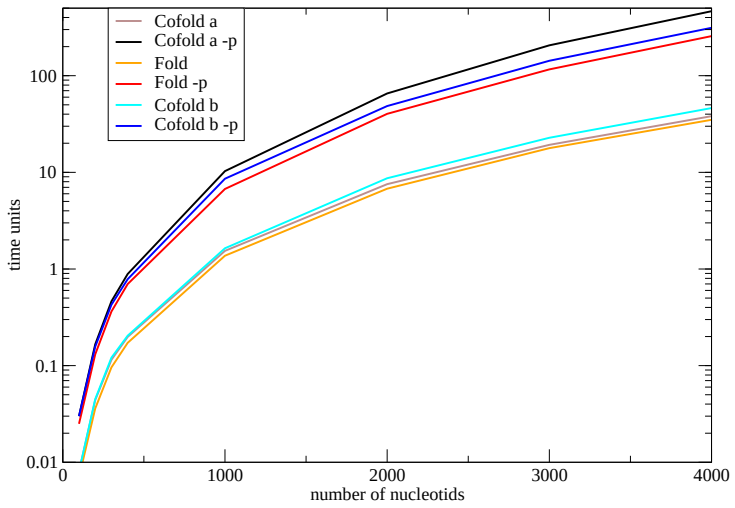


Figure 25: Performance Comparison of `RNAcofold` and `RNAfold`. 2 random molecules and their concatenation were folded. The two `RNAcofold` variants **a** and **b** correspond to two molecules of equal length and one molecule of length 20nt and a big one, resp. As can be seen, while `RNAfold` is always faster than `RNAcofold`, the worst results are seen in partition function folding (including pair probability computation) if two molecules share equal length.

requirements by a significant number. This can be illustrated by the fact that when the cut is and the end of the molecule, e.g. in the case where a miRNA is folded together with its target mRNA, `RNAcofold` is significantly faster compared to folding the same number of nucleotides in two molecules of equal length: there are much more loops containing the cut in the second case (Fig 25).

The base pairing probabilities are represented as a *dot plot* in which squares with an area proportional to P_{ij} represent the raw pairing probabilities, see Fig. 23. The dot plot is provided as Postscript file which is structured in such a way that the raw data can be easily recovered explicitly. `RNAcofold` also computes a table of monomer and dimer concentrations dependent on a set of user supplied initial conditions. This feature can readily be used to investigate the concentration dependence of RNA-RNA hybridization, see Fig. 24 for an example.

4.12 DNA

Like `RNAfold`, `RNAcofold` can be used to compute *DNA dimers* by replacing the RNA parameter set by a suitable set of DNA parameters. At present, the computation of DNA-RNA heterodimers is not supported. This would not only require a complete set of DNA-RNA parameters (stacking energies are available [135], but we are not aware of a complete set of loop energies) but also further complicate the evaluation of the loop energy contributions since pure RNA and pure DNA loops will have to be distinguished from mixed RNA-DNA loops.

Basically, there are two approaches to do that. The conceptionally easiest, and also more versatile way is to distinguish between RNA and DNA bases inside the algorithm by assigning different symbols/numbers. Instead of the 6 different base pairs, we would have 24. If we use $\dashv, \rfloor, \}, \sqcup$ as DNA bases, the set of possible base pairs B is:

$$\begin{aligned} B = \quad & \{AU, GU, GC, CG, UA, UG, at, gt, gc, cg, tg, ta, \\ & aU, gU, gC, tA, CtG, cG, At, Gt, Gc, Ua, Ug, Cg\} \end{aligned}$$

This makes the parameter files and the memory needed to store energy contributions bigger, but the effort is negligible compared to the memory requirements of the program as a whole. An advantage of this approach would be the possibility to compute the secondary structure of co-polymers of DNA and RNA nucleotides.

The second approach is to always use two different molecule types, and simply assign all bases smaller than c to be RNA bases and all others DNA bases. When evaluating the energies of the loops, if $i < c \leq j$ the mixed terms are used, if $j < c$ the RNA terms and if $c \leq i$ the DNA terms. While this approach is not as versatile, as co-polymers cannot be handled, it does not need as many different parameters. While the memory requirements for storing parameters are negligible, they must be somehow measured before we can use them. If less parameters are needed, this is cheaper and faster to do.

For both approaches, the changes to the algorithm would mainly apply to the energy evaluation parts and the setting of parameters, while the main routines do not have to be changed.

4.13 Applications

Intermolecular binding of RNA molecules is important in a broad spectrum of cases, ranging from mRNA accessibility to siRNA or miRNA binding, RNA probe design, or designing RNA openers [44]. An important question that arises repeatedly is to explain differences in RNA-RNA binding between seemingly very similar or even identical binding sites. As demonstrated e.g. in [24, 25, 85, 92], different RNA secondary structure of the target molecule can have dramatic effects on binding affinities even if the sequence of the binding site is identical.
Since the comparison of base pairing patterns is a crucial step in such investigations we provide a tool for graphically comparing two dot plots, see Fig. 26. It is written in Perl-Tk and takes two *dot plot* files and, optionally, an alignment file as input. The differences between the two *dot plots* are displayed in color-code, the *dot plot* is zoomable and the identity and probability(-difference) of a base pair is displayed when a box is clicked. A non interactive version where the probability differences are also color-coded is shown here and used henceforth in this chapter. As an example for the applicability of `RNAcofold`, we re-evaluate here parts of a recent study by Doench and Sharp [28]. In this work, the influence of GU base pairs on the effectivity of translation attenuation by miRNAs is assayed by mutating binding sites and comparing attenuation effectivity to wild type binding sites.

They used a construct containing 4 binding sites, where the inner two were mutated and the outer two always stayed wild type binding sites. We focus here on mutations that introduced GU base pairs. There were 3 sites within the binding site where a mutation of A to U was performed. on the mRNA.

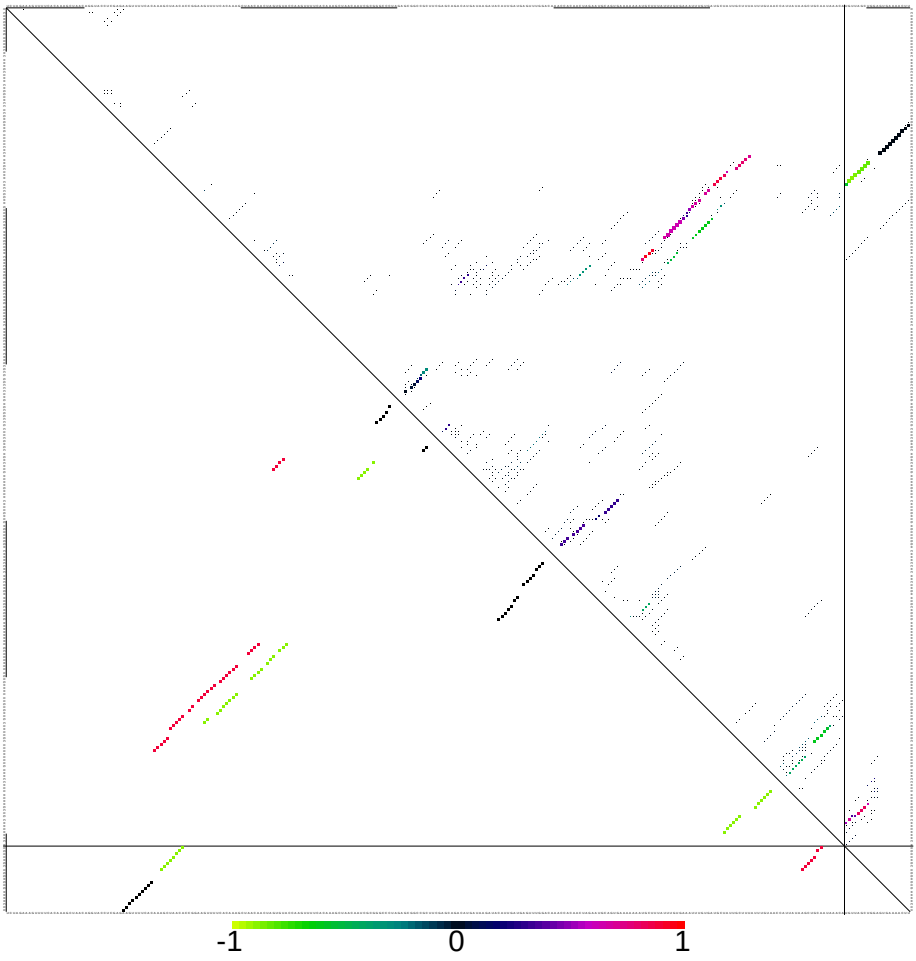


Figure 26: Difference dot Plot of native and mutated secondary structure of a 3GU mutation of the CXCR4 siRNA gene.
The color of the dots encodes the difference of the pair probabilities in the two molecules. For example, red squares denote pairs much more probable in the second molecule, and green ones base pairs that have a much higher probability in the first molecule (see color bar).
The area of the dots is proportional to the larger of the two pair probabilities.

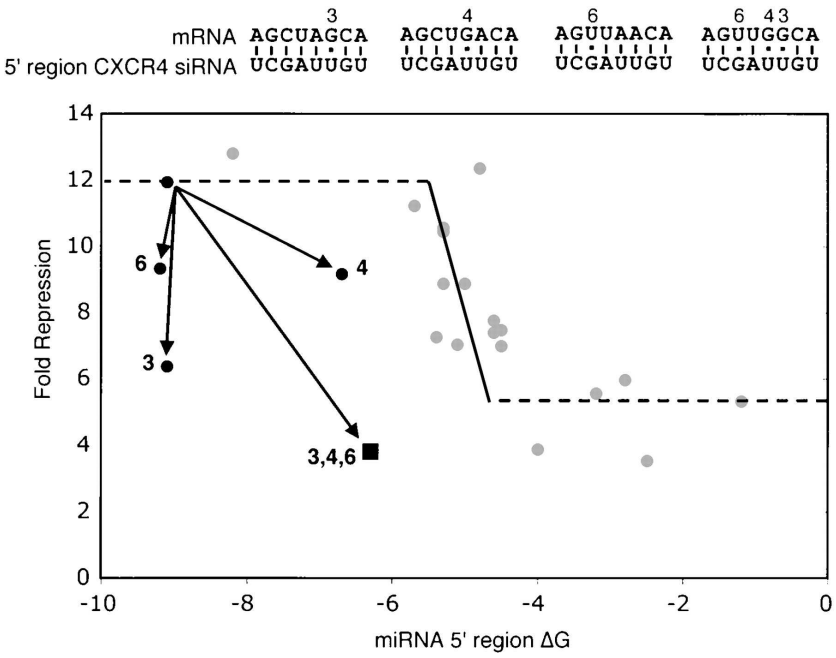


Figure 27: Repression of different GU constructs, figure taken from [28]

The resulting constructs did not repress translation as well as the wild type constructs, although the binding energy stayed almost the same. The worst repression was seen when all 3 mutations where made simultaneously. Their results are shown in Fig. 27

While Doench and Sharp concluded that miRNA binding sites are not functional because of the GU base pairs, testing the dimers with `RNAcofold` shows that there is also a significant difference in the cofolding structure that might account for the activity difference without invoking sequence specificities: Because of the secondary structure of the target, the binding at the 5' end of the miRNA is weaker than in the wild type, Fig. 28. The figures were computed in the following way: as we cannot compute pentamers using `RNAcofold`, we decided to "mask" the wild type binding sites by using constrained folding, i.e. we did not permit any base pairing for these sites, thus mimicking already bound miRNAs.

When comparing the mutated to the wild type binding, one can first say that

the binding energy is a little lower for the mutated cases, ($\Delta G = -12.9$ in the wild type case, $\Delta G = -12.3$ in case 4, $\Delta G = -11.3$ in case 3, $\Delta G = -10.7$ in case 6 and $\Delta G = -10.31$ in case 346, all in kcal/mol). This can indeed not explain the different behavior. However, if we look at the binding pattern, we see that, while the "best" GU mutations (4,6) retain the binding of the wild type, the worse ones (3,346) show a significantly lower binding of the 5' part of the miRNA. So, while we are not able to fully explain the differences in function using binding pattern and energy, we can at least explain why some GU mutations are even worse than others.

In a similar experimental setting, Brennecke et al. [12] tried to figure out the minimum requirements for miRNA-target recognition. They expressed a miRNA targeting EGFP with various target sites at its 3' UTR. Their results indicate that introducing GU base pairs destroys function, much as Doench's do (Fig. 29).

We computed the cofolded structures of the dimers and again found that while we cannot explain the diminished functions, we can easily explain the loss of function by introduced GU pairs through a very different binding pattern at the 5' part of the miRNA.

4.14 The `RNAup` approach

Because `RNAcofold` can not predict intermolecular pseudo knots, additional approaches are needed. An algorithm including pseudo knots (without so-called entanglers, which make the problem NP-hard) was presented by Pervouchin [103]. While this algorithm can predict almost all possible secondary structures, it is $\mathcal{O}(n^3m^3)$? in processor time (with n the length of the first and m the length of the second molecule). It is thus not suited for predicting structures for long molecules.

A different approach was taken in `RNAup` [92]. Here, the building of a dimer is split into two steps: the first is the opening of a binding site, which costs energy, the second is the formation of the dimer, where energy is gained. At the moment, this is only implemented for cases where a smaller

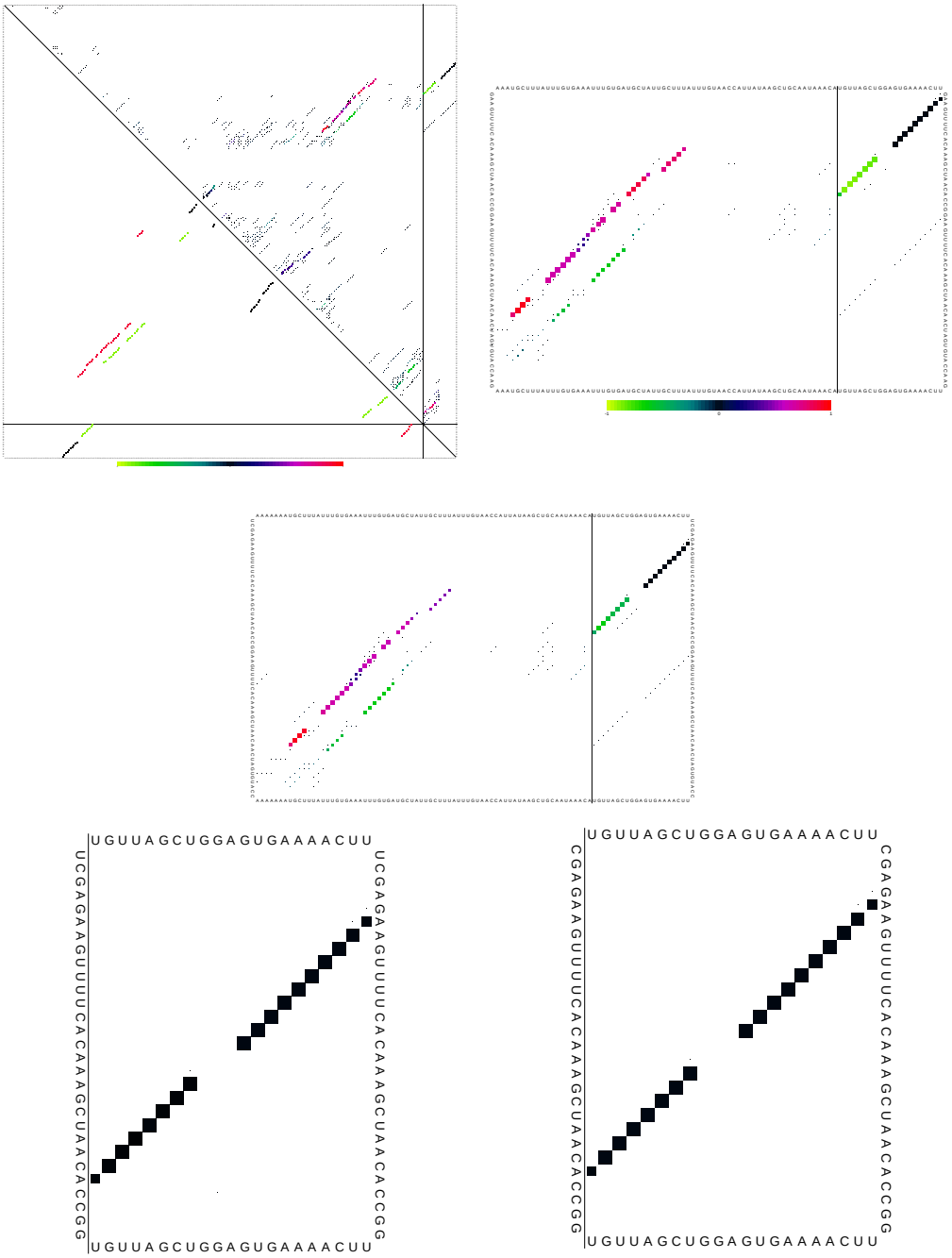
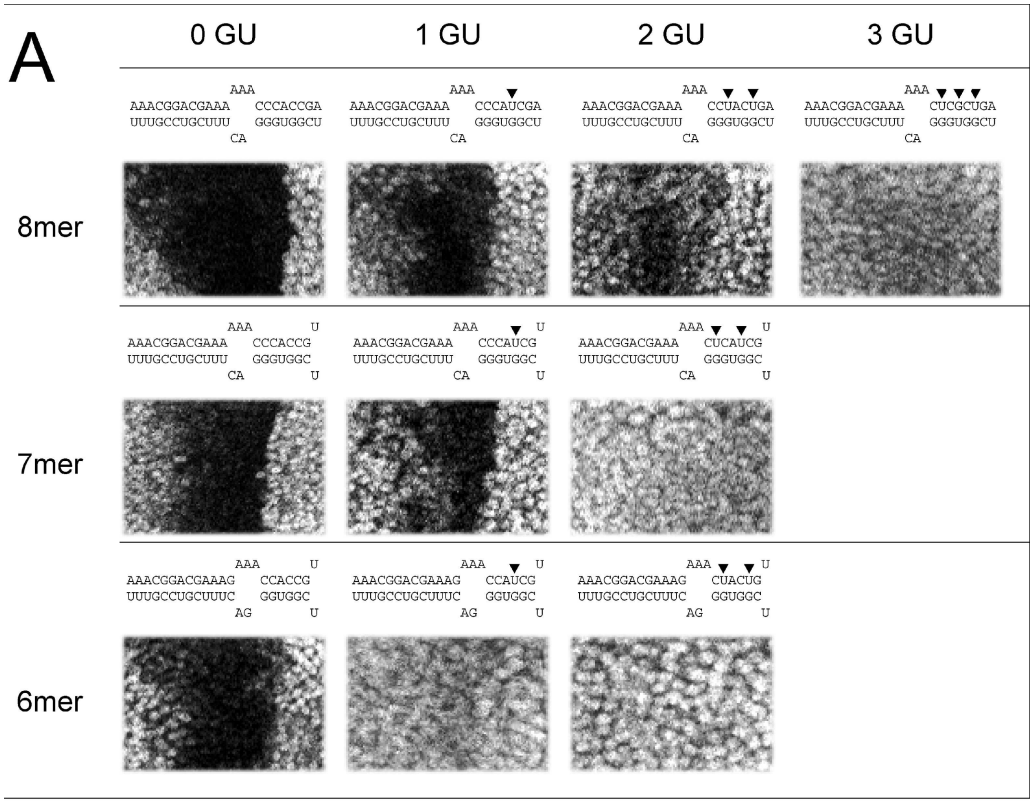


Figure 28: Difference dot plots of the 4 GU mutants of [28] against the original binding site.

Top: left: 346 mutant. right: zoom into the binding site, as can be seen, an extended stem in the mRNA inhibits binding of the 5' end of the miRNA.

Middle: 3 mutant, binding site. The situation is similar to the 346 mutant

Bottom: left 6 mutant, right 4 mutant, binding sites.



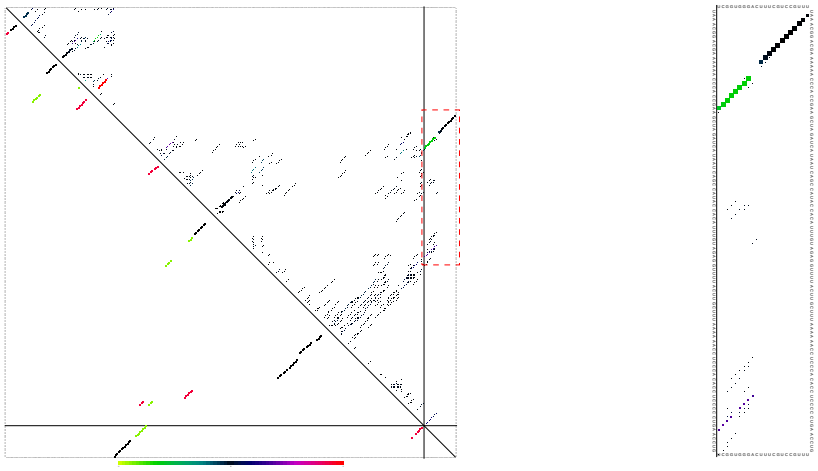


Figure 29: Results of target 3'UTR mutations to get GU base pairs, top Fig. taken from [12]. Attenuation of EGFP results in the black stripes in the center of the pictures. The bigger the black area, the better the miRNA function. From left to right, more GU mutations were introduced.

Bottom: Difference dot plot of the 3GU mutation against the zero GU version. On the left the whole dot plot, on the right a detailed view of the binding site (boxed) is shown. As can be seen, the seed region of the siRNA does bind much weaker in the 3 GU mutant, and a different binding site is populated.

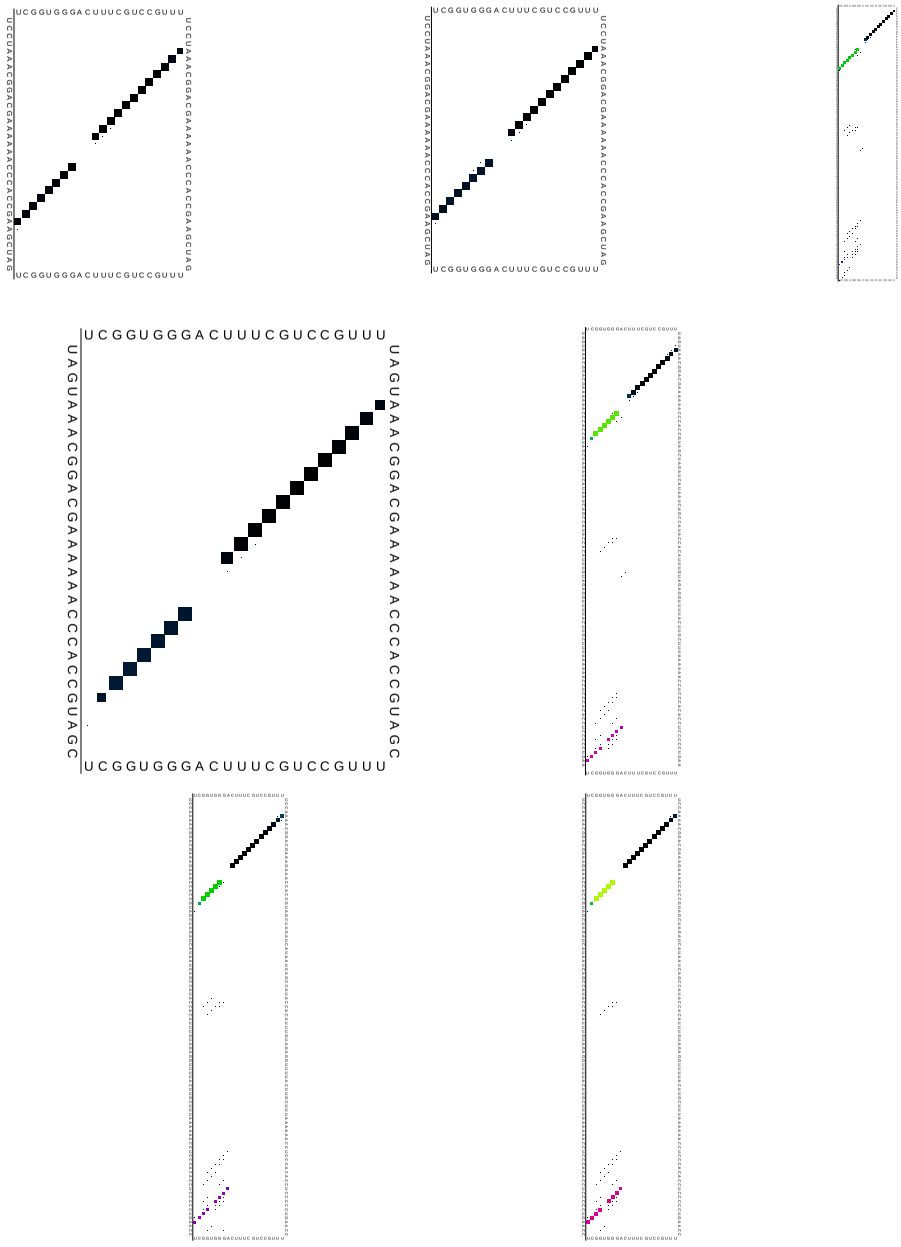


Figure 30: Difference dot plots of the GU mutants of [12]
First row: 8mer, difference of 0 GU to 1, 2, 3 GU (from left)
Second row: 7mer. difference of 0 GU to 1 and 2 GU (from left)
Third row: 6mer. difference of 0 GU to 1 and 2 GU (from left).
If compared with Fig 29, all loss of function mutants have a different binding pattern at the 5' site of the miRNA.

molecule binds to a bigger one, as it was mainly designed for the analysis of miRNA(siRNA)/mRNA interactions. However, the application of the approach to two molecules of arbitrary size is possible.

The program uses the `ViennaRNA` library to compute the probability of a stretch of s bases to be unpaired in the ensemble, where s is usually chosen a little larger (typically 10 nt) than the smaller molecule. This restricts the maximal length of the binding site to s. It then computes the partition function of an intermolecular duplex in this stretch and combines all this information. The output is the probability that there is an intermolecular duplex at a certain stretch in the longer molecule, given that there exists a duplex. (i.e. if a duplex is formed, where will the intermolecular basepairs be).

To get the probability that a stretch of s bases is unpaired, we have to compute the respective partition function. We again split the computation, into contributions where s is enclosed by a base pair and contributions where it is not enclosed by a basepair. If s starts at nucleotide x_s, the probability that s is unpaired and not included by a basepair is

$$p_e^u(x_s, s) = \frac{Q(1, x_s - 1)Q(x_s + s - 1, n)}{Q(1, n)}, \tag{29}$$

the product of the partition functions outside s. As the partition function for s to be unpaired is 1, we neglect this term here. To get the terms where s is included by a base pair i, j, we have to sum up the partition functions that i, j closes a hairpin-loop, interior-loop or a multi-loop up over all $i < x_s$ and $x_s + s \leq j$.

$$p_h^u(x_s, s) = \frac{1}{Q(1, n)} \sum_{i<x_s, x_s+s\leq j} \hat{Q}^B(i, j) \tag{30}$$

Without invoking the limitation of the loop size, the interior-loop contribu-

tions look like this:

$$p_i^u(x_s, s) = \frac{1}{Q(1,n)} \sum_{i<x_s, x_s+s\leq j} \hat{Q}^B(i,j) \quad \sum_{i<k<l<x_s} \quad Q^B(k,l) + \\ + \quad \sum_{x_s+s\leq k<l<j} \quad Q^B(k,l)$$

For the multi-loop contributions, we have to compute $Q^{M2}(i,j)$, the partition function that there are at least 2 components of a multi-loop between i and j.

$$Q^{M2}(i,j) = \sum_{i<k<j} Q^M(i,k) Q^{M1}(k+1,j) \tag{31}$$

with this, we can compute

$$p_m^u(x_s, s)s = \frac{1}{Q(1,n)} \sum_{i<x_s, x_s+s\leq j} \hat{Q}^B(i,j) \left(Q^{M2}(i+1, x_s-1) + \right. \\ + \quad Q^{M2}(x_s+s-1, j-1) + \\ + \quad \left. Q^M(i+1, x_s-1) Q^M(x_s+s-1, j-1)\right),$$

the probabilities that s is 5' of all components, 3' of all components or between the components of the multi-loop, respectively. Combination of these terms then lead to the total probability that a stretch s is unpaired. As there may be different affinities for binding into multi-loops, interior-loops, hairpin-loops and exterior-loops, the single terms can be accessed separately.

This approach can deal with intermolecular pseudoknots that bind into a hairpin, interior-loop or multi-loop. However, the strictly additive model used for the energy computation of pseudo knots is probably too simplifying. Sterical considerations have to be taken into account, e.g. whether it is possible to have an intermolecular base pair directly neighboring an intramolecular one. Another question is as the binding leads to a helical formation, how long the helical intermolecular part can be before building such a substructure is either sterically impossible or would at least require the molecules to first unfold themselves substantially, which is kinetically unfavorable.

The drawback of this approach is that binding sites can not be split, and the base pair probabilities are not computed (which is possible in principle). As the intermolecular pseudo knots are not included in the RNAcofold approach, but split binding sites are, and `RNAup` is $\mathcal{O}(n)^3 s^2$, the combination of the two approaches is an obvious approach to get better results.

4.15 Combination of `RNAcofold` and `RNAup`

RNAup uses the arrays created by the partition function version of `RNAfold` to compute the probabilities to be unpaired (p^u). There are two main possibilities if we consider whether a stretch of s bases (from v to w) is unpaired in a secondary structure. The first possibility is that no base pair i, j spans vw, i.e. if $i, j \in B \Rightarrow j < v \vee i > w$. As these cases do not lead to a pseudoknotted structure, they are already taken into account by `RNAcofold`, so they will not concern us here. The other possibility is that some basepair i, j encloses vw. If a base between v and w takes part in an intermolecular base pair, this will lead to a pseudoknot. These are the cases we have to add to the `RNAcofold` approach.

If we consider the innermost base pair that encloses vw, this base pair can either close a hairpin, an interior- or a multi-loop. The hairpin case does not split up any further, while in the interior-loop case, the unpaired stretch vw can either be 5' of the interior base pair k, l ($i < w < k < j$) or 3' of it. Accordingly, there are 3 cases for multi-loops: either the unpaired stretch is 5' of all "inner" base pairs of the multi-loop (i,e, there is no component 5' of vw, it is 3' of all components or it is in between components.In section 5, the exact computations of the unpaired probabilities (or their partition functions) is explained. For now, it is sufficient to know that almost all we need to compute these is computed by `RNAcofold` already. The only arrays needed additionally are outer partition functions of single molecules, which are computed separately in exactly the way they are in `RNAfold`. To unify `RNAcofold` with `RNAup`, a stepwise approach is used:

1. Compute partition function version of `RNAcofold`

2. Compute outer partition functions for long molecules
3. Compute duplex partition functions
4. Integrate duplex energies with Q^Bs
5. Compute pseudo knotted partition function
6. Compute total partition function as sum off the pseudo knotted and the `RNAcofold` partition functions.
7. Compute the outer partition functions given a duplex creating a pseudoknot.
8. Update pair probabilities for base pairs

The duplex partition functions are computed analogous to `RNAup`. The values computed, $q^d(i,j)$, are partition functions subject to the constraint that there is at least one intermolecular base pair between a base of the sub-sequence $i...j$ of the first molecule and the second molecule. For computational convenience, the partition function of the second (smaller) molecule will be totally included into this term. The duplex analogons of the usual arrays are then computed using these q^ds. The Feynman diagrams (Fig. 31)show the forward recursion of $Q_d^B(i,j)$, the partition function between i and j given that i,j form a base pair and there are intermolecular base pairs somewhere between i and j.
The recursion for the pseudo knotted partition function Q_d then looks like this:

$$Q_d(i,j) = Q_d(i+1,j) + \sum_k Q_d^B(i,k)Q(k+1,j) + Q^B(i,k)Q_d(k+1,j) \quad (32)$$

The outer partition functions $\hat{Q}_d^B$ are computed in the same way.

Preliminary results

As the explained algorithm is expansive to compute, it is preferable to compute `RNAcofold` for most purposes. In most cases, the usage of the unified

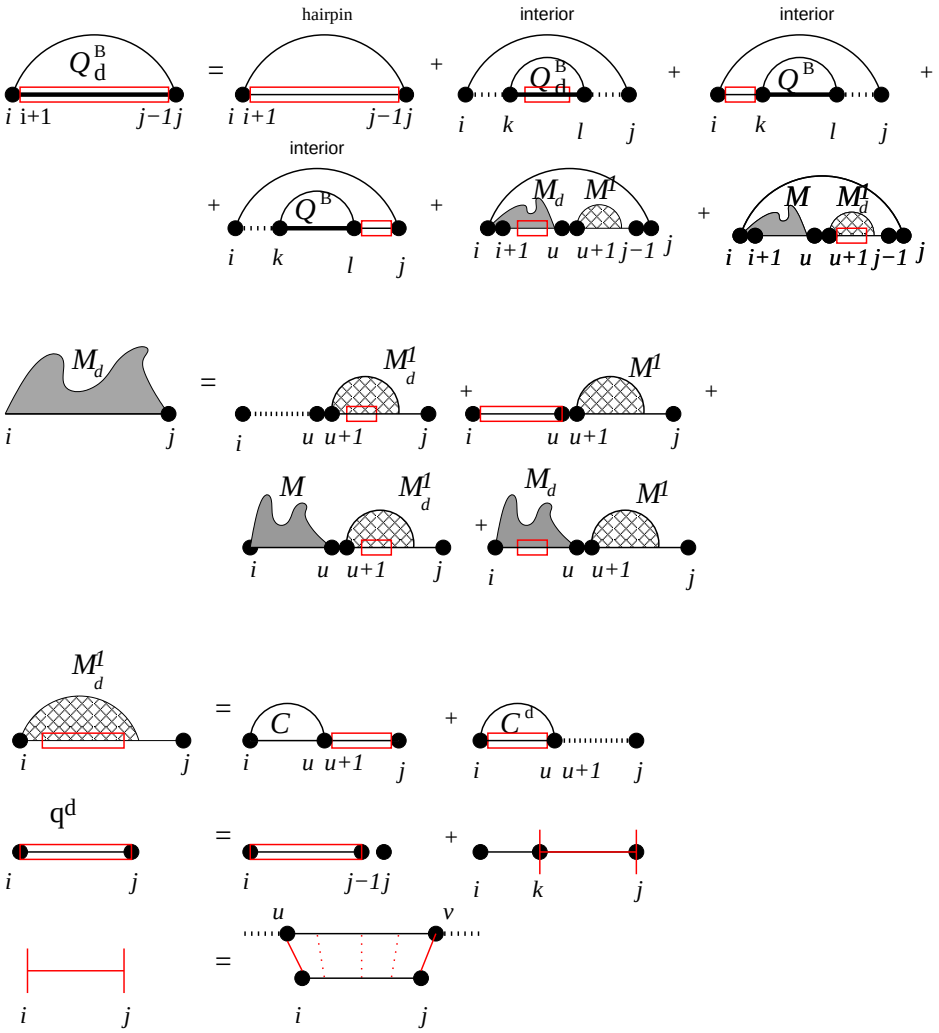


Figure 31: Feynman diagrams of the recursion to compute $Q_d^B(i,j)$. Note that in the last row, i and j have to be the outermost bases that pair intramolecularly.

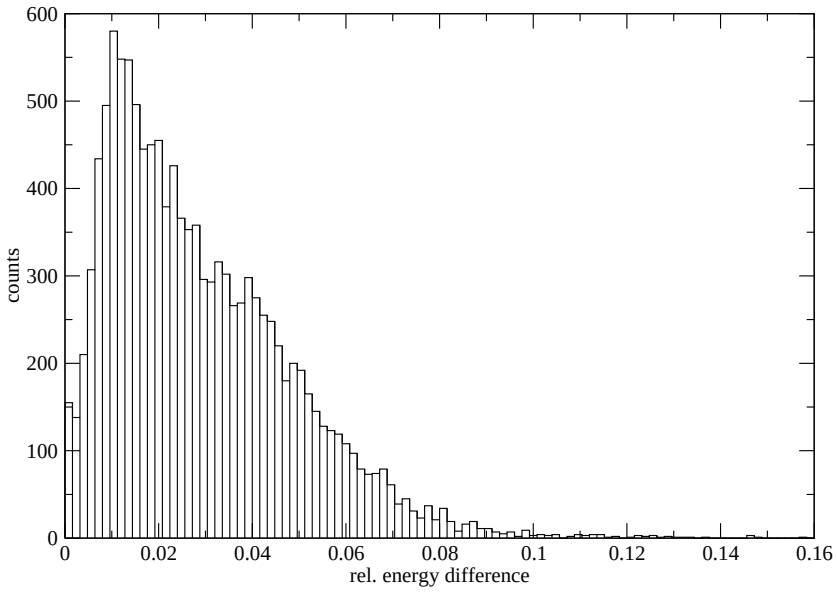


Figure 32: Histogram of the relative differences of the Gibbs' free energies of `RNAcofold` and Unified computed dimers. The mean relative difference is 0.029. As data set, 12403 mRNA/siRNA pairs generated by Novartis were used.

algorithm is not really necessary, as the differences of the results are small (Fig. 32). However, sometimes there are big differences in the probabilities of the base pairs.

4.16 Folding more than two molecules

We have described here an algorithm to compute the partition function of the secondary structure of RNA dimers and to model in detail the thermodynamics of a mixture of two RNA species. At present, `RNAcofold` implements the most sophisticated method for modeling the interactions of two (large) RNAs. Because the no-pseudoknot condition is enforced to limit computational costs, our approach disregards certain interaction structures that are known to be important, including e.g. kissing hairpin complexes.

The second limitation, which is of potential importance in particular in histochemical applications, is the restriction to dimeric complexes. More complex oligomers are likely to form in reality. The generalization of the present ap-

proach to trimers or tetramers is complicated by the fact that for more than two molecules the results of the calculation are not independent of the order of the concatenation any more, so that for M-mers $(M-1)!$ permutations have to be considered separately. This also leads to bookkeeping problems since every secondary structure still has to be counted exactly once. However, Dirks et al. [26] have elegantly solved that problem of computing only fully connected substructures. They also gave a proof by induction for the problem of the permutations, i.e. that no secondary structure must be counted twice when appearing in two permutations. As a matter of fact, for fully connected structures (i.e. there is no molecule which is not connected to any other) this can not happen: For three molecules A, B, C the argument can be clarified like this: If the molecules are fully connected, there has to be either base pairs i_A, j_B and k_B, l_C (we will call that case 1) or i_A, j_C and k_B, l_C (case 2). It can easily be seen that for the concatenated molecule, the inequality

$$i_A < j_B < k_B < l_C$$

must hold. If molecules B and C are now swapped (to get the only different circular permutation for three molecules), then the inequality is

$$i_A < l_C < j_B < k_B,$$

which means there is a pseudoknot. For case two, the inequality is

$$i_A < k_B < l_C < j_C$$

and the swapped one is

$$i_A < l_C < j_C < k_B$$

which is again a pseudoknot. Since there are no other possibilities to fully connect three molecules, and this is also the case in principle for n molecules, there cannot be two fully connected structures in two different permutations that have the same secondary structure.

As opposed to Dirks et al., we compute all possible secondary structures for all permutations (i.e. also not fully connected ones), and then proceed to

subtract all not fully connected ones that appear more than once. As these are post-processing steps rather than changes in the algorithm, they can be neglected as far as the performance is concerned. The only changes in the algorithm described above first concern the fact that there is more than one cut point, so we introduce an array of cut points as well as an array that holds on which molecule a certain base is situated. The second difference is in the computation of the exterior-loop contribution to $Q^B(i,j)$, where we had:

$$Q^B(i,j) += Q(i+1, c_p - 1)Q(c_p, j-1)$$

for two molecules. This has to be replaced by the sum over all contributions where there is at least one cut point not spanned by an intermolecular base pair. To compute this, we have to use the following ansatz:
We denote $Q_k^<(i,j)$ the set of all structures between i and j that have at least k unspanned cut points. These sets can be computed by dividing the partition functions into $k+1$ parts, i.e. for $k=2$, we have to compute all possibilities to use two cut points out of a set of cut points $C(ij)$ between i and j. As an example, for 4 cut points, the partition function will be computed as follows:

$$Q_k^<(i,j) = \sum_{c_i \le a \le c_j - 3} \sum_{a < b \le c_j - 2} \sum_{b < d \le c_j - 1} \sum_{d < e \le c_j} Q(i,a)Q(a,b)Q(b,d)Q(d,e)Q(e,j)$$

However, these computations will count all structures with more than k cut points multiple times. If we want to correct for that, we will have to consider how often a structure with l cut points is counted by computing $Q_k^<(i,j)$. It is easy to see that this will be the case exactly $\binom{l}{k}$ times. If we use

$$\sum_{1 \le k \le n} (-1)^{k+1} \binom{n}{k} = 1,$$

we can then compute the partition function $Q^g(i,j)$ that there is at least one unspanned cut point between i and j by:

$$Q^g(i,j) = \sum_{1 \le k \le m} (-1)^{k+1} Q_k^<(i,j)$$

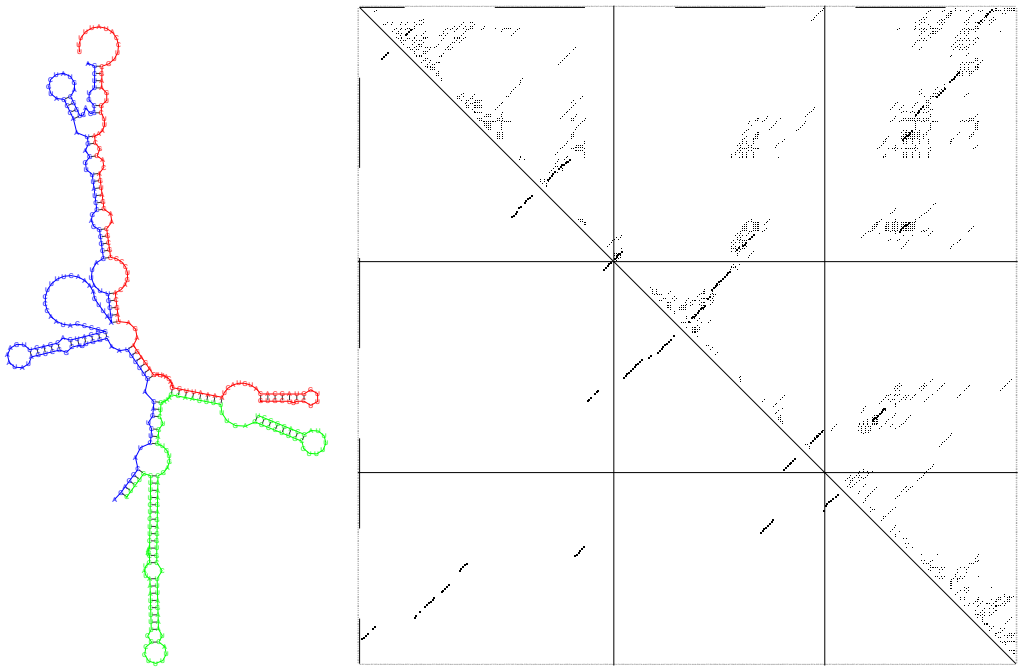


Figure 33: Trimeric structure of human U4, U5 and U6 spliceosomal RNAs.
The mfe secondary structure shown on the left has the order U4 (blue) U5 (green) U6 (red). Its mfe is -114.3 kcal/mol. (As opposed to -105.9 kcal/mol for the other order).
At the right, the dot plot is shown. (Free Energy:-123.8 kcal/mol, equilibrium constants: $K_{ABC} = 5 \cdot 10^{13}$, $K_{AB} = 1.4 \cdot 10^{10}$, $K_{BC} = 2.8 \cdot 10^{6}$, $K_{AC} = 1.1 \cdot 10^{8}$

where m is the number of cut points between i and j. This computations are $\mathcal{O}((n-1)^{n-1})$ for n the number of strands, but for small n, this is still feasible. So, the recursion for n molecules looks like:

The above strategy can also be used for the computation of all "true" n-mers. It gets increasingly complicated, however, so that we will instead implement the connected structures only variant of the algorithm.

5 The importance of folding local, RNAplfold

5.1 Local RNA secondary structures

There are two main reasons why one wants to compute a secondary structure for smaller parts of an RNA molecule. The first one is that the computation is faster. If you restrict the maximum span of a base pair (i.e. the maximum number of bases lying between the paired bases) to a number x, the time complexity of folding is reduced to $\mathcal{O}(nx^2)$, while the memory requirements reduce to $\mathcal{O}(n + x^2)$. This enables us to compute local substructures for chromosome length molecules, as time complexity and memory consumption are linear with respect to the length of the molecule.
The second reason is motivated by nature. It splits up in two parts: There are many structural motifs which are substructures of long molecules having an ill-defined secondary structure. Examples for this are the Internal Ribosome Entry Site (IRES), the Iron Response Element (IRE) or the SElenoCysteine Insertion Sequence (SECIS), which are all part of mature mRNAs. Viral examples are 5'SL and 3'SL structures in tick-borne encephalitis virus or HC structures at the 5' end of HCV.
Moreover, in many cases, like e.g. in micro RNAs, the exact boundaries of the transcript are not known, which makes it necessary to find local substructures. Indeed, the independence of the miRNA precursor structure of the exact boundaries of the transcript has been successfully used by Sewer et al. to identify new miRNAs [119].
A fact that makes computing local secondary structures useful is that not only is prediction accuracy inversely proportional to the base pair span (i.e. to $j - i$), but even in big molecules like 23 S RNA, the vast majority of base pairs has a span below 100 [29].

An algorithm to predict the local minimum free energy secondary structure of an RNA molecule is included in the`ViennaRNA` package [53]. In this program, `RNALfold`, the first and most important step is to restrict the maximum base

pair span to some l. This step leads to a performance gain from $\mathcal{O}(n^3)$ to $\mathcal{O}(nl^2)$. Then, using the usual recursions, an array $E^L(i)$ holding the energy of the sequence 3' of i is computed. To get locally stable structures, a backtrack is started whenever $E^L(i-1) \geq E^L(i) \wedge E^L(i) < E^L(i+1)$, i. e. adding another base 5' will not lead to a better energy while adding i did. As i can only be paired with bases $j \leq i + l$, the backtrack can be restricted to a stretch of the molecule of length l.

Computing the partition function of a locally folded RNA has been neglected up to now, however, in [53], Hofacker et al. gave a short introduction about how this could be done using the Nussinov Model.
When predicting base pair probabilities, we have the problem to define what exactly we mean by a "local pair probability". To do this, we first have to look at the so called sliding window approach.

5.2 Sliding windows

The favorite way to predict local structural features is using "windows" of a certain length that "slide" over a longer sequence. This means that you do your computations for small, overlapping parts of a molecule. An approach like this is used e.g. in RNAz [129], where alignments are split into overlapping blocks. The speed-up you can get is dependent on the overlap - the bigger it is, the slower the program will run. Furthermore, the computation time needed is dependent on the size of the "window" used. Again, the bigger the window, the smaller the performance gain.
The drawback of this approach is that you will loose any information that is not contained in one of the windows. This also applies to small structures if they are situated between windows — to get all possible information, the overlap will have to be maximal, the step-size (the distance between single windows) has to be 1. However, this will slow down the procedure, so that usually bigger step-sizes are used. (Default values of RNAz are window size: 120, step-size: 40).

5.3 Local pair probabilities

We define a local pair probability as the *average* probability of a base pair, where the averaging is done over all windows that can contain the base pair. To get maximum accuracy, we are forced to use a step-size of one (or maximum overlap).
Computing all possible windows may appear to be costly, but this is not the case. Because of the dynamic programming approach, we can simply add one base, say 3', and subtract one base on the other end (5') to get the partition function of the next window.

$$Q(i,j) = Q(i,j-1) + \sum_k Q_b(k,j)Q(i,k-1)$$

We call the probability of a base pair appearing in a window of size L, starting at position u, $p^{u,L}$. The computation of these probabilities is $\mathcal{O}(nL^3)$. However, what we are interested in are the average pair probabilities $\pi^L(i,j)$ to get the base pair i,j within a window of length L.

$$\pi^L(i,j) = \frac{1}{L-(j-i)+1} \sum_{u=j-L}^{i} p^{u,L}(i,j)\,. \tag{33}$$

For $i+L > n$ and $j-L < 1$, the edges of the molecule, there are fewer sequence windows, and the factor in equ. (33) has to be modified accordingly. As we will see below, we do not have to explicitly calculate the $p^{u,L}$s to compute the π^Ls. Therefore, the computation can be done in $\mathcal{O}(nL^2)$.

We divide the computation of the pair probabilities, using the "inner" probabilities $Q^B(i,j)$, which are independent of the window considered – as long as the base pair in question is part of that window. For the outer part, we distinguish the two cases that the base pair is enclosed by another one or that it is not. If it is not enclosed by another base pair, then we simply add up for all windows:

$$p_o^{u,L}(i,j) = \frac{Q(u,i-1)Q^B(i,j)Q(j+1,u+L)}{Q^{u,L}(u,u+L)}$$

with $Q^{u,L}$ the partition function for a window of size L, starting at u.
If the base pair is enclosed by another one, we get:

$$p_e^{u,L}(i,j) = \sum_{u<k<i,j<l\leq u+L} \frac{p^{u,L}(k,l)}{L-(l-k)+1} \left(\frac{\mathcal{I}(kl,ij) + M(k+1,i-1)}{Q^B(k,l)} + \frac{M(j+1,l-1)}{Q^B(k,l)} + \frac{M(k+1,i-1)M(j+1,l-1)}{Q^B(k,l)} \right)$$

the average pair probability then is:

$$\begin{aligned}
\pi^L(i,j) &= \frac{1}{L-(j-i)+1} \sum_{j-L\leq u}^{i} p_o^{u,L}(i,j) + p_e^{u,L}(i,j) = \\
&= \frac{1}{L-(j-i)+1} \sum_{j-L\leq u}^{i} \frac{Q(u,i-1)Q^B(i,j)Q(j+1,u+L)}{Q^{u,L}(u,u+L)} + \\
&+ \sum_{u<k<i,j<l\leq u+L} \frac{p^{u,L}(k,l)}{L-(l-k)+1} \frac{\mathcal{I}(kl,ij)}{Q^B(k,l)} \\
&+ \sum_{u<k<i,j<l\leq u+L} \frac{p^{u,L}(k,l)}{L-(l-k)+1} \frac{M(k+1,i-1)+M(j+1,l-1)}{Q^B(k,l)} \\
&+ \sum_{u<k<i,j<l\leq u+L} \frac{p^{u,L}(k,l)}{L-(l-k)+1} \frac{M(k+1,i-1)M(j+1,l-1)}{Q^B(k,l)}
\end{aligned}$$

we can now use the fact that

$$\sum_{j-L\leq u} \frac{p^{u,L}(k,l)}{(L-(l-k)+1)} = \pi(k,l)$$

to get to

$$\begin{aligned}
\pi^L(i,j) &= \frac{1}{L-(j-i)+1} \sum_{j-L\leq u}^{i} p_o^{u,L}(i,j) + \\
&+ \sum_{u<k<i,j<l\leq u+L} \frac{\pi(k,l)(\mathcal{I}(kl,ij) + M(k+1,i-1) + M(j+1,l-1))}{Q^B(k,l)} \\
&+ \sum_{u<k<i,j<l\leq u+L} \frac{\pi(k,l)M(k+1,i-1)M(j+1,l-1)}{Q^B(k,l)}
\end{aligned}$$

The algorithm is $\mathcal{O}(nw^2)$ in processor time (see Fig. 34) and $\mathcal{O}(n+w^2)$.

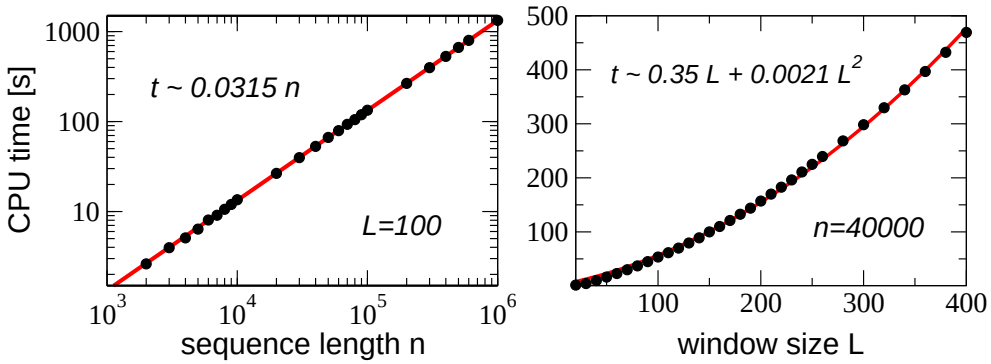


Figure 34: The performance of `RNAplfold` on a Pentium 4 3.2 GHz confirms the theoretical scaling of the CPU time as $\mathcal{O}(nL^2)$

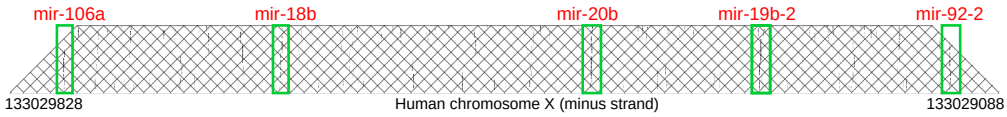


Figure 35: Local pair probabilities (L=100, W=120) in a 740 nt region of the human X chromosome containing five miRNAs annotated in miRBase 7.1 [41]

5.3.1 Visualization of local pair probabilities

The default way to visualize pair probabilities is the dot plot. If we now locally fold a very long molecule, our dot plots will get very big squares. In addition to that, the information content of these dot plots is very small, as only a band of width l along the diagonal can contains base pairs. We therefore decided to show only this small band of the dot plot, and turn it by 45 degrees. What we get is a trapeze with the sequence written on its baseline, angles of 45 degrees. The two lines parallel to the two sides of the trapeze connect any given point with its corresponding bases. Using this approach makes it possible to add local pair probabilities to genome browsers.

5.3.2 Caveats when working with local pair probabilities

Because the $\pi(i, j)$s are averages over different intervals, they can not be easily compared with each other. For example, the probabilities for one base

to be paired do not add up to one. If you e.g. have one base pair $i-w, i$ and one $i, i+w$, the average pair probabilities can both be close to one, so the "probability" of i to be paired can be close to 2. Therefore, we are unable to directly compute properties like the probability of one base not to be paired or the local Shannon Entropy $S = \sum p \ln p$. In case this information is needed, the following approach can be used: We multiply the average pair probabilities by the number of windows the base pair can appear in and divide the result by the number of windows the base can appear in. Doing this, we get as the probability that base i is paired:

$$p(i) = \sum_{j<i} \frac{\pi(i,j)(L-(j-i)+1)}{L} + \sum_{i<j} \frac{\pi(j,i)(L-(i-j)+1)}{L}$$

In this approach, short range base pairs have a higher weight than long range ones. This is equivalent to adding up the probabilities for every window separately and then divide by the number of windows.

5.4 Locally unpaired probability

When looking for possible binding sites for RNA interactions (e.g. miRNA-mRNA binding), it can be crucial to know the probability that a possible binding site – or at least a certain part of it – is unpaired. This question is obviously closely related to the pair probabilities. Therefor, we can also compute the mean probability that a given stretch of bases is unpaired in all windows it occurs in. We closely follow the computation of the "probability of being unpaired" in [92], but again use mean pair probabilities. In the case of no enclosing base pairs or enclosing interior-loops, this is straightforward: For a multi-loop containing an unpaired region, we need to compute $Q^{MM}(i,j)$, which is the partition function of multi-loop contributions with at least 2 stems (elements). Q^{MM} can be computed like multi-loop contributions without closing base-pair:

$$Q^{MM}(i,j) = \sum_k Q^M(i,k-1)Q^{M1}(k,j)$$

We can then use the Q^M and the Q^{MM} arrays to compute the probability that a certain stretch between i and j is in the unpaired region of a multi-loop:

$$
\begin{aligned}
p = \sum_{p<i,q>j} & \pi(p,q) b \frac{Q^{MM}(p+1,i-1)a(q-i+1)}{Q^b p,q} + \\
+ & \pi(p,q) b \frac{Q^{MM}(j+1,q-1)a(j-p+1)}{Q^b p,q} + \\
+ & \pi(p,q) b \frac{Q^{M}(p+1,i-1)Q^{M}(j+1,q-1)a(j-i+1)}{Q^b p,q}
\end{aligned}
$$

The three terms represent the distinct cases where the unpaired region is right of, left of or between two multi-loop components, resp.

5.5 Parameter issues

We want to avoid averaging over a small number of values, so we also have to choose a maximum base pair span well lower than the window size. In the `RNAplfold` program, there are three important parameters that can be changed by the user. These are the window size W, the maximum base pair span L and, if it is computed, the length of the unpaired region u.
As can be expected, variation of these parameters leads to changes in the results while retaining their general shape. In Fig 36 and 37, we visualize the effect of varying different values while the others stay constant. The parameters have to be carefully chosen for different problems (see Fig 38), and may have to be adapted to e.g. GC content or kingdom, as recent unpublished results show.

5.6 Applications

RNAplfold can be used to predict local base pairing probabilities for large sequences - whole chromosomes and even genomes. Future versions of the UCSC genome browser will contain `RNAplfold` information (M. Höchsmann, personal communication).

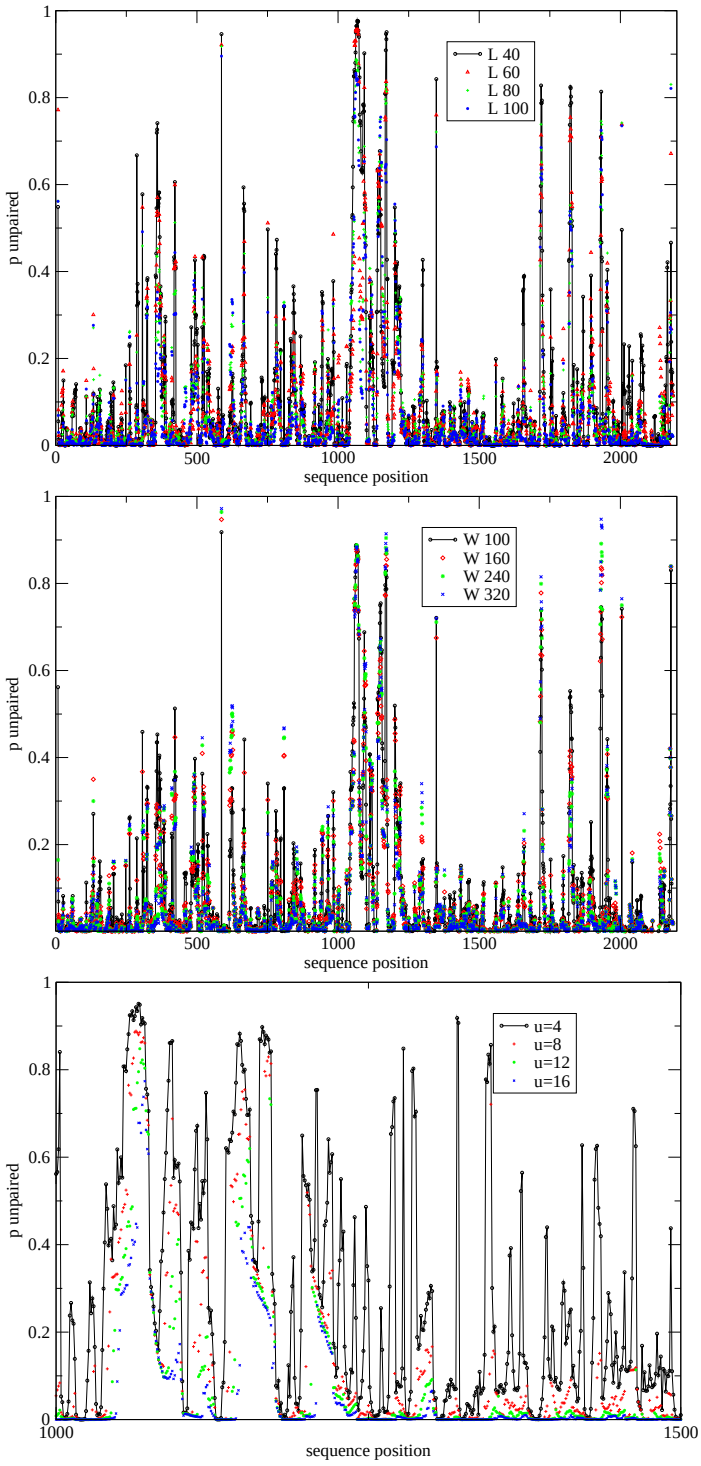


Figure 36: Effect of variation of `RNAplfold` parameters on probability to be unpaired.
top: Variation of L, $W = 100$, $u = 8$.
middle: Variation of W, $L = 80$, $u = 8$.
bottom: variation of u, $W = 100$, $L = 80$, detail.
As can be seen, the general shape, i.e. the location of the peaks, of the two upper curves stays the same, but the amplitudes can vary a lot – with unpredictable directions, as there is no correlation like increasing W will lower unpaired probabilities. Obviously, for varying u (bottom), there is a direct size/probability correlation.
In this example, a part of homo sapiens transmembrane protein 63B mRNA (NM_018426) was used.

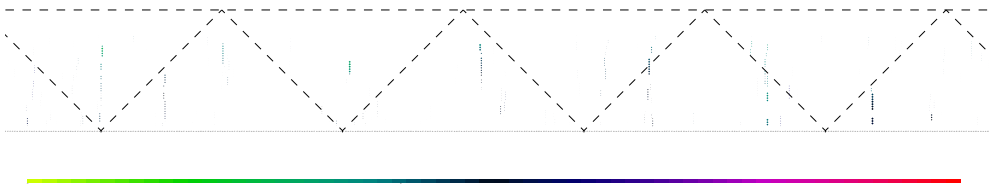


Figure 37: effect of varying W on pair probabilities. Probability differences between $W = 100$ and $W = 240$ in a detail of a difference dot plot, with green higher probability in $w = 100$ and red higher probability in $W = 240$.

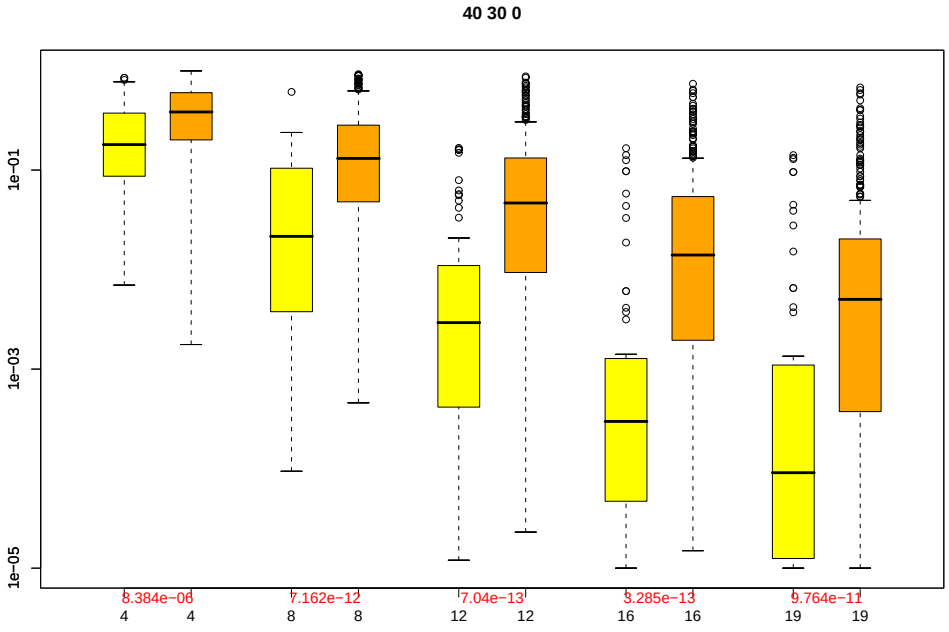


Figure 38: Change of discriminative power of `RNAplfold` for the prediction of the quality of siRNAs. The u parameter was varied (for $W = 40$, $L = 30$) between 4 and 19. A dataset of working and nonworking siRNAs was compared concerning target site accessibility, and the separation of yellow (non working) and orange (working) siRNAs was depicted. Figure taken from [121]

5.7 Further improvements

Almost any RNA molecule will have some kind of secondary structure. The pair probabilities alone are not enough to identify non-coding RNAs. The free energy alone is also not a good indicator of whether a structure belongs to a non-coding RNA. However, combining this information would lead to more useful data, especially if we could incorporate the energy z-score.
Plotting the Gibbs free energy of every window is simple – it is computed in course of the computations already. What we rather would like to have is information on how the energy compares to the energy we would expect from a random sequence. As the energy model considers loops and dangling bases only, we would need sequences with the same dinucleotide content as random sample. For this purpose, machine learning approaches seem to be well suited. RNAz has used a support vector machine to compute the energy z-score of RNA structures, but does not take into account their dinucleotide content. The energy z-score is a measure of how good an energy is.
The energy of random sequences is extreme value distributed, and the z-sore is the difference between the mean of this distribution and the energy score of the molecule of interest, described in units of the standard deviation. The problem lies in the definition of the random sequences. Obviously, the sequences should have the same base content, but because of the nature of the energy function, it is the dinucleotide content that is mostly responsible for the energy of the secondary structures of random sequences. It is possible to compute energy z-scores that include dinucleotide content, but the training of the SVM is not as trivial as for the mononucleotide case used in RNAz.

5.7.1 Local cofolding

The sliding window approach used in `RNAplfold` can, alas, not be converted to the prediction of dimeric structures. The reason for this lies in the fact that in the `RNAcofold` approach, the two molecules are concatenated. This means that instead of replacing or adding one base at the end of the molecule, we would have to replace or add a base between the molecules, which means

in the middle of the molecule because of the concatenation.
However, if we add or replace a base at the center of a given loop, the energy of the loop will change, as it is dependent on its size. This effect can only be taken into account if all respective loops are evaluated again, as the contribution of an unpaired base is different for different loop types and sizes. As the base added (or replaced) is the one closest to the cut c, it can only be part of an exterior-loop if it is not paired, which means that it has no influence on the energy at all. If, however, the base happens to be paired, every single loop containing such a base pair has to be re-evaluated. This is in essence as expansive as simply computing the whole structure from scratch, leading to a time consumption of at least $\mathcal{O}(nl^3)$
An approach that seems to be better is a combination of a fast duplex structure computation program like RNAhybrid [65] or the even faster `RNAplex`, which has been programmed by Tafer in Vienna [121] and the unpaired version of `RNAplfold`. This would in principal correspond to a local version of `RNAup`. The combination of these two approaches has lead to promising results in miRNA target prediction and siRNA target evaluation (Hakim Tafer, personal communication).

6 Using covariance information for RNA secondary structure prediction

6.1 Motivation

RNA secondary structure prediction using dynamic programming and the Turner energy model is not accurate to the satisfaction of Bioinformaticians. Therefore, ways to improve the predictive power of secondary structure prediction have been sought. One approach to do that is to take evolution into account. Assume we know that a certain function of an RNA is conserved in evolution. If we can relate that function to the structure (and not to sequence motifs, which is the other main possibility), we can do two things. Firstly, we can try to find the best structure different molecules have in common, i.e. a structure that has low energy and can be adopted at least in the majority of molecules. Secondly, we can try to find so called **compensatory** or **consistent mutations**. These are mutations that, while changing the primary sequence of the RNA molecule, leave the structure intact. A consistent mutation would change a GU base pair to a GC or an AU base pair, while in a compensatory mutation, both bases are changed to keep the pairing facility. Within the `ViennaRNA` package, this consensus structure prediction is done by the program `RNAalifold` [51]. A problem not yet tackled in `RNAalifold` is the question of how to quantify the contributions of compensatory and consistent mutations. At the moment, the values chosen are arbitrary.

6.2 Implementation

There are different approaches to get to a consensus structure of RNA molecules. The first way, which is done by `RNAalifold`, is first to get an alignment of the molecules and then predicting a secondary structure for that alignment. The second one is to start with a secondary structure and align the sequences to it. The last one is to do the folding and alignment in the same step. An

algorithm to do that has been proposed by Sankoff [115]. Implementations of his algorithm include Dynalign [82], Foldalign 2 [46], PMcomp [50], Stemloc [55] or LocARNA [133].

In principle, the approach of `RNAalifold` is a generalization of the normal RNA secondary structure prediction algorithm to alignments. But before the algorithm can be used, a multiple alignment of the sequences in question has to be generated. If the pairwise sequence identity is high enough (over about 0.75 [134]) simple sequence based tools like Clustalw [124] will suffice. In other cases, the more elaborate Sankoff based algorithms can be used.
The scoring function optimized by `RNAalifold` consists of two parts, a thermodynamic score and a covariance score. The role of the covariance score is to favor structures that contain compensatory or consistent mutations as well as penalize structures that can not be realized by all sequences. To get the covariance score, every possible base pair between every column i and j of the multiple alignment is looked at. The types of base pairs between the positions (here, position means the base in column i of sequence number x) are evaluated. The number of **different** base pairs as well as the number of counter examples (i.e. sequences where the bases i and j are not allowed to form a base pair) is counted. Out of these numbers the covariance score is computed, which is lower (better) if there are many different base pairs and no counter examples, and higher if there are only a few possible base pairs and many counter examples. Usually, a base pair with over 50% counter examples is not considered as an allowed base pair in the folding algorithm. Note that the score used is not a mutual information score. This first step replaces the part of the folding algorithm where the type of base pair is assigned for every pair of bases.
With this information, we turn to the computation of the energies of the different loops. We compute here the average energy of a given loop for all sequences of the alignment. Therefore, every energy evaluation is made not once, but m times, where m is the number of sequences in the multiple

alignment. In the Nussinov case, this leads to:

$$N_{i,j} = \min \left\{ \begin{array}{l} N_{i+1,j}, \\ \min_{i<k\leq j} N_{i+1,k-1} + F_{k+1,j} + \sigma(i,k) \end{array} \right\} \rightarrow$$

$$\rightarrow \quad F_{i,j} = \min \left\{ \begin{array}{l} N_{i+1,j}, \\ \min_{i<k\leq j} N_{(k+1,j)} + N_{i+1,k-1} + c_{cp}(i,j) \sum_{n=1}^{m} \sigma(i,k)_n \end{array} \right\}$$

with $c_{cp}(i,j)$ the co-variation score of columns i and j. As can be seen, this is at most m times slower than the single sequence folding algorithm.
For the Turner model, computing a consensus secondary structure leads to the following recursions, where $F_{i,j}, B_{i,j}, M_{i,j}$ and $M^1_{i,j}$ have the same meaning as in equ. 2:

$$\begin{aligned}
F_{i,j} &= \min \left\{ F_{i+1,j}, \min_{i<k\leq j} B_{i,k} + F_{k+1,j} \right\} \\
B_{i,j} &= \min \left\{ \begin{array}{l} \sum_{n=1}^{m} \mathcal{H}(i,j)_n + c_{cp}(i,j), \\ \min_{i<p<q<j} B_{p,q} \sum_n \mathcal{I}(ij,pq)_n + c_{cp}(i,j), \\ \min_{i<k<j} \alpha + \beta + M_{i,k} + M^1_{k+1,j-1} + c_{cp}(i,j) \end{array} \right. \\
M_{i,j} &= \min \left\{ M_{i+1,j} + m\gamma, \min_k B_{i,k} + M_{k+1,j} + \sum_n \beta(i,k)_n, M^1_{i,j} \right\} \\
M^1_{i,j} &= \min \left\{ M^1_{i,j-1} + m\gamma, B(i,j) + \sum_n \beta(i,j)_n \right\}
\end{aligned}$$

The differences to equ. 2 are minor. Because $\beta(i.j)$ is dependent on the type of base pair, a sum over all sequences has to be made there also. Again, this is at most m times slower than the single sequence case. For the partition function case, the energy contributions have to be added up, in a general example where $\mathcal{E}$ denotes the energy function (i.e. either $\mathcal{H}(i,j)$ or $\mathcal{I}(ij,kl)$)

$$\begin{aligned}
E &= e^{\sum_m \mathcal{E}_m} \\
&= \Pi_m e^{\mathcal{E}_m}
\end{aligned}$$

To speed up the computation, the exponential functions are computed only once for every possible case and kept in memory.

6.3 Extensions of `RNAalifold`

There are two main problems we addressed concerning the computation of consensus secondary structures introduced above. The first is sequence weighting, the second is energy evaluation.

6.3.1 Sequence weighting

For the case of sequence weighting, there are several pathological cases where the unweighted algorithm will lead to results which are not meaningful or wrong. These are cases where

i One sequence shows up more than once in an alignment

ii There are several closely related sequences and only one (or some few) outliers.

iii The alignment used is erroneous.

In the first two cases, the `RNAalifold` output will be biased to one (class) of sequences, when what you want to see is a structure which can be formed by all sequences. These problems also turn up in multiple alignment programs, and sequence weighting is a common way to solve this problem. In the third case the outcome is unpredictable, in the best case simply neglecting a wrongly aligned sequence altogether.
As has been indicated, one way to address this is to weight the sequences and their contribution to the energy of the consensus structure. We do that by considering a sequence tree, which can in principle be made by any convenient way of sequence tree design.
Most of these trees, however, will have weighted edges, so we need a way to get weights for the sequences out of these trees. Here, we follow the approach used by Clustalw [124]. The first step is to (re)root the tree (if it is unrooted, create a root, if not then re-root it). Because the rooting described in [124] is not unambiguous in all cases, we decided to use the midpoint method. The

root vertex is set to be in the center of the longest branch of the tree, i. e. the longest distance between two leafs.
The weight W_n of the sequences n, using l_i as the weight of an edge i and o_i its order, i.e. the number of leafs which share edge i on the path to the root is computed by summing over all edges in the path P from the leaf to the root:

$$W_s = \sum_{i \in P} \frac{l_i}{o_i}$$

Basically, these two steps divide the tree into two sub-trees of equal weight. That means if you have an outlier or a smaller and a bigger group, the structure computed will over proportionally reflect the smaller group. The caveat is of course that this will lead to meaningless results if the two groups of molecules do not share a common fold.
How does weighting help in the case of a misalignment? Obviously, instead of the 50% of sequences cutoff for allowing base pairs a cutoff based on the weights (50% weights) has to be used. If one sequence is misaligned, this usually means that the alignment does not share a common fold. While this would be neglected in the normal `RNAalifold` algorithm, it will lead to a majority of base pairs being disallowed in the case of a weight based cutoff, because wrongly aligned sequences tend to be outliers. So the generated consensus structure will be of bad quality, which should lead to careful examination of the alignment used.

Including sequence weights into `RNAalifold` is algorithmically straightforward. Whenever energy is evaluated, you multiply the energy contribution

of sequence m by its weight w_m:

$$
\begin{aligned}
F_{i,j} &= \min\left\{ F_{i+1,j}, \min_{i<k\leq j} B_{i,k} + F_{k+1,j} \right\} \\
B_{i,j} &= \min\left\{ \begin{array}{l} \sum_{n=1}^{m} \mathcal{H}(i,j)_n w_n + c_{cp}(i,j) \\ \min_{i<p<q<j} B_{p,q} \sum_n \mathcal{I}(ij,pq)_m w_m + c_{cp}(i,j) \\ \min_{i<k<j} \alpha + \beta + M_{i,k} + M^1_{k+1,j-1} + c_{cp}(i,j) \end{array} \right. \\
M_{i,j} &= \min\left\{ M_{i+1,j} + \sum_n w_n\gamma, \min_k B_{i,k} + M_{k+1,j} + \sum_n w_n\beta(i,k)_n, M^1_{i,j} \right\} \\
M^1_{i,j} &= \min\left\{ M^1_{i,j-1} + \sum_n w_n\gamma, B(i,j) + \sum_n w_n\beta(i,j)_n \right\}
\end{aligned}
$$

The transition to the partition function case works similarly to that of single molecules, but for a little inconvenience when computing the energy contributions. As we want to avoid extensive use of the exponent function for performance reasons, using

$$\Pi_m \exp \mathcal{E}^{w_m}$$

is not convenient. Instead, we first compute the sum over all energies

$$\sum_m \mathcal{E} w_m$$

and then have to compute only one exponential function per energy evaluation. We do not pre-compute all possible energy contributions, as is done in the unweighted case, because there are too many different ones.

6.3.2 Energy evaluation

The previous version of `RNAalifold` did only score alignment columns. When doing that, you will get spurious results for every energy evaluation where a gap is scored as a sequence character. This problem is illustrated in Fig 39. It can lead to structures that e.g. are considered to be bulge loops even though in the respective sequence (but not in the alignment!) the bases directly stack onto each other. As most alignments will include gaps, we decided to use the

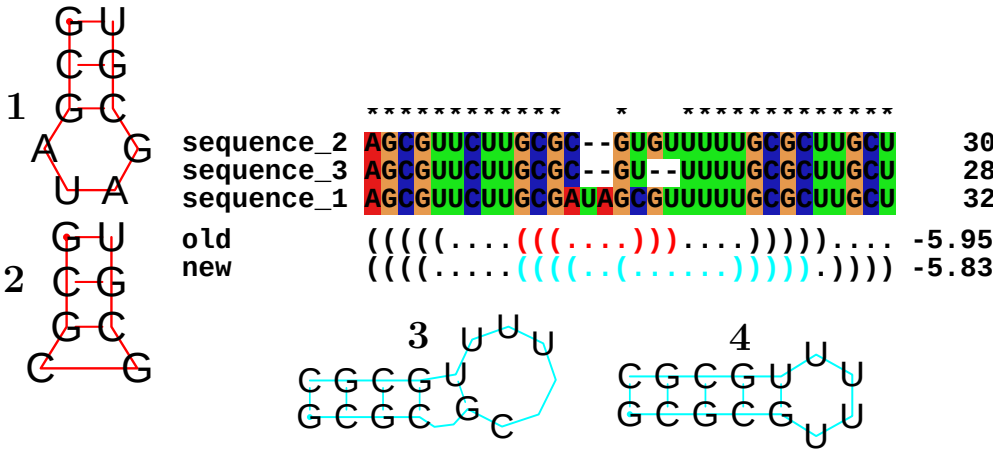


Figure 39: Difference in energy evaluation between the new and the old `RNAalifold` versions. The artificial alignment in the middle is scored. The old `RNAalifold` version gives the first, the new the second structure. The hairpin structure in red, which is predicted by the old version, cannot be adopted by two of the three sequences. While the old `RNAalifold` treats all three sequences as shown in **1**, the real situation for 2 sequences can be seen in **2** – the hairpin is too short, the structure not possible. The hairpin predicted by the new `RNAalifold` version (cyan) is treated like the bulged structure in **3** by the old version, while it looks more like **4** in two sequences, where the stem is 5 bp long.

real sequences, instead of the sequences including gaps, for energy evaluation. In terms of the algorithm this is again quite straightforward. We only have to keep track of which sequence position i^*_m corresponds to alignment position i and sequence m. This is done by including auxiliary arrays $S_m(i) = i^*_m$ that hold this information. To get things like dangles right, we also included two arrays holding the next base 5' ($S5$) or the next base 3' ($S3$) of an alignment position i in sequence m, as well as arrays holding the gap-free sequences.
For the minimum free energy case, the recursions now look like this:

$$
\begin{aligned}
F_{i,j} &= \min\left\{F_{i+1,j}, \min_{i<k\leq j} B_{i,k} + F_{k+1,j}\right\} \\
B_{i,j} &= \min\left\{\begin{array}{l} \sum_m \mathcal{H}(S_m(i), S_m(j)) + c_{cp}(i,j), \\ \min_{i<p<q<j} B_{p,q} \sum_m \mathcal{I}(S_m(i)S_m(j), S_m(p)S_m(q)) + c_{cp}(i,j), \\ \min_{i<k<j} \alpha + \beta + M_{i,k} + M^1_{k+1,j-1} + c_{cp}(i,j) \end{array}\right. \\
M_{i,j} &= \min\left\{M_{i+1,j} + m\gamma, \min_k B_{i,k} + M_{k+1,j} + \sum_m \beta(S_m(i), S_m(k)), M^1_{i,j}\right\} \\
M^1_{i,j} &= \min\left\{M^1_{i,j-1} + m\gamma, B(i,j) + \sum_m \beta(S_m(i), S_m(j))\right\}
\end{aligned}
$$

Of course, this will mean trying to evaluate "impossible" structures. This happens when there are so many gaps in a hairpin-loop that the distance in sequence terms between columns i and j gets smaller than 4. Because of the necessary "backwards compatibility", we have to treat this cases outside the energy evaluation function (as it is used by all programs of the `ViennaRNA` package). The energy evaluation function returns "forbidden" in this case, which basically means a very high number. This will make it impossible for such base pairs to form even if there was only one counter example, in the worst case having only one base-pair less in a stack. Thus, we assign a penalty instead. This penalty, equal to the value assigned to a non WC or wobble base pair, while severe, guarantees that such pathological cases will not render the computation useless.
Intuitively, applying these changes makes `RNAalifold` a program which predicts better structures. However, the results have been somewhat disap-

pointing, especially in terms of recognizing ncRNAs from random controls. While the SCI, used by RNAz [129] as one descriptor, rises, it seems to do so for the "natural" as well as the shuffled "random" sequences. This is not surprising by itself: the new energy evaluation functions will usually not produce higher energies, because the only contribution that leads to higher energies is the "hairpin too short" penalty mentioned above. All other differences, i.e. shorter interior-loops and dangling of real bases will stabilize the secondary structure. Because the SCI uses the energy calculated by `RNAalifold` (SCI= $\frac{\delta G_{alifold}}{G_{fold}}$), it will almost always be higher for the new energy evaluation. However, it is unfortunate that this effect is equally big in random and real sequences, as it can thus not be used to help RNAz in classification.

6.3.3 Results of weighting and evaluating

Changing the code to include weighting and the new energy evaluation slows down `RNAalifold`. However, we did not want to duplicate the alifold functions, but used and changed the existing functions. Therefor, even when you do not want to use the new variant, `RNAalifold` still performs worse than before, considering processor time. Table 2 shows how much slower the new `RNAalifold` is. As one can see, the factor it is slower varies between 4.5 and 1.5 where the higher factors affect very small alignments, where even this quite high factor does not really matter.

While there are many more elaborate ways of designing phylogenetic trees out of sequence data, we used the sequence tree as constructed by Clustalw [124], which is a neighbor joining tree. Note that this does not mean the tree Clustalw uses as guide tree for aligning the sequences, but the tree generated from the final alignment. When looking at the quality of the predicted secondary structures, it can be seen that in "pathological" cases, the enhancement of `RNAalifold` works well:

We folded two molecules which had very gapped alignments, and compared the results of the two `RNAalifold` versions. In both cases, the new version

length	2 factor mfe	2 factor pf	5 mfe	5 pf	9 mfe	9 pf
73	5.9	2.95	3.125	2.56	4.15	2.37
385	1.73	2.48	1.63	2.48	1.78	1.75
1554	1.48	1.80	1.55	1.67	1.70	1.67
2952	1.41	1.52	1.55	1.49	2.07	1.72

Table 2: Performance loss with the new `RNAalifold` version. The factor that `RNAalifold` performs worse than before is shown for 4 different sequence length for alignments of 2, 5 and 9 molecules, resp. The performance loss is quite severe, but it seems manageable.

	New	Old
tRNA	95.43	94.92
g2 intron	95.35	91.82
rRNA	78.63	78.98
U5	87.09	86.15

Table 3: Performance of the weighted new `RNAalifold`. Mean MCCs approximate correlations of the 4 different classes of molecules of BRALIbase II ([37]). As can be seen, the new version outperforms the old one in all but one molecule class. However, the gains are rather small. Only in the g2 intron, the rise in performance seems significant.

outperformed the old one (see Fig 40). As for the ability to distinguish between random alignments and ncRNAs, in a case where one outlier was introduced into an alignment of 4 sequences, the new `RNAalifold` also did better than the old one (Fig 41). However, as can be seen using the BRALIbase I [36] and BRALIbase II [37] alignments(Tables 43) the new `RNAalifold` sometimes performs worse than the old one. A reason for this may be that the outliers, which are weighted more strongly, are the molecules where the alignment is more likely to contain errors. Therefore, `RNAalifold` gets more sensitive to alignment errors. However, while the losses in MCC are only small, there are bigger gains in other molecules.

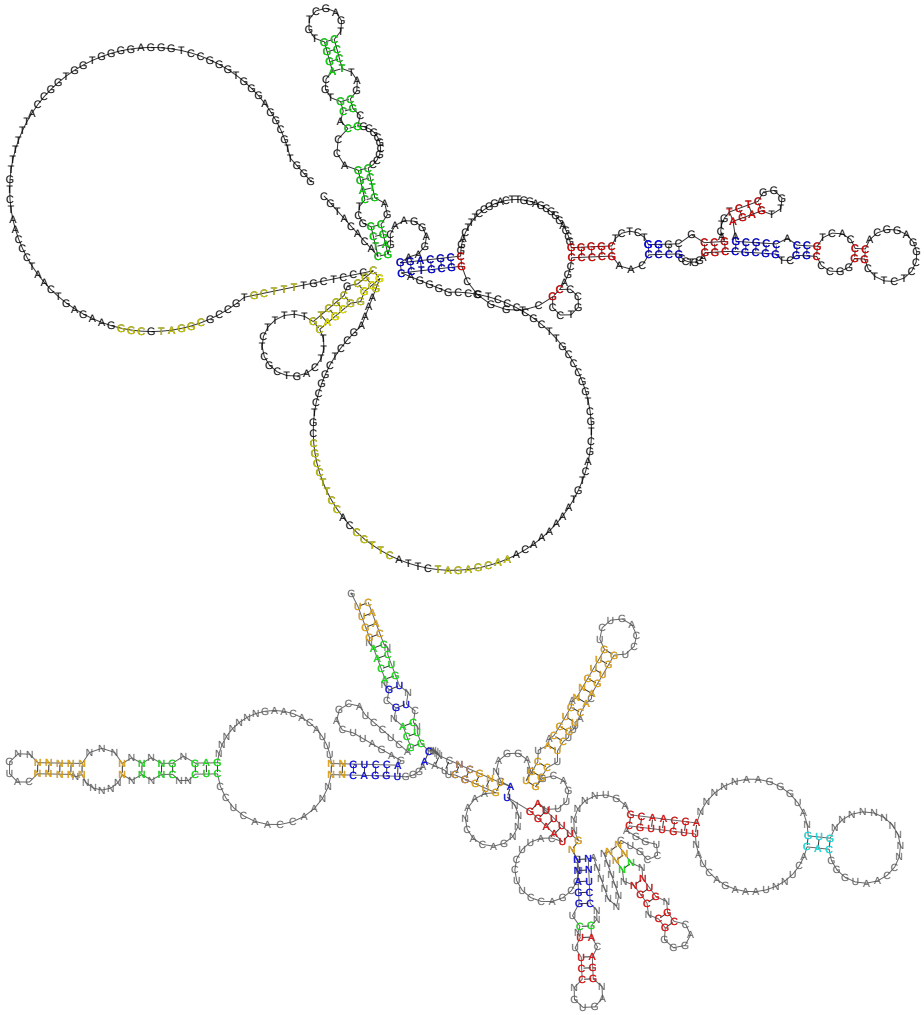


Figure 40: Predictive power of the new `RNAalifold`. Two alignments have been folded using both alifold versions, and the results have been compared to the consensus structure of these molecules. Color code: blue: both wrong, red: both right, green: new right, cyan: old right, yellow/orange: both wrong, but prediction impossible.
The top molecule is telomerase, the structure contains a pseudo knot, so the part of the molecule where the yellow bases are could not be predicted correctly even in theory using `RNAalifold`. As can be seen, the new version of our program can predict an additional stem and gets at least the overall structure right where it possibly can.
The bottom molecule is folded from the seed alignment of nuclear RNAse P from Rfam [42, 43]. Again, additional stems are predicted by the new version of `RNAalifold`, but a small stem is lost, also. The orange bases correspond to base pairs that are part of the consensus structure, but can not be adopted by more than 70% of the molecules in the seed, as they either have gaps or non-WC base-pairs there. Note that many unpaired bases, mostly containing gaps, have been cut out from the interior-loops to get the figure to be concise.

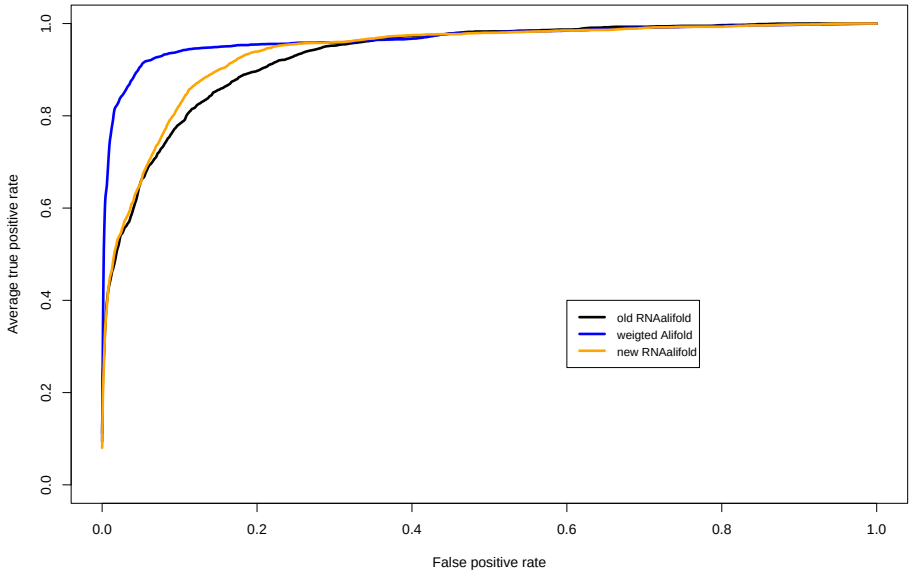


Figure 41: Predictive power of the new `RNAalifold`. In this artificial case, a sequence was added to 4 sequences with mean pairwise identity of 0.95 to get a mean pairwise identity of 0.6. The possibility of the SCI to distinguish between shuffled alignments and ncRNA alignments was measured. The new `RNAalifold` (weighted:blue, unweighted: ocher) fares better than the old one (black) in this example. However, in more "normal" cases the differences are much smaller.

Sequence	sens	select	MCC
LSU M Old	57.0	63.6	60.3
New	54.6	62.9	58.8
Weight	57.4	62.0	59.7
LSU.H	51.1	54.6	52.9
New	51.6	54.9	53.2
Weight	56.3	61.2	58.7
SSU.M	81.8	85.7	83.8
New	82.7	86.2	84.4
Weight	81.8	85.7	83.8
SSU.H	67.1	68.4	67.8
New	69.0	69.9	69.5
Weight	73.1	74.5	73.8
RNaseP.H.O.	72.7	72.1	72.4
New	70.9	70.3	70.6
Weight	72.7	72.1	72.4
RNaseP.M.O	80.0	88.0	84.0
New	73.6	75.0	74.3
Weight	80.0	87.1	83.6

Table 4: Performance of the new `RNAalifold` on the BRALIbase I molecules. Red: the old version outperforms the new one, green: the new version is better. H and M denote high and low sequence conservation, resp. The computation of sensitivity, selectivity and MCC were performed as described in [36]

6.4 Predicting locally conserved structures

The folding of blocks of large alignment is e. g. used in RNAz, a program to predict conserved non coding RNAs. A sliding windows approach as mentioned above is used with an overlap usually about a third of the window size. RNAz uses the structural conservation index (SCI) for the training of the underlying support vector machine. This value is computed out of the consensus minimum free energy and the single minimum free energies:

$$\mathrm{SCI} = \frac{E_A}{\bar{E}}$$

with E_A the `RNAalifold` minimum free energy and $\bar{E}$ the mean over all free energies of the single sequences [129]. Because it is possible that conserved structures are lying outside all used windows, an approach similar to `RNAplfold` or `RNALfold` may increase the performance of the program. However, it is not straightforward to incorporate these into RNAz.

The same approach used in `RNAalifold` [51] can also be applied to local prediction. Only the energy evaluations have to be changed. We combined the local variants of `RNAalifold` to `RNALalifold`, and used the option "-p" for the `RNAplfold` analogon.
We applied this to an alignment of the human miR cluster shown in Fig. 35 with 3 other mammalian species. As the results in Fig. 42 show, this approach can significantly reduce noise, as the pre-miRNAs are the only secondary structures which show high conservation.

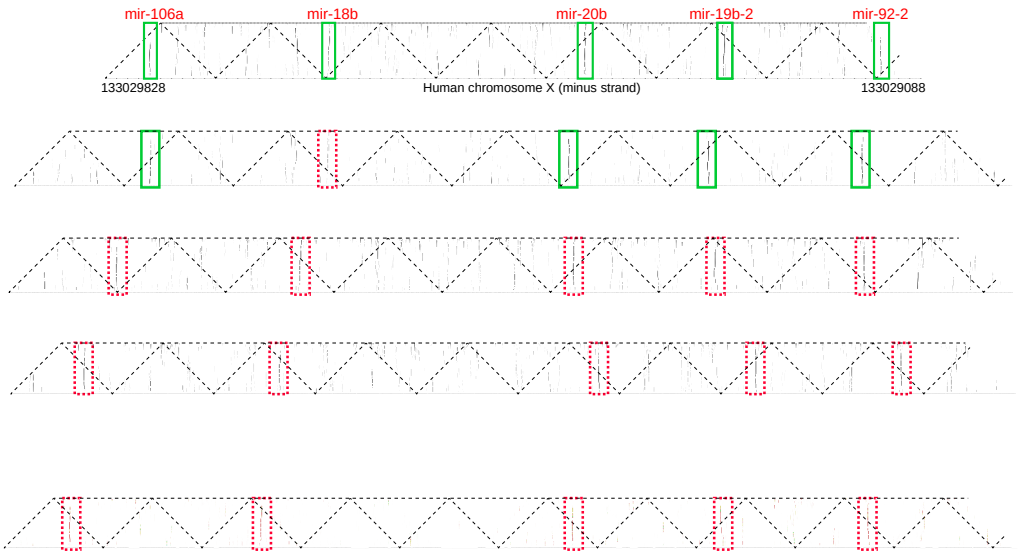


Figure 42: Local pair probabilities of a homologue miR cluster of 4 different mammalian species and of their alignment. From top to bottom: human, mouse, dog, opossum, alignment. The green boxes correspond to miRs annotated in miRBase [41], the red dashed boxes are miRNAs not annotated in miRBase. The alignment has significantly less noise, while the 5 miRNAs are the only long stems that can be seen [19]

7 Conclusion and outlook

We presented here valuable additions for different fields of RNA bioinformatics. All programs and modifications of programs are or will soon be part of the `ViennaRNA` software package. The essential functions are also part of the `ViennaRNA` C library and can be used by bioinformaticians to develop their own programs.

The modifications of the `RNAfold` program lead to a minor speed-up of the fastest thermodynamic RNA folding program freely available. Furthermore, they open up the possibility for partition function calculations of canonical structures. This, as we have shown, can be an asset for the prediction of the secondary structures of certain families of RNA molecules, e.g. tRNAs.

Our `RNAcofold` algorithm can give a good approximation to the structure of RNA dimers. It can also be used for predicting equilibrium concentrations of monomers and dimers in solution. Its applications include the study of miRNA or siRNA binding, where the structural effects of mutations can be predicted *in silico*. This can provide explanations for the phenotypic effect of these mutations. As there are many different non coding RNAs that essentially work by base pairing and dimerization, `RNAcofold` could as well be a valuable tool for the analysis of their functional mechanisms.
There are two main ways we are planning to expand the `RNAcofold` program. Firstly, we will merge this approach with `RNAup` to be able to incorporate special cases of intermolecular pseudo knots into the algorithm. We have described how this can be done. Secondly, we will use the slightly different approach by Dirks et al. [26] to incorporate higher order multimers into our computations.

With `RNAplfold`, we provide a tool that can be used for partition function computations at a genomic scale. By including the unpaired option, we also provide a fast tool for accessibility prediction of putative target sites. Target

site accessibility has only recently been identified as an important determinant for the efficacy of miRNA and siRNA function [2, 74]. One important asset of our program is that the results of the computations can easily be stored for later use. This means that once you have computed the accessibility of say all human 3'UTRs, you can then simply look up the accessibility at any target site of interest. Also, if e.g. a new RNA is found, you can immediately get a first idea about its local secondary structure.
Future work we are planning concerning `RNAplfold` is to identify unusually stable structures by use of an energy z-score. Structural stability has been identified as an useful criterion for the identification of functional RNAs. Furthermore, we plan to develop specialized tools e.g. for miRNA target prediction that combine accessibility calculations using RNAplfold with domain specific knowledge. The local pair probabilities created by `RNAplfold` can also be used as a basis for structural alignment tools like LocARNA as well as homology search tools.

We introduced sequence weighting as well as a more elaborate energy evaluation in the `RNAalifold` program. While this helps to get better results in very special cases of consensus structure prediction, the performance gain was not as big as we hoped for. However, we think that using alignments of better quality, e.g. by identifying misaligned sequences and realigning them, will lead to better results. A more elaborate sequence weighting will also increase the performance. As the weighting itself is a pre-processing step, more sophisticated weighting schemes can be introduced with ease.
We also introduced `RNALalifold`, combining local folding with consensus structure prediction. Local pair probabilities of alignments are considerably less noisy than single predictions are. This can be used to identify important structures in conserved non coding RNAs. Furthermore, `RNAz` is using a sliding window approach on alignments to find conserved non coding RNAs. While we did not yet succeed in replacing this with `RNALalifold`, the usage of our maximum overlap approach should enhance the sensitivity of `RNAz`.

To further improve `RNAalifold`, an important step will be to put the rather ad hoc balance between the energy score and the consensus score on a probabilistic basis. This could entail pre-processing steps to compute optimal penalties and bonuses that are now just constants.

References

[1] C. Alkan, E. Karakoc, J. H. Nadeau, S. C. Sahinalp, and K. Zhang. RNA-RNA interaction prediction and antisense RNA target search. *J Comput Biol*, 13(2):267–82, 2006.

[2] Stefan .L Ameres, Javier Martinez, and Renee Schroeder. Molecular basis for target rna recognition and cleavage by human risc. *cell*, 2007. in press.

[3] M Andronescu, Z C Zhang, and A Condon. Secondary structure prediction of interacting RNA molecules. *J. Mol. Biol.*, 345:987–1001, 2005.

[4] L. Argaman and S. Altuvia. fhla repression by oxys rna: kissing complex formation at two sites results in a stable antisense-target rna complex. *J Mol Biol*, 300(5):1101–12, Jul 28 2000.

[5] Nenad Ban, Poul Nissen, Jeffrey Hansen, Peter B. Moore, and Thomas A. Steitz. The complete atomic structure of the large ribosomal subunit at 2.4 resolution. *Science*, 289, 2002.

[6] B. Basham, G. P. Schroth, and P. S. Ho. An a-DNA triplet code: thermodynamic rules for predicting a- and b-DNA. *Proc Natl Acad Sci*, 92:6464–6468, 1995.

[7] S. H. Bernhart, I. L. Hofacker, and P. F. Stadler. Local RNA base pairing probabilities in large sequences. *Bioinformatics*, 22(5):614–5, Mar 1 2006.

[8] S H Bernhart, H Tafer, U Mückstein, C Flamm, P F Stadle r, and I L Hofacker. Partition function and base pairing probabilities of RNA heterodimers. *Algorithms for Molecular Biology*, 1(3), March 2006.

[9] Athanasius F. Bompfünewerer, Rolf Backofen, Stephan H. Berhart, Jana Hertel, , Ivo L. Hofacker, Peter F. Stadler, and Sebastian Will.

Variations on RNA folding and alignment: Lessons from benasque. *J. Math. Biol.*, 2007. in press.

[10] Athanasius F. Bompfünewerer, Christoph Flamm, Claudia Fried, Guido Fritzsch, Ivo L. Hofacker, Jörg Lehmann, Kristin Missal, Axel Mosig, Bettina Müller, Sonja J. Prohaska, Bärbel M. R. Stadler, Peter F. Stadler, Andrea Tanzer, Stefan Washietl, and Christina Witwer. Evolutionary patterns of non-coding RNAs. *Th. Biosci.*, 123:301–369, 2005.

[11] Jean Brachet. La localisation des acides pentosenucliques dans les tissus animaux et les oeligufs d'amphibiens en voie de l'eveloppment. *Arch. Biol. (Lige)*, 53, 1941.

[12] J. Brennecke, A. Stark, R.B. Russell, and S.M. Cohen. Principles of microRNA-target recognition. *PLoS Biol*, 3(3):e85, 2005.

[13] P. Brion and E. Westhof. Hierarchy and dynamics of RNA folding. *Annu Rev Biophys Biomol Struct*, 26:113–37, 1997.

[14] Guerrier-Takada C., Gardiner K., Marsh T., Pace N, and Altman S. The RNA moiety of ribonuclease p is the catalytic subunit of the enzyme. *Cell*, 35:849–57, Dec 1983.

[15] Neil A. Campbell. *Biology*, chapter 1. Benjamin-Cummings Pub Co, 4 edition, 1996.

[16] T. Caspersson. Studien ber den eiweiumsatz der zelle. *Naturwissenschaften*, 29, 1941.

[17] Noam Chomsky. On certain formal properties of grammars. *Inform. Cont.*, 2:137–167, 1959.

[18] Supratim Choudhuri. The path from nuclein to human genome: A brief history of DNA with a note on human gen sequencing and its impact

on future research in biology. *Bulletin of Science Technology Society*, 23:360–367, 2003.

[19] Athanasius F. Bompfünewerer Consortium, R. Backofen, S. H. Bernhart, C. Flamm, C. Fried, G. Fritzsch, J. Hackermuller, J. Hertel, I. L. Hofacker, K. Missal, A. Mosig, S. J. Prohaska, D. Rose, P. F. Stadler, A. Tanzer, S. Washietl, and S. Will. RNAs everywhere: genome-wide annotation of structured RNAs. *J Exp Zoolog B Mol Dev Evol*, 308(1):1–25, Jan 15 2007.

[20] Francis H. C. Crick. On protein synthesis. *Symp. Soc. Exp. Biol*, XII:139–163, 1958.

[21] Francis H. C. Crick. Central dogma of molecular biology. *Nature*, 227:561–563, 1970.

[22] Larry J. Croft, Martin J. Lercher, Michael J. Gagen, and John S. Mattick. Is prokaryotic complexity limited by accelerated growth in regulatory overhead? 2003.

[23] R. A. Dimitrov and Michael Zuker. Prediction of hybridization and melting for double-stranded nucleic acids. *Biophys. J.*, 87:215–226, 2004.

[24] Ye Ding and Charles E. Lawrence. Statistical prediction of single-stranded regions in RNA secondary structure and application to predicting effective antisense target sites and beyond. *Nucl. Acids Res.*, 29:1034–1046, 2001.

[25] Ye Ding and Charles E. Lawrence. A statistical sampling algorithm for RNA secondary structure prediction. *Nucl. Acids Res.*, 31:7180–7301, 2003.

[26] Robert M. Dirks, Justin S. Bois, Joseph M. Schaeffer, Erik Winfree, and Niles A. Pierce. Thermodynamic analysis of interacting nucleic acid strands. *SIAM*, 49(1), 2007.

[27] C.B. Do, D.A. Woods, and S. Batzoglou. Contrafold: RNA secondary structure prediction without physics-based models. *Bioinformatics*, 22(14):90–98, 2006.

[28] J G Doench and P A Sharp. Specificity of microRNA target selection in translational repression. *Genes Devel.*, 18:504–511, 2004.

[29] K. J. Doshi, J. J. Cannone, C. W. Cobaugh, and R. R. Gutell. Evaluation of the suitability of free-energy minimization using nearest-neighbor energy parameters for RNA secondary structure prediction. *BMC Bioinformatics*, 5:105, Aug 5 2004.

[30] R. D. Dowell and S. R. Eddy. Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction. *BMC Bioinformatics*, 5:71, 2004.

[31] Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchinson. *Biological sequence analysis*. Cambridge University Press, 1998.

[32] Sean R. Eddy and Richard Durbin. RNA sequence analysis using covariance models. *Submitted to Nucleic Acids Research*, unknown(unknown):unknown, 1994.

[33] M. G. Ferreira, K. M. Miller, and J. P. Cooper. Indecent exposure: when telomeres become uncapped. *Mol Cell*, 13(1):7–18, Jan 16 2004.

[34] Andrew Fire, SiQun Xu, Mary K. Montgomery, Steven A. Kostas, Samuel A. Driver, and Craig C. Mello. Potent and specific genetic interference by double-stranded RNA in caenorhabditis elegans. *Nature*, 391, 1998.

[35] C. Flamm, W. Fontana, I. L. Hofacker, and P. Schuster. RNA folding at elementary step resolution. *RNA*, 6(3):325–38, March 2000.

[36] P. P. Gardner and R. Giegerich. A comprehensive comparison of comparative RNA structure prediction approaches. *BMC Bioinformatics*, 5:140, Sep 30 2004.

[37] P. P. Gardner, A. Wilm, and S. Washietl. A benchmark of multiple sequence alignment programs upon structural RNAs. *Nucleic Acids Res*, 33(8):2433–9, 2005.

[38] D. Gautheret, S. H. Damberger, and R. R. Gutell. Identification of base-triples in RNA using comparative sequence analysis. *J Mol Biol*, 248(1):27–43, Apr 21 1995.

[39] D. Gautheret, D. Konings, and R. R. Gutell. A major family of motifs involving g.a mismatches in ribosomal RNA. *J Mol Biol*, 242(1):1–8, Sep 9 1994.

[40] S. Gottesman. Micros for microbes: non-coding regulatory rnas in bacteria. *Trends Genet*, 21(7):399–404, July 2005.

[41] S. Griffiths-Jones. mirbase: the microRNA sequence database. *Methods Mol Biol*, 342:129–38, 2006.

[42] S Griffiths-Jones, A Bateman, M Marshall, A Khanna, and S R Eddy. Rfam: an RNA family database. *Nucleic Acids Res.*, 31(1):439–41, 2003.

[43] S. Griffiths-Jones, S. Moxon, M. Marshall, A. Khanna, S. R. Eddy, and A. Bateman. Rfam: annotating non-coding RNAs in complete genomes. *Nucleic Acids Res*, 33(Database issue):D121–4, 2005.

[44] Jörg Hackermüller, Nicole-Claudia Meisner, Manfred Auer, Markus Jaritz, and Peter F. Stadler. The effect of RNA secondary structures on RNA-ligand binding and the modifier RNA mechanism: A quantitative model. *Gene*, 345:3–12, 2005.

[45] P. W. Haebel, S. Gutmann, and N. Ban. Dial tm for rescue: tmRNA engages ribosomes stalled on defective mrnas. *Curr Opin Struct Biol*, 14(1):58–65, February 2004.

[46] J. H. Havgaard, R. B. Lyngso, G. D. Stormo, and J. Gorodkin. Pairwise local structural alignment of RNA sequences with sequence similarity less than 40%. *Bioinformatics*, 21(9):1815–24, May 1 2005.

[47] M. W. Hentze and L. C. Kühn. Molecular control of vertebrate iron metabolism: mRNA-based regulatory circuits operated by iron, nitric oxide, and oxidative stress. *Proc Natl Acad Sci U S A*, 93(16):8175–82, Aug 6 1996.

[48] M. B. Hoagland, M. L. Stephenson, J. F. Scott, L. I. Hecht, and P. C. Zamecnik. A soluble ribonucleic acid intermediate in protein synthesis. *J Biol Chem*, 231(1):241–57, March 1958.

[49] M. B. Hoagland, P. C. Zamecnik, and M. L. Stephenson. Intermediate reactions in protein biosynthesis. *Biochim Biophys Acta*, 24(1):215–6, April 1957.

[50] I. L. Hofacker, S. H. Bernhart, and P. F. Stadler. Alignment of RNA base pairing probability matrices. *Bioinformatics*, 20(14):2222–7, Sep 22 2004.

[51] I. L. Hofacker, M. Fekete, and P. F. Stadler. Secondary structure prediction for aligned rna sequences. *J Mol Biol*, 319(5):1059–66, Jun 21 2002.

[52] I L Hofacker, W Fontana, P F Stadler, S Bonhoeffer, M Tacker, and P Schuster. Fast folding and comparison of RNA secondary structures. *Monatsh. Chemie*, 125(2):167–188, 1994.

[53] I. L. Hofacker, B. Priwitzer, and P. F. Stadler. Prediction of locally stable RNA secondary structures for genome-wide surveys. *Bioinformatics*, 20(2):186–90, Jan 22 2004.

[54] Ivo L. Hofacker, Peter Schuster, and Peter F. Stadler. Combinatorics of RNA secondary structures. *Discr. Appl. Math.*, 89:177–207, 1998.

[55] I. Holmes. Accelerated probabilistic inference of RNA structure evolution. *BMC Bioinformatics*, 6:73, 2005.

[56] J. Isaksson and Chattopadhyaya J. A uniform mechanism correlating dangling-end stabilization and stacking geometry. *Biochemistry*, 44, 2005.

[57] H. Isambert and E. D. Siggia. Modeling RNA folding paths with pseudoknots: application to hepatitis delta virus ribozyme. *Proc Natl Acad Sci U S A*, 97(12):6515–20, Jun 6 2000.

[58] Homer Jacobson and Walter H. Stockmayer. Intramolecular reaction in polycondensations. i. the theory of linear systems. *The Journal of Chemical Physics*, 18(12):1600–1606, 1950.

[59] S. K. Jang, H. G. Krausslich, M. J. Nicklin, G. M. Duke, A. C. Palmenberg, and E. Wimmer. A segment of the 5' nontranslated region of encephalomyocarditis virus RNA directs internal entry of ribosomes during in vitro translation. *J Virol*, 62(8):2636–43, August 1988.

[60] R. J. Keenan, D. M. Freymann, R. M. Stroud, and P. Walter. The signal recognition particle. *Annu Rev Biochem*, 70:755–75, 2001.

[61] B. Knudsen and J. Hein. Pfold: RNA secondary structure prediction using stochastic context-free grammars. *Nucleic Acids Research*, 31(13):3423–3428, 2003.

[62] A. S. Krasilnikov, Y. Xiao, T. Pan, and A. Mondragon. Basis for structural diversity in homologous rnas. *Science*, 306(5693):104–7, Oct 1 2004.

[63] A. S. Krasilnikov, X. Yang, T. Pan, and A. Mondragon. Crystal structure of the specificity domain of ribonuclease p. *Nature*, 421(6924):760–4, Feb 13 2003.

[64] A. Krol. Evolutionarily different RNA motifs and RNA-protein complexes to achieve selenoprotein synthesis. *Biochimie*, 84(8):765–74, August 2002.

[65] J. Kruger and M. Rehmsmeier. Rnahybrid: microRNA target prediction easy, fast and flexible. *Nucleic Acids Res*, 34(Web Server issue):W451–4, Jul 1 2006.

[66] Kelly Kruger, Grabowski Paula J., Arthur J. Zaug, Julie Sands, Daniel E. Gottschling, and Thomas R. Cech. Self-splicing RNA: Autoexcision and autocyclization of the ribosomal RNA intervening sequence of tetrahymena. *Cell*, 31:147–57, 1982.

[67] K. Y. Kwek, S. Murphy, A. Furger, B. Thomas, W. O'Gorman, H. Kimura, N. J. Proudfoot, and A. Akoulitchev. U1 snRNA associates with tfiih and regulates transcriptional initiation. *Nat Struct Biol*, 9(11):800–5, November 2002.

[68] J. C. Labbe, S. Hekimi, and L. A. Rokeach. Assessing the function of the ro ribonucleoprotein complex using caenorhabditis elegans as a biological tool. *Biochem Cell Biol*, 77(4):349–54, 1999.

[69] A. Lambert, J. F. Fontaine, M. Legendre, F. Leclerc, E. Permal, F. Major, H. Putzer, O. Delfour, B. Michot, and D. Gautheret. The ERPIN server: an interface to profile-based RNA motif identification. *Nucleic Acids Res*, 32(Web Server issue):W160–5, 2004.

[70] K. Lari and S.J. Young. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*, 4:35–56, 1990.

[71] Rosalind C. Lee, Rhonda L. Feinbaum, and Victor Ambros. The c. elegans heterochronic gene lin-4 encodes small RNAs with antisense complementarity to lin-14. *Cell*, 75:843–854, 1993.

[72] N. B. Leontis, A. Lescoute, and E. Westhof. The building blocks and motifs of RNA architecture. *Curr Opin Struct Biol*, 16(3):279–87, 2006.

[73] N. B. Leontis, J. Stombaugh, and E. Westhof. The non-watson-crick base pairs and their associated isostericity matrices. *Nucleic Acids Res*, 30(16):3497–531, Aug 15 2002.

[74] D. Long, R. Lee, P. Williams, C. Y. Chan, V. Ambros, and Y. Ding. Potent effect of target structure on microrna function. *Nat Struct Mol Biol*, 14(4):287–94, April 2007.

[75] T. M. Lowe and S. R. Eddy. tRNAscan-se: a program for improved detection of transfer RNA genes in genomic sequence. *Nucleic Acids Res*, 25(5):955–64, Mar 1 1997.

[76] R. B. Lyngso and C. N. Pedersen. RNA pseudoknot prediction in energy-based models. *J Comput Biol*, 7(3-4):409–27, 2000.

[77] Rune B. Lyngsoe, Michael Zuker, and Christian N. S. Pedersen. Fast evaluation of internal loops in RNA secondary. *Bioinformatics*, 15(5):440–445, 1999.

[78] Francis H. Martin, Olke C. Uhlenbeck, and Paul Doty. Self complimentary oligoribonucleotides: Adenylic acid-uridylic acid block copolymers. *JMB*, 57:201–215, 1971.

[79] B. Masquida and E. Westhof. On the wobble gou and related pairs. *RNA*, 6(1):9–15, January 2000.

[80] D. H. Mathews, M. E. Burkard, S. M. Freier, J. R. Wyatt, and D. H. Turner. Predicting oligonucleotide affinity to nucleic acid targets. *RNA*, 5(11):1458–69, November 1999.

[81] D H Mathews, J Sabina, M Zuker, and D H Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J. Mol. Biol.*, 288:911–940, 1999.

[82] D. H. Mathews and D. H. Turner. Dynalign: an algorithm for finding the secondary structure common to two RNA sequences. *J Mol Biol*, 317(2):191–203, Mar 22 2002.

[83] D. H. Mathews and D. H. Turner. Experimentally derived nearest-neighbor parameters for the stability of RNA three- and four-way multi-branch loops. *Biochemistry*, 41(3):869–80, Jan 22 2002.

[84] J S McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29:1105–1119, 1990.

[85] Nicole-Claudia Meisner, Jörg Hackermüller, Volker Uhl, Andras Aszódi, Markus Jaritz, and Manfred Auer. mRNA openers and closers: A methodology to modulate AU-rich element controlled mRNA stability by a molecular switch in mRNA conformation. *Chembiochem.*, 5:1432–1447, 2004.

[86] F. Michel, A. D. Ellington, S. Couture, and J. W. Szostak. Phylogenetic and genetic evidence for base-triples in the catalytic domain of group i introns. *Nature*, 347(6293):578–80, Oct 11 1990.

[87] Johann Friedrich Miescher. Ueber die chemische zusammensetzung der eiterzellen. *Medisch-chemische Untersuchungen*, 4:441–460, 1871.

[88] K. Missal, D. Rose, and P. F. Stadler. Non-coding rnas in ciona intestinalis. *Bioinformatics*, 21 Suppl 2:ii77–ii78, Sep 1 2005.

[89] K. Missal, X. Zhu, D. Rose, W. Deng, G. Skogerbo, R. Chen, and P. F. Stadler. Prediction of structured non-coding rnas in the genomes of the nematodes caenorhabditis elegans and caenorhabditis briggsae. *J Exp Zoolog B Mol Dev Evol*, 306(4):379–92, Jul 15 2006.

[90] J. P. Morrissey and D. Tollervey. Birth of the snornps: the evolution of rnase mrp and the eukaryotic pre-rRNA-processing system. *Trends Biochem Sci*, 20(2):78–82, February 1995.

[91] M. H. Mossink, A. van Zon, R. J. Scheper, P. Sonneveld, and E. A. Wiemer. Vaults: a ribonucleoprotein particle involved in drug resistance? *Oncogene*, 22(47):7458–67, Oct 20 2003.

[92] U. Mückstein, H. Tafer, J. Hackermuller, S. H. Bernhart, P. F. Stadler, and I. L. Hofacker. Thermodynamics of RNA-RNA binding. *Bioinformatics*, 22(10):1177–82, May 15 2006.

[93] Carolyn Napoli, Christine Lemieux, and Richard Jorgensen. Introduction of a chimeric chalcone synthase gene into petunia results in reversible co-suppression of homologous genes in trans. *The Plant Cell*, 2:279–289, 1990.

[94] M. Nirenberg, P. Leder, M. Bernfield, R. Brimacombe, J. Trupin, F. Rottman, and C. O'Neal. RNA codewords and protein synthesis, vii. on the general nature of the RNA code. *Proc Natl Acad Sci U S A*, 53(5):1161–8, May 1965.

[95] Poul Nissen, Jeffrey Hansen, Nenad Ban, Peter B. Moore, and Thomas A. Steitz. The structural basis of ribosome activity in peptide bond synthesis. *Science*, 289, 2002.

[96] R Nussinov and A. B. Jacobson. Fast algorithm for predicting the secondary structure of single-stranded RNA. *PNAS*, 77:6309–6313, 1980.

[97] R Nussinov, G Piecznik, J R Griggs, and D J Kleitman. Algorithms for loop matching. *SIAM J. Appl. Math.*, 35(1):68–82, 1978.

[98] Ruth Nussinov, George Piecznik, Jerrold R. Griggs, and Daniel J. Kleitman. Algorithms for loop matching. *SIAM J. Appl. Math.*, 35(1):68–82, 1978.

[99] A.Y. Ogurtsov, S.A. Shabalina, A.S Kondrashov, and M.A. Roytberg. Analysis of internal loops within the RNA secondary structure in almost quadratic time. *Bioinformatics*, 22(11), 2006.

[100] Tatsuo Ohmichi, Shu ichi Nakano, Daisuke Miyoshi, and Naoki Sugimoto. Long RNA dangling end has large energetic contribution to duplex stability. *J.Am.Chem.Soc.*, 124, 2002.

[101] A. A. Patel and J. A. Steitz. Splicing double: insights from the second spliceosome. *Nat Rev Mol Cell Biol*, 4(12):960–70, December 2003.

[102] J. S. Pedersen, G. Bejerano, A. Siepel, K. Rosenbloom, K. Lindblad-Toh, E. S. Lander, J. Kent, W. Miller, and D. Haussler. Identification and classification of conserved RNA secondary structures in the human genome. *PLoS Comput Biol*, 2(4):e33, April 2006.

[103] D. D. Pervouchine. Iris: intermolecular RNA interaction search. *Genome Inform*, 15(2):92–101, 2004.

[104] J. M. Pipas and J. E. McMahon. Method for predicting RNA secondary structure. *Proc Natl Acad Sci U S A*, 72(6):2017–21, 1975.

[105] The ENCODE project consortium. dentification and analysis of functional elements in 1% of the human genome by the encode pilot project. *Nature*, 447, 2007.

[106] G. J. Quigley and A. Rich. Structural domains of transfer RNA molecules. *Science*, 194(4267):796–806, Nov 19 1976.

[107] J Reeder and R Giegerich. Design, implementation and evaluation of a practical pseudoknot folding algorithm based on thermodynamics. *BMC Bioinformatics*, 5, 2004.

[108] Brenda J. Reinhart, Earl G. Weinstein, Matthew W. Rhoades, Bonnie Bartel, and David P. Bartel. MicroRNAs in plants. *Genes & Development*, 16:1616–1626, 2002.

[109] A. Rich and J. D. Watson. Some relations between DNA and RNA. *Proc Natl Acad Sci U S A*, 40(8):759–64, August 1954.

[110] Elena Rivas and Sean R. Eddy. A dynamic programming algorithm for RNA structure prediction including pseudoknots. *J Mol Biol.*, 285(5):2053–68, 1999.

[111] Elena Rivas and Sean R. Eddy. Noncoding RNA gene detection using comparative sequence analysis. *BMC Bioinformatics*, 2(8), 2001.

[112] Nicoletta Romano and Giuseppe Macino. Quelling: transient inactivation of gene expression in neurospora crassa by transformation with homologous sequences. *Mol. Microbiol.*, 6:3343–53, 1992.

[113] Y. Sakakibara, M. Brown, R. Hughey, I. S. Mian, K. Sjolander, R. C. Underwood, and D. Haussler. Stochastic context-free grammars for tRNA modeling. *Nucleic Acids Res*, 22(23):5112–20, 1994.

[114] Yasubumi Sakakibara, Michael Brown, Richard Hughey, I. Saira Mian, Kimmen Sjölander, Rebecca C. Underwood, and David Haussler. Recent methods for RNA modeling using stocastic context-free grammars. In *CPM*, pages 289–306, 1994.

[115] D. Sankoff. Simultaneous solution of the RNA folding, alignment, and proto-sequence problems. *SIAM J. Appl. Math.*, 45:810–825, 1985.

[116] S. Schubert, A. Grunweller, V.A. Erdmann, and J. Kurreck. Local RNA target structure influences siRNA efficacy: Systematic analysis of intentionally designed binding regions. *J. Mol. Biol.*, 348(4):883–93, 2005.

[117] D. Schumperli and R. S. Pillai. The special sm core structure of the u7 snrnp: far-reaching significance of a small nuclear ribonucleoprotein. *Cell Mol Life Sci*, 61(19-20):2560–70, October 2004.

[118] H. R. Schwarz. *Numerische Mathematik*. B.G. Teubner, Stuttgart, 1986.

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

[120] A. Stark, J. Brennecke, R.B. Russell, and S.M. Cohen. Identification of drosophila microRNA targets. *PLoS Biol.*, 1(3):e60, 2003.

[121] Hakim Tafer. RNAplex – a fast and flexible RNA-RNA interaction search tool. 2007. submitted.

[122] R. J. Taft, M. Pheasant, and J. S. Mattick. The relationship between non-protein-coding DNA and eukaryotic complexity. *Bioessays*, 29(3):288–99, March 2007.

[123] T. H. Tang, J. P. Bachellerie, T. Rozhdestvensky, M. L. Bortolin, H. Huber, M. Drungowski, T. Elge, J. Brosius, and A. Huttenhofer. Identification of 86 candidates for small non-messenger rnas from the archaeon archaeoglobus fulgidus. *Proc Natl Acad Sci U S A*, 99(11):7536–41, May 28 2002.

[124] J.D. Thompson, D.G. Higgins, and T.J. Gibson. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice. *NAR*, 22, 1994.

[125] Ignacio Jr Tinoco and Carlos Bustamante. How RNA folds. *J.Mol.Biol.*, 293:271–281, October 1999.

[126] K. T. Tycowski, A. Aab, and J. A. Steitz. Guide RNAs with 5' caps and novel box c/d snoRNA-like domains for modification of snrnas in metazoa. *Curr Biol*, 14(22):1985–95, Nov 23 2004.

[127] Olke C. Uhlenbeck, Francis H. Martin, and Paul Doty. Self complimentary oligoribonucleotides: Effects of helix defects and guanylic acid-cytidylic acid base pairs. *JMB*, 57:217–229, 1971.

[128] S. Washietl, I. L. Hofacker, M. Lukasser, A. Huttenhofer, and P. F. Stadler. Mapping of conserved RNA secondary structures predicts thousands of functional noncoding rnas in the human genome. *Nat Biotechnol*, 23(11):1383–90, November 2005.

[129] S. Washietl, I. L. Hofacker, and P. F. Stadler. Fast and reliable prediction of noncoding rnas. *Proc Natl Acad Sci U S A*, 102(7):2454–9, Feb 15 2005.

[130] J. D. Watson and F. H. Crick. The structure of DNA. *Cold Spring Harb Symp Quant Biol*, 18:123–31, 1953.

[131] James D. Watson. The involvement of RNA in the synthesis of proteins. Nobel Lecture, 1962.

[132] E. Westhof and C. Massire. Structural biology. evolution of RNA architecture. *Science*, 306(5693):62–3, Oct 1 2004.

[133] S. Will, K. Reiche, I. L. Hofacker, P. F. Stadler, and R. Backofen. Inferring noncoding rna families and classes by means of genome-scale structure-based clustering. *PLoS Comput Biol*, 3(4):e65, Apr 13 2007.

[134] A. Wilm, I. Mainz, and G. Steger. An enhanced RNA alignment benchmark for sequence alignment programs. *Algorithms Mol Biol*, 1:19, 2006.

[135] Peng Wu, Shu-ichi Nakano, and Naoki Sugimoto. Temperature dependence of thermodynamic properties for DNA/DNA and RNA/DNA duplex formation. *Eur. J. Biochem.*, 269:2821–2830, 2002.

[136] S Wuchty, W Fontana, I L Hofacker, and P Schuster. Complete suboptimal folding of RNA and the stability of secondary structures. *Biopolymers*, 49:145–165, 1999.

[137] S. Wuchty, W. Fontana, I. L. Hofacker, and P. Schuster. Complete suboptimal folding of RNA and the stability of secondary structures. *Biopolymers*, 49(2):145–65, 1999.

[138] T. Xia, , M. E. Burkard, R. Kierzek, S. J. Schroeder, X. Jiao, C. Cox, and D. H. Turner. Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with watson-crick base pairs. *Biochemistry*, 37(42):14719–35, 1998.

[139] X. Zhuang. Single-molecule RNA science. *Annu Rev Biophys Biomol Struct*, 34:399–414, 2005.

[140] M. Zuker. On finding all suboptimal foldings of an rna molecule. *Science*, 244(4900):48–52, Apr 7 1989.

[141] M Zuker and P Stiegler. Optimal computer folding of larger RNA sequences using thermodynamics and auxiliary information. *Nucl. Acids Res.*, 9:133–148, 1981.