The Folding Kinetics of RNA

From Folding Simulations to the
Design of Synthetic Molecules

Von der Fakultät für Mathematik und Informatik
der Universität Leipzig
angenommene

DISSERTATION

zur Erlangung des akademischen Grades

Doctor rerum naturalium
(Dr. rer. nat.)

im Fachgebiet

Informatik,

vorgelegt von

Herrn Felix Kühnl, M. Sc. Bioinformatik,
geboren am 20.03.1992 in Leipzig.

Die Annahme der Dissertation wurde empfohlen von:

1. Prof. Dr. Peter F. Stadler, Universität Leipzig
2. Prof. Dr. Yann Ponty, École Polytechnique, Frankreich

Die Verleihung des akademischen Grades erfolgt mit Bestehen
der Verteidigung am 09.11.2022 mit dem Gesamtprädikat

„magna cum laude“.

Bibliographic Description

Title:	The Folding Kinetics of RNA
Subtitle:	From Folding Simulations to the Design of Synthetic Molecules
Type:	Dissertation
Author:	Felix Kühnl
Year:	2022
Professional discipline:	Bioinformatics
Language:	English
Pages in the main part:	106
Chapters in the main part:	6
Number of Figures:	37
Number of Tables:	4
Number of Appendices:	2
Number of Citations:	151
Key Words:	RNA folding kinetics, synthetic riboswitches, RNA design, energy landscapes, neomycin riboswitch

This thesis is based on the following publications:

S. Findeiß, S. Hammer, M. T. Wolfinger, F. Kühnl, C. Flamm, and I. L. Hofacker (2018). “In silico design of ligand triggered RNA switches”. In: *Methods* 143. Methods and advances in RNA characterization and design, pp. 90–101. DOI: `10.1016/j.ymeth.2018.04.003`.

C. Günzel*, F. Kühnl*, K. Arnold, S. Findeiß, C. E. Weinberg, P. F. Stadler, and M. Mörl (2020). “Beyond Plug and Pray: Context Sensitivity and in silico Design of Artificial Neomycin Riboswitches”. In: *RNA Biology*, pp. 1–11. DOI: `10.1080/15476286.2020.1816336`.

F. Kühnl, P. F. Stadler, and S. Findeiß (2019). “Assessing the Quality of Cotranscriptional Folding Simulations”. In: *RNA Design*. Ed. by R. Lorenz. Methods in Molecular Biology. Manuscript accepted for publication. Berlin: Springer Nature.

---

*The authors share first authorship.

Abstract

RNAs are biomolecules ubiquitous in all living cells. Usually, they fold into complex molecular structures, which often mediate their biological function. In this work, models of RNA folding have been studied in detail. While an analysis of the three-dimensional, or *tertiary*, structure of RNAs is difficult, a close approximation is achieved by resorting to the notion of *secondary* structures. In this model, each nucleobase may engage in at most one base pair; and any two base pairs must be either "parallel" to, or nested into each other. "Crossing" base pairs are not allowed. Every structure is assigned a Gibbs free energy quantifying its stability using the nearest-neighbor energy model. The set of all possible structures of a given sequence is called its *ensemble*.

One can distinguish two fundamentally different approaches to RNA folding. The first one is the *thermodynamic* approach, which yields information about the distribution of structures in the ensemble *in its equilibrium*. Since elementary folding reactions of RNA structures happens on the microsecond timescale, this assumption is often reasonable. The second approach, which is required to study the dynamics of folding during the course of time, is the *kinetic* folding analysis. It is much more computationally expensive, but allows to incorporate changing environmental parameters as well as time-dependent effects into the analysis.

In the thermodynamic framework, the structures of a given RNA sequence are assumed to follow a Boltzmann distribution. Even for long RNA molecules, their *partition function* – the sum of all structures' Boltzmann weights – can efficiently be computed using dynamic programming. The probability $\Pr[Y]$ of any set Y of structures can thus easily be determined and is also referred to as the *coverage* of Y with respect to the ensemble. It measures to which extent the structures of Y cover those of a random sample from the full ensemble, and thus how representative Y is.

Kinetic simulations, on the other hand, consider the transitions between structural states. Since the number all possible structures is enormous even for short sequences, it is necessary to greatly reduce the number of states. This can be achieved by various established methods that discard supposedly unimportant structures, or aggregate multiple structures into a smaller set of macrostates. However, a major problem with these heuristics is that it is unclear to which extent they alter the results of the simulation. To alleviate this issue, the concept of coverage was employed to predict the quality of the models after the application of the heuristics. This method offers researchers a reliable criterion for choosing parameters and to quantify the credibility of the computation.

Building on the methods above, the *BarMap* framework (Hofacker, Flamm, et al., 2010) allows to chain several pre-computed models and thus simulate

folding reactions in a dynamically changing environment, e. g., to model cotranscriptional folding. However, there is no obvious way to identify spurious output, let alone assessing the quality of the simulation results. Additionally, the implementation of *BarMap* is prototypical, simplistic, and very general, such that it is laborious and cumbersome to apply and to evaluate the results. As a remedy, *BarMap-QA*, a semi-automatic software pipeline for the analysis of cotranscriptional folding, has been developed. For a given input sequence, it automatically generates the models for every step of the RNA elongation, applies *BarMap* to link them together, and runs the simulation. Post-processing scripts, visualizations, and an integrated viewer are provided to facilitate the evaluation of the unwieldy *BarMap* output. Three novel, complementary quality measures are computed on-the-fly, allowing the analyst to evaluate the coverage of the computed models, the exactness of the computed mapping between the individual states of each model, and the fraction of correctly mapped population during the simulation run. In case of deficiencies, the output is automatically re-rendered after parameter adjustment. The pipeline is provided for download as free and open source software. A *Docker* image including *BarMap-QA* and all required dependencies is publicly available via *Docker Hub* for zero configuration deployments.

Statistical evidence is presented that, even when coarse graining the ensemble, kinetic simulations quickly become infeasible for longer RNAs. One reason is that, to compute the macrostates of the coarse graining, all secondary structures up to a global enumeration threshold have to be generated first. However, within the individual gradient basins, most high-energy structures only have a marginal probability and could safely be excluded from the analysis. To tell relevant and irrelevant structures apart, a precise knowledge of the distribution of probability mass within a basin is necessary. Both a theoretical result concerning the shape of its density, and possible applications like the prediction of a basin's partition function are given.

To demonstrate the applicability of computational folding simulations to a real-world task of the life sciences, we conducted an *in silico* design process for a synthetic, transcriptional *riboswitch* responding to the ligand neomycin. Riboswitches are small, regulatory RNAs located in the $5'$ untranslated region of some genes. The designed riboswitch was then transfected into the bacterium *Escherichia coli* by a collaborative partner and could successfully regulate a fluorescent reporter gene depending on the presence of its ligand. Additionally, it was shown that the sequence context of the riboswitch could have detrimental effects on its functionality, but also that RNA folding simulations are often capable to predict these interactions and provide solutions in the form of decoupling spacer elements.

Taken together, this thesis offers the reader deep insights into the world of RNA folding. It provides statistical analyses and results concerning the distributions of energies and probabilities of structures. Existing methods to conduct *in silico* RNA folding analyses were extended, and novel approaches to quantify the quality of folding simulations were developed and implemented for immediate application. They were then applied to design a synthetic biomolecule, which was shown to successfully regulate the expression of a reporter gene.

Zusammenfassung

RNAs sind universelle, in allen lebenden Zellen anzufindende Biomoleküle. In der Regel falten sie sich zu komplexen molekularen Strukturen auf, welche dann ihre biologische Funktion vermitteln. In der vorliegenden Arbeit wurden Modelle der RNA-Faltung gründlich untersucht. Während eine Analyse der dreidimensionalen, oder *tertiären*, Struktur einer RNA im Allgemeinen schwierig ist, kann eine gute Näherung erreicht werden, indem man sich auf das Konzept der *Sekundärstrukturen* bezieht. In diesem Modell kann jede Nukleobase nur Bestandteil eines einzigen Basenpaars sein, und je zwei Basenpaare müssen entweder „parallel“ oder ineinander verschachtelt sein. Sich „kreuzende“ Basenpaare sind nicht gestattet. Jeder Struktur wird durch Anwendung des „Nächste-Nachbarn“-Energiemodells (Mathews, Turner und Zuker, 2007) eine Gibbs-Energie zugeordnet, welche ihre Stabilität bemisst. Die Menge aller möglichen Strukturen einer gegebenen RNA-Sequenz wird als ihr *Ensemble* bezeichnet.

Man kann zwei grundverschiedene Herangehensweisen zur Modellierung der RNA-Faltung unterscheiden. Die erste ist der *thermodynamische* Ansatz, welcher Informationen über die Verteilung der Strukturen im Ensemble liefert, sofern sich letzteres in seinem Gleichgewichtszustand, oder *Equilibrium*, befindet. Da die elementaren Faltungsreaktionen von RNA-Strukturen auf der Zeitskala von Mikrosekunden stattfinden, ist diese Annahme oftmals gerechtfertigt. Der zweite Ansatz ist die *kinetische* Analyse der RNA-Faltung, welcher angewendet werden muss, wenn die Dynamik des Faltungsprozesses über einen bestimmten Zeitraum studiert werden soll. Derartige Methoden sind wesentlich rechenintensiver, erlauben dafür aber die Berücksichtigung veränderlicher Umgebungsbedingungen sowie weiterer zeitabhängiger Effekte in der Analyse.

Im thermodynamischen Kontext wird angenommen, dass die Strukturen einer gegebenen RNA-Sequenz einer Boltzmann-Verteilung folgen. Die Zustandssumme – also die Summe der Boltzmann-Gewichte aller Strukturen – kann dann durch die Anwendung dynamischer Programmierung selbst für lange RNA-Moleküle effizient berechnet werden. Die Wahrscheinlichkeit $\Pr[Y]$ einer beliebigen Menge Y von Strukturen kann somit einfach bestimmt werden und wird in dieser Arbeit auch als *Abdeckung* von Y bezüglich des Ensembles bezeichnet. Sie misst, zu welchem Grad die Strukturen aus Y die aus einer zufälligen Stichprobe aus dem Ensemble abdecken, und somit wie repräsentativ Y für das Ensemble ist.

Kinetische Simulationen hingegen betrachten die Übergänge zwischen einzelnen strukturellen Zuständen. Da jedoch selbst kurze Sequenzen eine enorme Anzahl möglicher Strukturen aufweisen, ist es für eine Simulation notwendig, die Anzahl der Zustände drastisch zu reduzieren. Das kann durch die Anwendung verschiedener etablierter Methoden erreicht werden, welche vermeintlich

unwichtige Strukturen aussortieren oder mehrere Strukturen zu einer kleineren Menge von Makrozuständen zusammenfassen. Ein erhebliches Problem bei der Anwendungen solcher Heuristiken ist jedoch der Umstand, dass unklar ist, in welchem Ausmaß sie Auswirkungen auf den Ausgang der Simulation haben. Um Abhilfe zu schaffen wurde in dieser Arbeit das Konzept der Abdeckung eingesetzt, um die Qualität der durch die Anwendung der Heuristiken erzeugten Modelle zu beurteilen. Diese Vorgehensweise bietet dem Forscher die Möglichkeit, geeignete Modellparameter zu wählen und die Validität der durchgeführten Berechnung zuverlässig zu quantifizieren.

Aufbauend auf den beschriebenen Methoden erlaubt *BarMap* (Hofacker, Flamm u. a., 2010) die Verkettung mehrerer zuvor berechneter Modelle und ermöglicht dadurch die Simulation von Faltungsreaktionen unter sich dynamisch verändernden Umgebungsbedingungen, z. B. zur Modellierung von kotranskriptionellem Falten. Es bietet jedoch keinerlei Möglichkeiten, um fehlerhafte Ergebnisse zu erkennen oder gar die Qualität der Simulation genauer beurteilen zu können. Hinzu kommt, dass die Implementierung von *BarMap* eher prototypischer Natur und sehr allgemein gehalten ist, sodass sowohl die Anwendung als auch die Auswertung der Ergebnisse aufwendig und umständlich ist. Zur Lösung dieser Probleme wurde *BarMap-QA* entwickelt, eine halbautomatische Software-Pipeline zur Analyse der kotranskriptionellen Faltung von RNA. Für eine gegebene RNA-Sequenz erzeugt sie automatisch je ein Modell für jeden Schritt der RNA-Elongation, wendet *BarMap* zu deren Verknüpfung an und führt dann die Simulation durch. Skripte zur Nachbearbeitung und Visualierung der Daten werden ebenso mitgeliefert wie ein integrativer Ergebnis-Betrachter, wodurch die Auswertung der umfangreichen Ausgaben von *BarMap* erheblich erleichtert wird. Drei neuartige, sich gegenseitig ergänzende Qualitätsmaße werden automatisch berechnet und erlauben dem Analysten, die Abdeckung der generierten Modelle, die Exaktheit der konstruierten Abbildungen zwischen den einzelnen Zuständen der Modelle, sowie den Anteil der während der Simulation korrekt abgebildeten Populationen zu bewerten. Im Falle von Defiziten werden die Ergebnisse nach einer Anpassung der Simulationsparameter automatisch regeneriert. Die Pipeline wird als freie, quelloffene Software zum Download angeboten. Ein Docker-Image, welches *BarMap-QA* und alle notwendigen Abhängigkeiten enthält, wurde auf *Docker Hub* öffentlich verfügbar gemacht und ermöglicht eine konfigurationslose Bereitstellung auf den meisten gängigen Plattformen.

Es werden statistische Nachweise erbracht die zeigen, dass kinetische Simulationen für längere RNAs selbst bei der Anwendung von Methoden zur Vergröberung des Ensembles oft nicht mehr durchführbar sind. Ein Grund dafür ist die Tatsache, dass zur Berechnung der Makrozustände im Zuge der Vergröberung zunächst alle Sekundärstrukturen bis hinauf zu einem globalen Enumerierungsschwellwert erzeugt werden müssen. Da die meisten hochenergetischen Strukturen innerhalb eines einzelnen Gradientenbassins nur mit äußerst geringer Wahrscheinlichkeit auftreten, könnten sie problemlos von der Analyse ausgeschlossen werden. Um jedoch wichtige von unwichtigen Strukturen unterscheiden zu können, sind genaue Kenntnisse der Verteilung der Wahrscheinlichkeitsmasse innerhalb eines Bassins vonnöten. Sowohl ein theoretisches Resultat zur Form dieser Verteilung, als auch mögliche Anwendungen wie die Schätzung der Zustandssumme eines Bassins werden vorgestellt.

Um die Anwendbarkeit computergestützter Faltungssimulationen auf eine realistische Problemstellung der Lebenswissenschaften zu demonstrieren, wurde *in silico* ein synthetischer, transkriptioneller *Riboswitch* entworfen, der durch den Liganden Neomycin gesteuert wird. Riboswitche, oder RNA-Schalter, sind kleine regulatorische RNAs, welche in der untranslatierten Region (UTR) am $5'$-Ende mancher Gene zu finden sind. Der entworfene Riboswitch wurde durch einen Kollaborationspartner in einen Stamm des Bakteriums *Escherichia coli* transfiziert und konnte dort erfolgreich ein fluoreszierendes Reportergen abhängig von der Anwesenheit seines Liganden regulieren. Zusätzlich wurde gezeigt, dass der Sequenzkontext des RNA-Schalters seine Funktionsfähigkeit erheblich vermindern kann, jedoch auch, dass Faltungssimulationen diese Interaktionen oft vorhersagen können und dabei helfen, Lösungen in Form von entkoppelnden Trennelementen zu entwickeln.

Alles in allem gewährt diese Arbeit dem Leser tiefe Einblicke in die Welt der RNA-Faltung. Sie beinhaltet statistische Analysen und Resultate zu Verteilungen von Energien und Wahrscheinlichkeiten von Strukturen. Existierende Methoden zur Durchführung von RNA-Faltungssimulationen wurden erweitert, und neuartige Wege zur Quantifizierung der Simulationsqualität wurden erdacht und zur direkten Anwendung implementiert. Sie wurden dann genutzt, um ein synthetisches Biomolekül zu entwerfen, welches nachweislich in der Lage war, die Expression eines Reportergens zu steuern.

Acknowledgments

Thanks to all who supported me during my doctoral studies. I cannot name all of you, for you are many. Thanks to Peter for giving me the opportunity to work in our wonderful institute. Thanks to all of my colleagues; especially to Sven, for all his scientific, administrative and personal support – I owe you. Thanks to all who I enjoyed talking to and laughed with. Thanks to all who helped proofreading this thesis (poor you!). Thanks to Fabian for his great thesis template. Thanks to Petra, Jens and Steve for both the personal support and keeping the institute going. Thanks to Berni for not piercing me with a dart. Thank you all for the funny lunch breaks, delicious cakes, boozy Beyerhaus evenings, sunny BBQs and cheerful Christmas parties.

Thanks to my friends and, of course, my family: my wife and my kids, but also my parents and parents-in-law, who help us out so often.

Contents

1 Introduction **2**

2 Background **6**
- 2.1 The biology of nucleic acids 6
- 2.2 The thermodynamics of RNA folding 10
- 2.3 Structure in motion: the folding kinetics of RNA 17
- 2.4 The RNA design problem 27

3 Assessing the Quality of RNA Folding Models **30**
- 3.1 Coverage: representative subsets of structures 30
- 3.2 The coverage of low-energy bands 31
- 3.3 Canonical energy landscapes 32
- 3.4 *BarMap-QA*: cotranscriptional folding with quality assurance . 35
- 3.5 Conclusion 53

4 Statistics of Free Energies for Sequences, Structures, and Gradient Basins **56**
- 4.1 The distribution of minimum free energies of random sequences 56
- 4.2 The statistics of canonical structures 63
- 4.3 The statistics of low-energy minima 65
- 4.4 The distribution of free energies within gradient basins 68
- 4.5 The distribution of probability mass in gradient basins 71
- 4.6 Conclusion 76

5 The Design of Artificial Cotranscriptional Riboswitches **80**
- 5.1 Biological background 80
- 5.2 Design and analysis of transcriptional neomycin-dependent riboswitches 86
- 5.3 Materials and methods 86
- 5.4 Results 89
- 5.5 Discussion 95

6 Conclusion **100**
- 6.1 Summary 100
- 6.2 Outlook 106
- 6.3 Concluding remarks 106

Appendices 107

A SI: Design of Artificial Cotranscriptional Riboswitches 109
A.1 Supplemental figures 110
A.2 Supplemental data 114

B SI: Assessing the quality of cotranscriptional folding models 117
B.1 Table of output files 118

List of Abbreviations 119

Bibliography 121

Curriculum Scientiae 133

CHAPTER 1

Introduction

If there is one thing that has been pushing humanity forward, it is the natural curiosity that keeps us exploring the world around us, looking for the answers to all questions that can possibly be asked. For a long time, our ability to explore the world was limited by what our senses could perceive. With the advent of new technologies, however, we were able to push the boundaries of what can be observed, and the unravelling of previously inexplicable mysteries came into reach. While the invention of telescopes in the early 17th century enabled us to catch a glimpse of an endless universe, in 1665 the precursor of the modern microscope allowed Robert Hooke, for the first time, to see the elementary building block of life, for which he coined the term *cell*. Ever since, microbiology seeks to answer the question how the astonishing complexity of life as we know it can emerge from a structure this tiny. Today, we know that the entire hereditary information of a cell – the blueprint of life – is stored in the form of DNA, the double-helical shape of which has become iconic even in popular culture. We know that genes encode for various biomolecules like RNAs and proteins, each serving a specific purpose, thus keeping up the cell's metabolism. However, many details remain in the dark, and there is still so much we do not know yet.

One aspect that seems especially paradoxical is that, despite their obvious dissimilarity, almost all cells of complex organisms share the exact same genetic material. If proteins are the major players driving so many cellular processes, one could expect that cells with the same set of genes exhibit the same phenotype. This paradox can be explained by the existence of a vast number of diverse mechanisms that, in concert, tightly control the expression of each and every gene. As a result of this gene regulation, the proportions of the individual gene products are precisely balanced to enable the cell to efficiently perform specific tasks, to proliferate and also to cope with ever-changing environmental conditions. Examples for gene-regulatory mechanisms include methylation of DNA (Singal and Ginder, 1999), and variable rates of transcription initiation, translation, and degradation of transcripts (Timmers and Tora, 2018). In eukaryotes, there are even more regulatory mechanisms: various chemical modifications of histones, onto which the DNA is rolled up, are known to promote or inhibit the expression of proximal genes (Bannister and Kouzarides, 2011). Alternative splicing allows the cell to produce different proteins from the same gene depending on the presence or absence of regulating proteins (Kelemen et al., 2013). As a concluding example, microRNAs, in conjunction with the RNA-induced silencing complex (RISC), cleave specific target transcripts and thus silence the corresponding gene (Tijsterman and Plasterk, 2004).

Obviously, the surroundings of a cell may vary rapidly, e. g., due to changes in temperature, the availability of nutrients or oxygen, or external signals in multicellular organisms (Hancock, 2017). To survive and prosper, living organisms must adapt to those changes quickly. On the level of the cell, this often means that a different set of gene products is required to be expressed, e. g., heat shock proteins when the temperature rises, or lactase when the glucose level is low but lactose is available as a nutrient. Under normal conditions, the expression of these specialised proteins would be a waste of resources at best, and at worst they would even have detrimental effects. Thus, the rates of transcription and, in the case of proteins, translation of the respective gene products must be able to immediately respond to environmental stimuli. Also, if a very quick response is needed, there may be no time to start transcription only when the gene product is required; the transcript should possibly be synthesized already, but its translation should be delayed.

A specific example of how this can be achieved is the synthesis of the above-named heat shock proteins. Their mRNA transcripts often contain so-called ROSE elements in their $5'$ untranslated region (UTR). ROSE stands for "repression of heat shock gene expression", and as the name suggests, these elements prevent the transcript from being translated, suppressing the expression of the protein encoded by the open reading frame downstream (Krajewski and Narberhaus, 2014). ROSE elements consist of several structural components called hairpin loops, the last of which is believed to sequester the ribosomal binding site (RBS) and thus prevent the initiation of translation. These hairpins are sensitive to temperature changes. If the cell heats up, they destabilize and the RBS becomes accessible to the ribosome, which in turn starts the translation of the heat shock protein now required. ROSE elements belong to a bigger class of regulatory elements referred to as RNA thermometers, which are characterized by their structural response to temperature changes leading to a change in the expression of a gene. The existence of RNA thermometers stresses the importance of RNA secondary structure for gene regulation. It is thus obvious that a thorough understanding of the RNA folding process is required to fully comprehend at least some of the regulatory mechanisms.

While RNA thermometers can only respond to changes in temperature, other classes of RNAs are capable to recognize other stimuli as well. Riboswitches are small regulatory elements that are able to regulate the expression of a specific gene in response to the presence or absence of another small molecule, referred to as its *ligand*. Exploiting the fact that many substances can bind to RNA, a binding pocket specifically matching the ligand is formed, which is then stabilized if it is actually bound. In the absence of the ligand, the riboswitch adopts an alternative structural conformation, which will in turn regulate the controlled gene. This simple yet effective mechanism allows the cell to quickly respond to either internal or external signals as soon as the concentration of the ligand in the cell rises. In fact, natural riboswitches are abundant in bacteria, and their diverse ligands show the flexibility of this way to affect gene expression. Also note that, as in the case of ROSE elements, the change of structure of the RNA is mediating its function. Thus, a thorough knowledge and powerful tools for simulating RNA folding are they key to understanding the underlying mechanism of action not only of riboswitches, but for many classes of non-coding RNAs.

While the study of natural riboswitches is highly interesting by itself, in-depth knowledge of their mechanics gives rise to even more exciting opportunities: using both natural and artificially designed components, synthetic riboswitches can be engineered to respond to almost any ligand and to control any gene of interest. By choosing multiple foreign, non-toxic ligands, orthogonal switches can be constructed, controlling multiple genes independently, which can then be regulated by the researcher by simply changing the ligand concentration in the medium. Beside possible applications in research, riboswitches may also be used as detector for various substances. If the riboswitch is coupled with a fluorescent reporter gene, the presence of the ligand can readily be observed visually. Such systems could be used in situations where the laboratory equipment required for other means of detection are not available, e. g., during a field study in a remote area (Sahu, Roy, and Anand, 2022).

Reflecting on these examples, the central role of RNA structure to mediate its function on many different levels becomes obvious. This work will therefore focus on explaining techniques that empower researchers to elucidate the mysteries of RNAs. To this end, the realm of RNA structures will be analyzed thoroughly, and models and criteria characterizing their behaviour will be presented. It will also be shown how these techniques can be applied in practice to design novel molecules in an effective an cost-efficient manner, minimizing laborious and expensive trial-and-error procedures by precise computational predictions. This way, the author hopes to contribute a tiny piece to the huge puzzle that mankind tries to solve ever since the dawn of our species: the quest for understanding the world we live in. It is a long journey we are on; a journey that will never end, but which defines who we are.

This thesis is organized as follows. The reader is familiarized with the biology of RNA folding, its thermodynamic and kinetic properties, and important existing models in Chapter 2. In Chapter 3, results concerning the quality control of RNA folding simulations are presented. Then, Chapter 4 presents an in-depth statistical analysis of the distributions of structures, energies, and their probabilities. In Chapter 5, finally, the techniques presented previously are applied to design synthetic riboswitches and predict their behaviour in the cell.

CHAPTER 2

Background

Contents

2.1 The biology of nucleic acids **6**
  2.1.1 DNA: information storage of living beings 6
  2.1.2 RNA: both messengers and workhorses 6
  2.1.3 The sequence of RNA molecules 7
  2.1.4 The synthesis of RNA molecules 7
  2.1.5 Base pairs structure RNAs 9
2.2 The thermodynamics of RNA folding **10**
  2.2.1 Of primary, secondary and tertiary structures . . 10
  2.2.2 Quantifying stability: the Gibbs free energy . . . 12
  2.2.3 The minimum free energy structure 13
  2.2.4 The equilibrium probabilities of structures 13
  2.2.5 Numerical considerations for the computation of partition functions 15
  2.2.6 Suboptimal RNA structures 16
2.3 Structure in motion: the folding kinetics of RNA **17**
  2.3.1 Refolding as a sequence of elementary transition reactions 17
  2.3.2 RNA energy landscapes 17
  2.3.3 The transition rates of RNA folding 18
  2.3.4 Coarse graining energy landscapes 19
  2.3.5 Transition rates for macrostates 22
  2.3.6 Simulating first-order reaction kinetics for RNA folding 23
  2.3.7 A model for cotranscriptional folding 24
  2.3.8 Stochastic simulations of folding kinetics 26
2.4 The RNA design problem **27**

This chapter introduces the reader to the basic concepts and terminology used in the remainder of the thesis. The topics include genetics, the chemistry of biomolecules, the thermodynamics of RNA structures, the kinetics of RNA folding, and important models and algorithm for them.

Serving as an introduction to readers not familiar with the matter of the following chapters, this part of the thesis does not present original research of the author. The presented information is mostly common knowledge as found in standard text books and current reviews of the life sciences and its journals. If a passage of text is based on information from a single reference, it is cited only once.

2.1 The biology of nucleic acids

This section provides general biological information about DNA and RNA that is referred to, explicitly or implicitly, in the other parts of this work. The presented information is, to a great extent, common knowledge and can be found in any standard text book of the field, e. g., in Alberts, B. (2022). *Molecular biology of the cell.* Seventh edition. New York: W. W. Norton & Company. ISBN: 978-0-393-88482-1.

2.1.1 DNA: information storage of living beings

Deoxyribonucleic acid, commonly referred to as DNA, is probably the most renowned biomolecule and found in all known forms of live. It can be considered as a storage of blueprints that organisms use to make all the components required to keep up their metabolism and to proliferate. More precisely, the DNA comprises the *genome* of a cell, i. e., the sum of all its genetic make-up. It encodes for proteins and functional RNAs, both of which fulfill and assist numerous important tasks such as catalysis and regulation of biochemical reactions, signalling, transport etc., or serve as building block for cellular compartments. In the cell, DNA occurs as a stable double helix consisting of two strands storing complementary information in the form of different nucleotides. Specific locations (*loci*) on these strands that encode for a certain RNA product are called *genes.* These serve as templates for the synthesis of RNA molecules, as we will see in the next section. Each cell of an organism has an exact copy of the same DNA. When a cell divides, the DNA is replicated with extremely low error rates. This precise replication and the stability are the properties that make DNA so well-suited to safely store the genome of a cell.

2.1.2 RNA: both messengers and workhorses

Ribonucleic acids, or RNAs, fulfill at least two critical tasks in a cell. Firstly, as mRNA, they act as a “messenger” to transport information from a gene to the ribosome, which *translates* them into proteins, which are the major workhorses of the cell fulfilling numerous tasks. Secondly, RNAs may, directly or indirectly, act in the cell without being translated. These are referred to as *non-coding RNAs.* The most prominent examples of this abundant and versatile class include ribosomal RNAs, which constitute, together with other proteins, the ribosome that translates mRNAs to proteins; tRNAs, which match amino acids to their respective nucleotide triplet during translation; ribozymes, which act

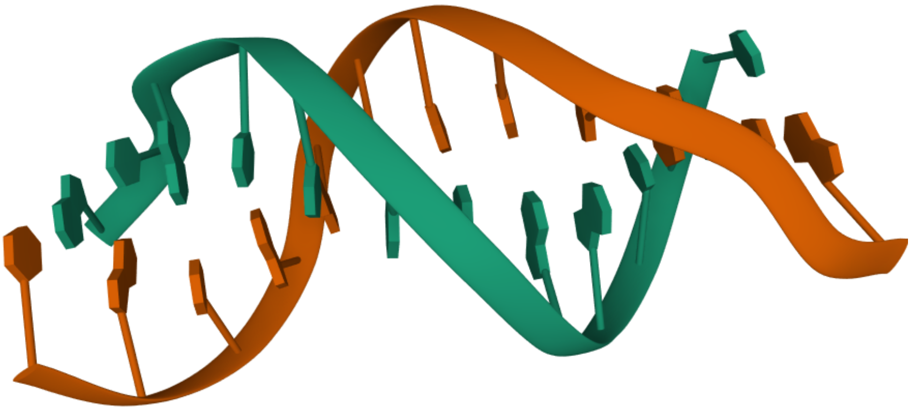


Figure 1: Three-dimensional crystal structure of the DNA double helix of the 12-mer 1BNA (Berman et al., 2000; Drew et al., 1981). The two strands are colored orange and green. The flat bands represent the backbone, and the individual nucleotides are depicted as polygonal shapes.

as catalysts; and small RNAs as well as long non-coding RNAs, which beside other tasks regulate the transcription and translation of mRNAs. While the aforementioned types are usually transcribed from dedicated loci referred to as RNA genes, there are also smaller functional elements that are part of other transcripts. For example, riboswitches and RNA thermometers are usually found at the beginning of mRNA transcripts and regulate their *expression*, i. e., transcription and translation, in response to external stimuli.

RNAs are the major topic of this work. In the following sections, their properties and behaviour will be introduced thoroughly.

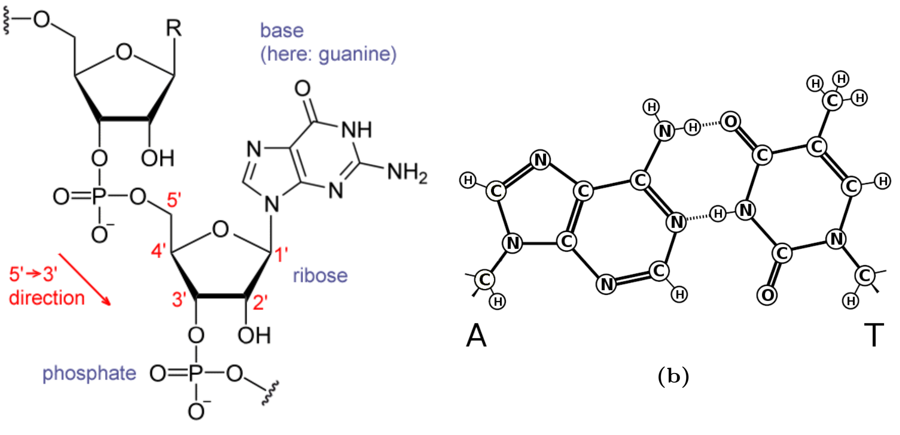
2.1.3 The sequence of RNA molecules

RNA is a chain-like molecule consisting of ribose (i. e., sugar) molecules linked together by phosphate groups, comprising the ribose–phosphate *backbone* of molecule. In RNA, the D-ribose occurs in its β-furanose form, i. e., as a ringlike molecule of five carbon atoms, which are consecutively numbered and referred to as $1'$ to $5'$. The phosphate group always links a $5'$ with a $3'$ carbon; the RNA chain thus has a *direction*: it is read from its $5'$ to its $3'$ end, cf. Figure 2.

To each ribose molecule, exactly one of the four nucleobases adenine, uracil, guanine, or cytosine – abbreviated as A, U, G, and C, respectively – is attached. A nucleobase and the ribose it is bound to are together referred to as a *nucleoside*. A nucleoside including a phosphate group, as in the backbone of an RNA molecule, is called a *nucleotide*. By reading the attached bases of an RNA molecule in $5'$ to $3'$ direction, one obtains the *sequence* information.

2.1.4 The synthesis of RNA molecules

RNAs are *transcribed*, i. e., synthesized from a DNA template, by a group of enzymes referred to as RNA polymerases (RNAPs). To this end, an RNAP attaches to a region at the beginning of the gene called the *promoter*, and unwinds a short section of the DNA double helix to form the *transcription*



(a)

(b)

Figure 2: The chemical structure of RNA and base pairs. (*a*) Two ribose molecules (pentagonal rings) molecules connected by a phosphate group (P). For the lower ribose molecule, the nucleobase is drawn, too – the purine guanine (two connected aromatic rings) in this case. (*b*) A Watson–Crick base pair of adenine (A) and thymine (T) as found in DNA. It consists of two hydrogen bonds (H).

(*a*): `en.wikipedia.org/wiki/File:RNA_chemical_structure.GIF` [2022-05-09]
(*b*): `commons.wikimedia.org/wiki/File:Hoogsteen_Watson_Crick_pairing-en.svg` [2022-05-09]

bubble. This process is referred to as the *initiation* of transcription. Next, the *elongation* of the RNA molecule begins: RNAP starts to slide over the DNA template (or *antisense*) strand in 3′ to 5′ direction. It reads one of the four nucleobases – adenine, thymine, guanine, and cytosine – from the template, and for each it appends a *complementary* nucleotide to the newly synthesized molecule in 5′ to 3′ direction. The complement of guanine is cytosine and *vice versa*; the complement of thymine is adenine, and thymine is replaced by uracil and thus not present in RNAs. The synthesized RNA thus matches the other (*coding* or *sense*) strand, except for the exchange of thymine by uracil. This process continues until the *termination* is triggered, and the DNA template, RNAP, and the newly synthesized RNA dissociate. In bacteria, termination can either be mediated by a protein called ρ factor, or triggered by a signal in the sequence itself. The latter case is called *intrinsic termination*, and is of great importance for the synthetic RNAs designed in Chapter 5. In eukaryotes, i. e., organisms whose cell have a nucleus, termination is not yet understood very well (Nielsen, Yuzenkova, and Zenkin, 2013). Well-known, however, is that they extensively post-process their transcripts by splicing, capping, and polyadenylation.

While in prokaryotes – i. e., bacteria and archaea – the DNA is located directly in the cytoplasm, it is enclosed by the nucleus in eukaryotes. Consequently, a prokaryotic transcript can interact with other molecules of the cell while it is still being synthesized, but a eukaryotic transcript needs to be fully transcribed, processed and exported from the nucleus first. These specifics have, again, consequences for the design of synthetic RNAs discussed later.

2.1.5 Base pairs structure RNAs

Inside the cell, RNA molecules are in an aqueous solution. Under these conditions, they exhibit a strong tendency to coil up into compact conformations. Many of the nucleobases engage in *base pairing*: both adenine and uracil, and guanine and cytosine are capable of forming hydrogen bonds between each other, thus constituting *Watson–Crick base pairs*. This is similar to the base pairing in DNA, with the exception that adenine pairs with uracil instead of thymine, cf. Figure 2. To some extent, guanine may also pair with uracil, forming a weak *wobble base pair*. Consecutive base pairs then stack on top of each other, hiding the less hydrophilic nucleobases inside a helical structure, the negatively charged sugar–phosphate backbone pointing to the outside, and thus greatly stabilize the molecule. Due to the polar backbone, DNA exhibits a good solubility in water. The stabilizing effect of stacking is attributed to the London dispersion forces arising from the interaction of the π-bonds of the aromatic rings in the nucleobases; the contribution of the hydrogen bonds is considered less relevant as they could also be formed with the solvent (Riley and Hobza, 2013). Like other physical systems, RNAs try to attain a stable, low-energy state, and since increasing the number of base pairs tends to stabilize the molecule, they usually exhibit quite compact structures with as many pairs as possible. In fact, a theoretical study found the expected distance of the $5'$ to the $3'$ end to be less than 6.8 steps along the backbone or any present base pair, even for very long RNAs (Clote, Ponty, and Steyaert, 2012).

Since each nucleobase can engage in a Watson–Crick pair with at most one other base, almost all RNA molecules will have both single-stranded (i. e., unpaired) and double-stranded (i. e., paired) regions under physiological condition. Consecutive double-stranded regions are also called *stems*; in the cell, they take the form of a helix. The *length* of a stem is the number of base pairs it consists of. A single-stranded region enclosed by a stem is referred to as *hairpin loop*, and the number of single-stranded nucleotides enclosed by the stem is the hairpin's *loop length*. A stem may be interrupted by a single-stranded region on one side (a *bulge*) or both sides (an *interior loop*) of the same base pair. A loop region exhibiting more than two stems is referred to as *multi-loop*. The regions of the molecule that are not enclosed by any base pair make up the so-called *exterior loop*; a term that is figurative in the sense that these nucleotides do not actually form a closed loop. Examples of these structural elements are presented in Figure 3.

Beside the standard Watson–Crick base pairs, there are also other, non-canonical types such as Hogsteen or chemically modified base pairs. The various non-modified base pair types arise because the nucleobases possess three edges, namely the Watson–Crick, Hogsteen and sugar edges, and can potentially base pair with each other in various combinations of these (Halder and Bhattacharyya, 2013). Though these may play a role in special cases like RNA triplex formation (Devi et al., 2015) and tRNA functionality (Suzuki, 2021), respectively, they are of minor importance for RNA folding in general and will thus not be considered here.

It should also be noted that other factors like the temperature, metal ions, certain enzymes (e. g., RNA chaperones) or ligands etc. may have a dramatic effect on the ability of RNA to form base pairs. For example, varying the magnesium ion concentration can completely change the structure of an

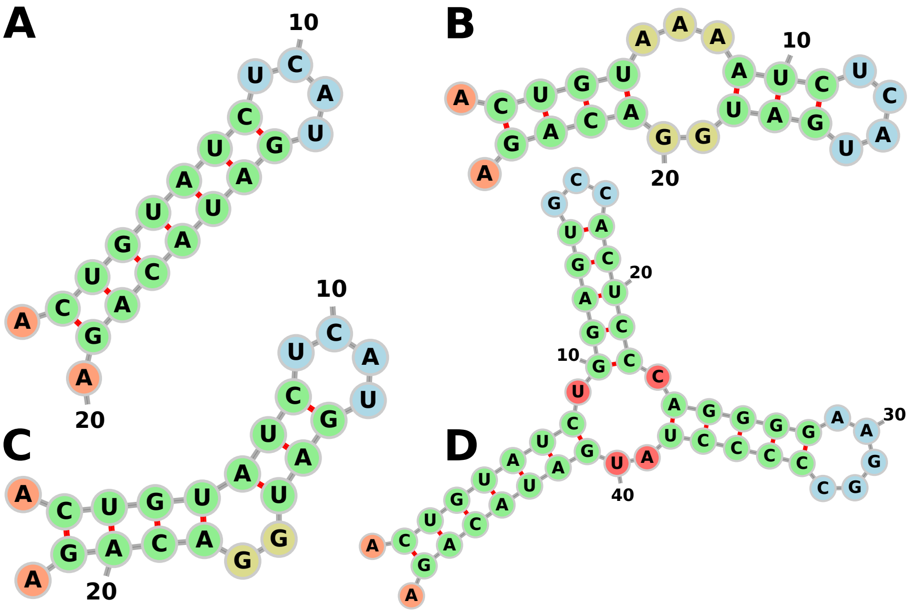


Figure 3: Examples of the various structural elements found in RNA secondary structures. Each letter in a circle denotes a nucleotide. The sequences are given in $5'$ to $3'$ direction as indicated by the numbers. *(A)* A double-stranded stem of seven base pairs (*green*) encloses a single-stranded tetraloop, i. e., a loop of length four (*blue*). Together, these form a hairpin loop. The two dangling nucleotides at position one and 20 (*orange*) belong to the exterior loop. *(B)* The same stem is now disrupted by an interior loop (*yellow*) consisting of three nucleotides on the $5'$ side and two nucleotides on the $3'$ side of the loop. *(C)* Here, the $5'$ part of the interior loop in *(B)* has been removed to form a bulge. Bulges can be interpreted as a special interior loops where either the $5'$ or the $3'$ part has length zero. *(D)* The loop region of the hairpin loop in *(A)* has been extended by two additional hairpin structures to form a multi-loop (*red*). A multi-loop may branch into arbitrarily many, possibly complex substructures.

RNA molecule (Onoa and Tinoco Jr, 2004). When describing the behaviour of biological systems in this work, it is thus assumed that they act under physiological conditions.

2.2 The thermodynamics of RNA folding

After characterizing important biological properties of nucleic acids, we will now put emphasis of how RNAs and their structures can be formalized. Important models and algorithms will be introduced, and the distribution of structures of equilibrated RNAs is described.

2.2.1 Of primary, secondary and tertiary structures

The sequence of an RNA molecule is also referred to as its *primary structure*. It can easily be written as a string over the alphabet $\{A, U, G, C\}$, where the letters denote the base at the corresponding position starting at the $5'$ end. Beside its central role in protein biosynthesis – the sequence encodes the order and type of amino acids of the protein to be synthesized –, it also determines to a great extent the spatial structure of the transcribed RNA molecule: since only some base pairs are energetically feasible, the sequence shapes the space of possible conformations. Thus, it is up to some point possible to reliably predict RNA folding by only analyzing the sequence of the molecule. To that end, a formal representation of RNA structures is necessary.

For a fixed RNA sequence of length n, we can think of an RNA structure x as a set of (valid) base pairs $\{p_1, \ldots, p_m\}$, where $p_k = \{(i, j) \,|\, 1 \leq i + \epsilon < j \leq n\}$ are pairs of indices marking the positions of the nucleobases that pair with each other in the sequence for all $k = 1..m$. ϵ is a parameter defining the minimal loop length, i. e., number of nucleotides enclosed by (i, j). It accounts for the fact that the sugar–phosphate backbone cannot be bend beyond a defined angle and is commonly set to 3 nt. The fact that each base pairs with at most one other implies for any two base pairs $(i, j), (k, l) \in x$ that i, j, k and l are pairwise distinct. Note that this representation of structures as a set is succinct in that it only captures the base pairing information and not, e. g., the exact spatial positions of the molecules. In the most cases, however, it describes the function of the molecule well enough (Fontana, Konings, et al., 1993), and it significantly reduces the degrees of freedom for modelling and computing possible structures.

Even when limiting the representation of structures to sets of base pairs, the number of possible structures for a given RNA sequence grows exponentially with its length n (Stein and Waterman, 1979). Already for a moderate n, this makes many computations entirely infeasible. Further restricting the space of structures is therefore necessary. This goal can be achieved by introducing the notion of *secondary structures.* Intuitively, these are the structures which can be drawn in two dimensions without any base pairs crossing each other or the backbone. More formally, we require for any two base pairs $(i, j), (k, l) \in x$ with $i < k$ that either $i < k < l < j$ or $j < k$ holds. In the first case, (k, l) is nested into (i, j), while in the second two cases, both base pairs are non-overlapping. Secondary structures can conveniently be represented as a *dot–bracket string*, i. e., in the order of the sequence, each base pair is denoted by a matching pair of parentheses, while unpaired bases are marked with a period.

Structures not adhering to the “no crossing” constraint are said to have *tertiary interactions* or *pseudo-knots.* Tertiary structures are three-dimensional representations of the molecule, cf. Figure 4. While there are examples of biologically relevant tertiary interactions like, e. g., kissing hairpins (Chang and Tinoco, 1997), the function of RNAs can be explained within the secondary structure model in many cases because it often depends on the presence of certain structural features that are well represented by it. One important thing that should be kept in mind, however, is the fact that the physical distance between two nucleotides is not necessarily related to their “distance” in the secondary structure, which can be defined e. g., in terms of path lengths on a graph representation of the structure. Still, the secondary structure model greatly reduces the number of structures to consider while retaining many of their important features. And even though there are still exponentially many secondary structures (Clote, Kranakis, et al., 2009), the imposed constraints allows for very powerful prediction algorithms, as we shall see in the next sections.

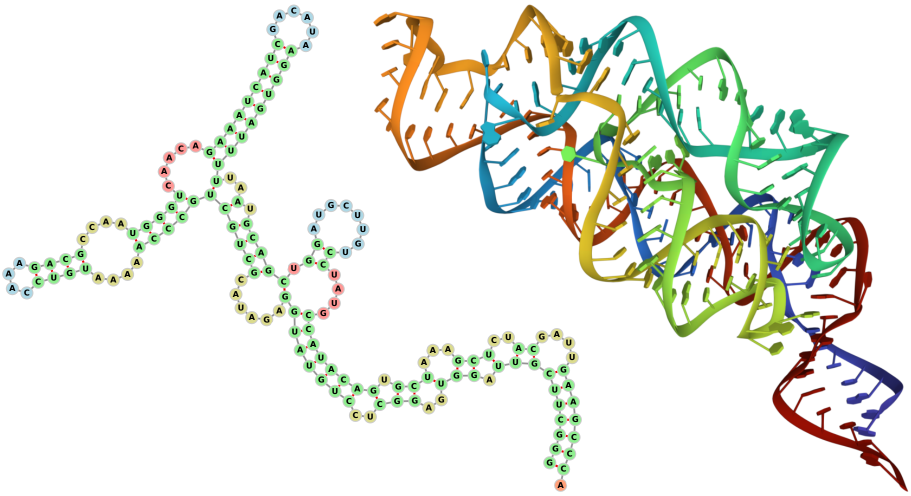


Figure 4: Secondary (*left*) and tertiary (*right)* structures of the manganese bound M-box RNA from *Bacillus subtilis.* The tertiary structure was originally obtained from an X-ray crystallography experiment (Ramesh, Wakeman, and Winkler, 2011) listed in the RCSB Protein Data Bank (Berman et al., 2000) as entry 3PDR. The secondary structure representation is part of the CompaRNA RNA structure prediction benchmark (Puton et al., 2013) and was drawn using Forna (Kerpedjiev, Hammer, and Hofacker, 2015).

2.2.2 Quantifying stability: the Gibbs free energy

The secondary structure model for RNAs can be interpreted as a partition of all possible three-dimensional structures. Each secondary structure is thus a class of this partition possessing a configurational *entropy*, which is a measure for the number of possible states of a system. Still, when assuming equilibration within these classes as well as a constant temperature, secondary structures can be considered as microscopic states. Their stability can thus be quantified in terms of their *Gibbs free energy* ΔG (Cooksy, 2014), where

$$\Delta G = \Delta H - T\Delta S.$$

Here, ΔH is the *enthalpy*, i. e., the sum of all energy stored in the bonds and interactions of the molecule, ΔS is the entropy, and T is the (absolute) temperature. ΔG is a real number, and a lower value of implies a more stable molecule. According to the SI standard, the Gibbs free energy – or free energy for short – is given in units of J (joule). However, in the context of nucleic acid biochemistry, it is common to give its values in $\mathrm{kcal\,mol^{-1}}$ (kilocalories per mole). We will adhere to this convention in this work.

An import aspect of the free energy is that it is a *relative* quantity. It is always given as a difference (thus the symbol Δ) to a defined reference value. For RNA conformations, the reference is the open chain, i. e., the molecule without any base pairs, which is defined to have a free energy of $0\,\mathrm{kcal\,mol^{-1}}$. As the open chain is usually a very unstable state for RNAs, stable conformations have a negative free energy.

From the definition of the free energy above, it follows that the stability of an RNA is temperature-dependent. Often, free energies are given for a temperature of 37 °C, which is the average temperature in the human body, but other values such as 25 °C (i. e., room temperature) are also common. Therefore, care has to be taken when comparing free energies from various sources.

Utilizing the concept of free energy, the problem of predicting the stability of an RNA structure becomes equivalent to the prediction of its free energy. Amongst others, Turner and Mathews (2010) have measured the free energy of various small oligonucleotides with a specific structure using melting experiments. Grounded on these values, an additive *nearest-neighbor* energy model can be derived, which assigns free energies to arbitrary secondary structures by adding up the individual contributions of its base pairs as well as the entropic contributions of specific substructures such as hairpin and interior loops. Such a model is implemented, e. g., in the program *RNAeval* from the Vienna RNA package (Lorenz, Bernhart, et al., 2011).

2.2.3 The minimum free energy structure

In nature, all systems strive to attain a state of minimal energy. Thus, for any RNA molecule, the structure of minimum free energy (MFE) is the most likely of all and thus of great interest. Often, the MFE structure is referred to as *the* structure of a given RNA, despite the fact that other structures may be functionally relevant as well.

Given the significance of the MFE structure, an obvious question is how to predict it for a given sequence. Unfortunately, the energy minimization problem is hard for *arbitrary* structures. While the efficient (i. e., polynomial-time) prediction of *secondary* structures as well as some simple classes of pseudo-knots is possible (Akutsu, 2000), the general problem is NP-complete (Lyngsø and Pedersen, 2000). In this work, we will therefore restrict to the prediction of secondary structures only.

The precursor of all RNA folding procedures was Nussinov's algorithm (Nussinov et al., 1978), which uses a recursive decomposition scheme to maximize the number of base pairs for a given sequence. Implemented by means of dynamic programming, it has a space and time complexity of $\mathcal{O}(n^2)$ and $\mathcal{O}(n^3)$, respectively. By extending the decomposition scheme such as to distinguish between all energetically different components appearing in RNA structures, Zuker and Stiegler (1981) were able to implement the full nearest-neighbor energy model into the recursion and thus compute the MFE structure for arbitrary sequences. The time complexity is $\mathcal{O}(n^4)$, but can be reduced to $\mathcal{O}(n^3)$ by bounding the length of interior loops. Zuker's folding algorithm is implemented in modern RNA folding software such as the *ViennaRNA* package (Lorenz, Bernhart, et al., 2011) or *mFold* (Mathews, Turner, and Zuker, 2007). *ViennaRNA* also offers a comprehensive script language interface and is used heavily throughout this work.

2.2.4 The equilibrium probabilities of structures

While the MFE structure is the most stable of all observed structures for a given sequence, it tells little about possible structural variation. There are known examples of bistable structures, which adopt different structures during their lifetime or may even remain in a non-MFE folding state, and it is common knowledge that structural features like stems undergo small structural changes over time (“helix breathing”). While these effects can only be properly described by considering the kinetics of RNA folding – a topic that will be covered in Section 2.3 –, their mere existence shows that alternative structures of RNAs play an important role and cannot be neglected. While the MFE is most stable, less stable structures can be observed as well. This leads to the question how exactly the stability of an RNA structure is related to its probability.

To answer this question, we will assume that the RNA is in its equilibrium. Theoretically, this is only the case after an infinite amount of time, but since the opening and closing of base pairs of an RNA molecule are reactions happening on a timescale of microseconds (Pörschke, Uhlenbeck, and Martin, 1973), this assumption is reasonable in practice for many cases. We consider the set X of all possible (secondary) structures of a given RNA sequence. X is called the structure *ensemble* of that sequence. As explained in Section 2.2.2, any structure $x \in X$ can be assigned a Gibbs free energy $\Delta G(x)$, which quantifies its stability. By the laws of thermodynamics, the probabilities of the structures in X follow a *Boltzmann distribution*. In this discrete probability distribution, each state (i. e., structure) is assigned a probability mass $\mathrm{Z}[x]$ depending only on its free energy:

$$\mathrm{Z}[x] = \exp\left(\frac{-\Delta G(x)}{RT}\right),$$

where T is the absolute temperature in kelvin, and $R \approx 1.987\,17\,\mathrm{cal\,K^{-1}\,mol^{-1}}$ is the universal gas constant. The exact value of R in this work has been chosen to match that used by the *ViennaRNA* package (Lorenz, Bernhart, et al., 2011). $\mathrm{Z}[x]$ is called the *Boltzmann weight* or *Boltzmann factor* of x. The term $(RT)^-1$ is also called the *inverse temperature* and mediates the temperature dependence of the probability of RNA structures. For convenience, we will also write $\mathrm{Z}[\eta]$ to denote the Boltzmann weight $\exp(-\eta/(RT))$ associated with an arbitrary free energy η. For a set of structures $Y \subseteq X$, its *partition function* is defined as

$$\mathrm{Z}[Y] = \sum_{y \in Y} \mathrm{Z}[y],$$

i. e., as the sum of the Boltzmann weights of the individual structures in Y. With $\mathrm{Z} := \mathrm{Z}[X]$, we can express the probability $\Pr[x \mid X]$ of a structure x in the ensemble X as

$$\Pr[x] := \Pr[x \mid X] = \frac{\mathrm{Z}[x]}{\mathrm{Z}},$$

which can be generalized to arbitrary sets of structures $Y_1 \subseteq Y_2$:

$$\Pr[Y_1 \mid Y_2] = \frac{\mathrm{Z}[Y_1]}{\mathrm{Z}[Y_2]}.$$

Of course, to compute these probabilities, the partition function has to be determined first. Since the number of structures in X is huge, an explicit enumeration is infeasible even for short sequences. An efficient approach relying on dynamic programming was provided by McCaskill (1990), who derived it from Zuker’s algorithm for MFE folding. To this end, the structure decomposition scheme was improved such that, for each structure, there is exactly one decomposition path. The dynamic programming matrices no longer store minimum free energies of subsequences, but partition functions of all possible substructures; and in each decomposition step, the products of the partition functions of every possible pair of subsequences are summed up. The time complexity of the approach is in $\mathcal{O}(n^3)$, just like the Zuker algorithm.

Partition functions of the ensemble X can thus be computed efficiently even for big molecules. Modern implementations like *RNAfold* from the *ViennaRNA* package also allow the computation of partition functions for certain subsets of X, characterized, e. g., by a specific substructure or the presence or absence of a particular base pair. To this end, a versatile way of specifying structural constraints is available (Lorenz, Hofacker, and P. F. Stadler, 2016), providing the analyst with a powerful and flexible tool set.

2.2.5 Numerical considerations for the computation of partition functions

Due to the exponential change in the value of the Boltzmann weight even for a moderately varying argument, the computation of partition functions can easily lead to numerical instabilities. This section will discuss implications and countermeasures to alleviate these issues.

As a first step, a scaled energy function $\Delta G^* : x \mapsto \Delta G(x) - c_{\Delta G}$ with some scaling constant $c_{\Delta G} \in \mathbb{R}$ can be used for the computation of the Boltzmann weights. Note that this will not change the probability $\Pr[x]$ of any structure $x \in X$ because

$$\begin{aligned}
\frac{\exp(-\beta(\Delta G^*(x)))}{\sum_{y\in X}\exp(-\beta(\Delta G^*(y)))} &= \frac{\exp(-\beta(\Delta G(x) - c_{\Delta G}))}{\sum_{y\in X}\exp(-\beta(\Delta G(y) - c_{\Delta G}))} \\
&= \frac{\exp(-\beta\Delta G(x))\exp(\beta c_{\Delta G})}{\sum_{y\in X}\exp(-\beta\Delta G(y))\exp(\beta c_{\Delta G})} \\
&= \frac{\exp(-\beta\Delta G(x))}{\sum_{y\in X}\exp(-\beta\Delta G(y))} \\
&= \frac{\mathrm{Z}[x]}{\mathrm{Z}} \\
&= \Pr[x]\,,
\end{aligned}$$

where β is the inverse temperature. Consequently, the free energies of a set of structures can freely be scaled by a constant value. It is often useful to rescale by the MFE, such that the MFE structure has a Boltzmann weight of 1.

Another technique comes in useful especially when *multiplying* partition functions of sets of structures Y_1, Y_2. Such operations arise frequently when the total number of states of two independent, non-interacting molecules or parts of one molecule shall be computed. Since partition functions may differ dramatically in their value, their direct multiplication $\mathrm{Z}[Y_1]\,\mathrm{Z}[Y_2]$ is prone to numerical issues. The problems get even worse for more than two partition functions. To avoid these, $\mathrm{Z}[Y_1]$ and $\mathrm{Z}[Y_2]$ are first computed as usual by summing over individual Boltzmann weights. Then, a logarithm is applied such that the product can be expressed as a sum since $\ln(ab) = \ln(a) + \ln(b)$. Finally, the exponential function is applied to the sum to invert the logarithm. Thus, by using $\mathrm{Z}[Y_1]\,\mathrm{Z}[Y_2] = \exp(\ln \mathrm{Z}[Y_1] + \ln \mathrm{Z}[Y_2])$, the error-prone multiplications are avoided.

For similar reasons, the *free energy of a set of structures* Y is sometimes used in place of the actual partition function value $\mathrm{Z}[Y]$. It is defined as $\Delta G(Y) := -RT \ln \mathrm{Z}[Y]$, i. e., the inverse of the operator Z is applied to partition function of Y. With the same argument as above, the product of the two partition functions of Y_1 and Y_2 can then be expressed as the sum of their free energies, because $\mathrm{Z}[Y_1]\,\mathrm{Z}[Y_2] = \mathrm{Z}[\Delta G(\mathrm{Z}[Y_1]\,\mathrm{Z}[Y_2])] = \mathrm{Z}[\Delta G(\mathrm{Z}[Y_1]) + \Delta G(\mathrm{Z}[Y_2])]$. The *ensemble energy*, i. e., the free energy of the ensemble X, is often used as a characterizing value, as it is more manageable than the corresponding partition function.

2.2.6 Suboptimal RNA structures

As pointed out before, the MFE structure is the most stable and most likely of all structures. However, other structures may be comparably stable, and there may even be multiple structures sharing the same MFE. In the nearest-neighbor energy model, this effect can be observed frequently because the energy parameters used to predict structures in practice are only precise to some point, after which they are usually truncated, leading to a quantized co-domain of the energy function. The parameters of Turner and Mathews (2010), for instance, are given in full decacalories. Therefore, depending on the exact sequence and its length, there are usually many structures sharing the same energy. However, this observation is *not* merely an artifact of the employed energy model, as is emphasized by the fact that there are numerous examples of multi-stable RNAs, i. e., RNAs that adapt multiple stable structures (Linnstaedt et al., 2006; Møller-Jensen, Franch, and Gerdes, 2001; Napierala and Krzyzosiak, 1997). A comprehensive structure analysis of a sequence should therefore not be limited to determining the MFE structure – or *one of* them –, but should include other stable structures as well.

This immediately leads to the question how to determine not only the *most* stable structure, but also other suboptimally stable structures – *suboptimal structures* for short. Zuker (1989) developed a concept of suboptimal structures by selecting, for each possible base pair, the most stable structure including this pair. Another approach is to enumerate all *saturated* structures, i. e., secondary structures to which no further valid base pair can be added, with a given number of base pairs (Clote, 2006). The set of all saturated structures is exactly the set of local minima in the simple Nussinov energy model, where each base pair contributes -1 to the structure's energy. A slight variation of this concept presented by Evers and Giegerich (2001) was termed *base stacking* energy model, where each *stacked* (i. e., non-isolated) base pair contributes -1 to the energy of the structure. Their contribution also includes a dynamic programming structure to enumerate the local minima with respect to this model. Yet another possibility to obtain relevant suboptimal structures is statistical or *Boltzmann sampling*. This can be achieved by randomly backtracking substructures according to their probability in the dynamic programming matrices generated by McCaskill's algorithm (McCaskill, 1990).

Most relevant to this work, Wuchty et al. (1999) developed an algorithm to efficiently enumerate *all* secondary structures of an RNA sequence with an energy not exceeding a threshold ΔG_{enum} above the MFE. The procedure employs a branch and bound approach during the backtracking phase of the MFE folding algorithm (Zuker and Stiegler, 1981). While this usually yields so many structures that the output is too verbose for manual analysis, it allows to extract only the significant part of the structure ensemble, which can then be further processed with other techniques. The parameter ΔG_{enum} is critical to balance between the number of generated structures and the fraction of the ensemble that is to be analyzed.

2.3 Structure in motion: the folding kinetics of RNA

The previous section characterized *static* properties of RNAs as well as the distribution of structures in equilibrium. Now, we will focus on the behaviour of these biomolecules *before* their equilibration. As explained in Section 2.1.4, RNAs are synthesized base by base from a DNA template by RNAP and thus do not exhibit any structure initially. While this *open chain* remains the favorable state for the first few nucleotides transcribed, the possibility to fold into much more stable structures arises quickly as transcription proceeds. The molecule is now out of equilibrium, and a dynamic refolding process begins. In this section, this process is modelled under varying conditions.

2.3.1 Refolding as a sequence of elementary transition reactions

The thermodynamic approach to RNA folding can easily be misinterpreted in the way that folding was some finite process after which the molecule attained a specific, permanent conformation with a defined probability. This is not the case. In fact, even an equilibrated RNA is permanently refolding, changing from one conformation to another. Only on average will the distribution of its structures match the computed equilibrium probabilities.

During refolding, a direct transition from one structure to another one will not happen between arbitrarily different structures. It is highly unlikely that, e. g., a sequence of three stable, adjacent hairpins immediately refolds into a multi-loop structure. Instead, microscopic rearrangements happen randomly all over the molecule and, in sum, may lead to a partial change of the global structure. The reactions considered *elementary* in this process, i. e., those consisting of “a single step”, are usually the opening and closing of a single base pair. Sometimes, base pair *shifts* are also considered elemental. A shift of a base pair (i,j) in structure x results in the new structure[1] $x \setminus (i,j) \cup (i',j)$ or $x \setminus (i,j) \cup (i,j')$ for some $i' \neq i$ or $j' \neq j$, respectively. Obviously, any structural rearrangement can be expressed as a series of elementary transitions. Also, these reactions are *reversible* by another elementary transition.

2.3.2 RNA energy landscapes

Now, we can define a basic model to capture the properties of the previously described RNA folding process in a formal representation. For such a model, one needs, in addition to the secondary structures given as the ensemble X, a notion of *neighborhood* to define which structures a given structure $x \in X$ can refold into within an elementary simulation step. Here, the set of adjacent or

---

[1] The common set operations are used to describe the addition ($\cup$) and removal ($\setminus$) of base pairs. For a clearer presentation, the braces around singleton sets are left off, e. g., $x \setminus (i,j)$ is used instead of $x \setminus \{(i,j)\}$.

neighbor structures $N(x)$ of x is defined as the set of all structures obtained by applying an elementary transition (cf. Section 2.3.1) to x. The set of allowed elementary transitions is also referred to as the *move set*. In this work, we usually consider only insertion and deletion as allowed moves, but shift moves could also be allowed. Note that, since the elementary transitions are reversible, adjacency is a symmetric property, i. e., $\forall x, y \in X : \ x \in N(y) \iff y \in N(x)$. Together with the Gibbs free energy, these ingredients define the *RNA energy landscape* $\mathcal{L} := (X, N, \Delta G)$ (Flamm, Hofacker, P. F. Stadler, et al., 2002).

As the term “energy landscape” suggests, $\mathcal{L}$ can be imagined – strongly simplified – as a natural landscape, where adjacent structures of X correspond to positions in that landscape that are close to each other, and the free energy measures the altitude of that positions. Thus, local minima correspond to valleys, maxima correspond to peaks, and transitions to adjacent structures can be interpreted as “walking” through the landscape. An example is given in Figure 6. This analogy, however, should not be over-interpreted as that may lead to wrong conclusions. One specific issue with the comparison to objects in the real world is the fact that $\mathcal{L}$ is very high-dimensional, since the number of neighbors is usually high for the most sequences. In general, it is thus not straightforward to generate a *representative*, two-dimensional plot of $\mathcal{L}$. A proper visualization requires a high degree of abstraction, for example using barrier trees as discussed in Section 2.3.4. Another unintuitive fact is that, as shown in Chapter 4, the vast majority of all possible structures is unstable, and the tiny fraction of stable, low-energy structures thus form deep holes and gorges in the landscape.

2.3.3 The transition rates of RNA folding

Chemical reactions occur at a specific rate. Especially when modelling a system of multiple reactions, their differing rates must be accounted for. This also applies to the elementary transition reactions of RNA folding. To this end, a *transition rate coefficient* $r_{x \to y} \geq 0$ is assigned to the refolding of x to y for any two structures $x, y \in X$. A high rate coefficient implies a quick refolding, and $r_{x \to y} = 0$ means that a (direct) transition is not possible at all. Since in nature, all systems strive to attain a state of low energy, the rate coefficient $r_{x \to y}$ should depend on the free energies of x and y, and $r_{x \to y}$ should be higher than $r_{y \to x}$ if $\Delta G(x) < \Delta G(y)$.

For this work, the transition rate coefficients were obtained by applying the rule of Metropolis et al. (1953):

$$r_{x \to y} := \begin{cases} \max\left\{1, \exp(-\frac{\Delta G(y) - \Delta G(x)}{RT})\right\} & \text{if } y \in N(x), \\ 0 & \text{otherwise.} \end{cases}$$

As discussed by Flamm and Hofacker (2008), this form corresponds to the Arrhenius equation $r = A\exp(-\beta \Delta G_{\mathrm{a}})$ with pre-exponential factor $A = 1$ and inverse temperature β, where the activation energy ΔG_{a} of the reaction is assumed to be 0 if $\Delta G(x) \geq \Delta G(y)$, and $\Delta G_{\mathrm{a}} = \Delta G(y) - \Delta G(x)$ otherwise. The same authors also discuss the symmetric Kawasaki rule $r' = A\exp(-\frac{1}{2}\beta[\Delta G(y) - \Delta G(x)])$ and conclude that the choice of the rate constant rule does not qualitatively change the outcome of the folding simulation.

The pair $L := (X, r_{\cdot \to \cdot})$, which implicitly determines $N(x)$ (because $y \in N(x)$ if and only if $r_{x \to y} \neq 0$), represents the *Markov process* induced by the Metropolis rule on the energy landscape $\mathcal{L}$. L can also be interpreted as a graph where each structure is a node, labelled by its energy, and a weighted, directed edge connects each pair of adjacent structures (x, y) such that the edge weight corresponds to the rate coefficient $r_{x \to y}$. The weighted graph L is connected since each structure can refold into any other one, e. g., by first opening all of its base pairs, and then closing all base pairs of the target structure, one after another. The Markov process L is thus *ergodic*, i. e., starting from any state, every other state will eventually be visited.

A property that a reaction systems composed of elementary reactions is often supposed to have is so-called *detailed balance* (Kampen, 2007). A system is said to be in detailed balance if, in its equilibrium, the rate of every elementary reaction is matched by the rate of its reverse reaction. For the folding reaction of an RNA, that means that $\forall x, y \in X : \Pr[x]\, r_{x \to y} = \Pr[y]\, r_{y \to x}$. It is easy to show that this assumption holds for the Metropolis rate constants.

Proof. Assume first that $y \notin N(x)$. Then, $x \notin N(y)$ and thus $r_{x \to y} = r_{y \to x} = 0$ and detailed balance holds. Now, assume $y \in N(x)$ and thus $x \in N(y)$. If $\Delta G(x) \leq \Delta G(y)$, then $\exp(-\beta[\Delta G(y) - \Delta G(x)]) \leq 1$ and the rate coefficients are $r_{x \to y} = \exp(-\beta[\Delta G(y) - \Delta G(x)]) = \exp(-\beta \Delta G(y)) \exp(\beta \Delta G(x))$ and, for the reverse reaction, $r_{y \to x} = 1$. It follows that

$$\begin{aligned} \Pr[x]\, r_{x \to y} &= \exp(-\beta \Delta G(x))\, \mathrm{Z}^{-1} \exp(-\beta \Delta G(y)) \exp(\beta \Delta G(x)) \\ &= \mathrm{Z}^{-1} \exp(-\beta \Delta G(y)) = \Pr[y] \cdot 1 = \Pr[y]\, r_{y \to x}, \end{aligned}$$

and thus detailed balance holds, too. The opposite case, $\Delta G(x) > \Delta G(y)$, is symmetric and the claim follows analogously. ☐

2.3.4 Coarse graining energy landscapes

Despite the reduction of structures achieved by applying Wuchty's algorithm, even for short RNAs the number of secondary structures is still far too high to use them directly as state set when performing kinetic folding simulations. One option to significantly remove the number of simulation states is to use the program *barriers* (Flamm, Hofacker, P. F. Stadler, et al., 2002), which performs a coarse graining of the structure ensemble X. It processes the *sorted* list of low-energy states $X' := \{x \in X \mid \Delta G(x) \leq \Delta G_{\text{enum}}\}$ generated by *RNAsubopt* (Lorenz, Bernhart, et al., 2011) and extracts (i) the representative set $\tilde{X}$ of local minima in X' and (ii) the energy barriers between them. Together, they comprise a tree-like structure called the barrier tree, which can readily be visualized, cf. Figure 5. *barriers* also (implicitly) assigns each $x \in X'$ to the local minimum $\Gamma(x)$ that is reached by performing a *gradient descent* in the energy landscape $\mathcal{L}$. A gradient descent (Day et al., 2016) is a path following the steepest descent of free energy. $\Gamma(x)$ can be defined recursively as

$$\Gamma(x) = \begin{cases} x & \text{if } \Delta G(x) \leq \min_{y \in N(x)} \Delta G(y), \\ \Gamma(\arg\min_{y \in N(x)} \Delta G(y)) & \text{otherwise,} \end{cases}$$

i. e., the final structure of the gradient descent is x if x is already a local minimum in $\mathcal{L}$. Otherwise, the procedure continues at the neighbor structure

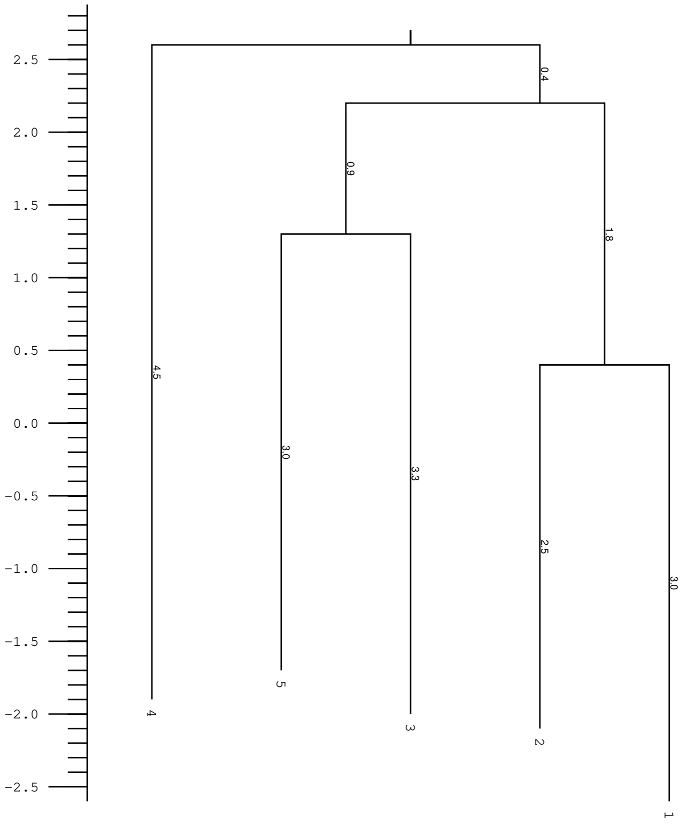


Figure 5: Visualization of the barrier tree for the five lowest-energy minima of sequence 5′-`AGCUCAAACCCUGACGUCGGCUUCCCUGCG`-3′, generated by *barriers*. The leaves of the tree represent the five local minima. The axis on the left-hand side shows the free energy of the minima and saddles in kcal mol^{-1}. The labels on the vertical edges denote the barrier height between siblings, and the length of the lines scale proportionally with it. For any two minima, their barrier heights can be determined by adding up the barrier heights on the path from the respective minimum to the least common ancestor with the other minimum.

of x that has the lowest free energy. For $\Gamma(x)$ to be defined uniquely, it is necessary that for all $x \in X$, its neighbors $N(x)$ have mutually different energies. Flamm, Hofacker, P. F. Stadler, et al. (2002) call energy landscapes with this property *locally invertible*. In practice, this formal requirement may be relaxed by employing a deterministic tie breaker rule that defines a total order on the adjacent structures, e. g., by using a lexicographic comparison on their dot–bracket representation if their energies are equal.

By assigning each $x \in X'$ to the local minimum $\Gamma(x)$, the structures are binned into well-defined *macrostates*. These are therefore simply sets of microstates, i. e., individual RNA structures. The macrostates can be interpreted as equivalence classes of the equivalence relation $x \mathcal{R}_\Gamma y :\Longleftrightarrow \Gamma(x) = \Gamma(y)$, each uniquely represented by its local minimum, which together form a partition of the ensemble. Thus, the macrostate containing the structure $x \in X$ is defined as $[x] = \{y \in X \mid \Gamma(y) = \Gamma(x)\}$, and $[x] = [y]$ if and only if $y \in [x]$ for

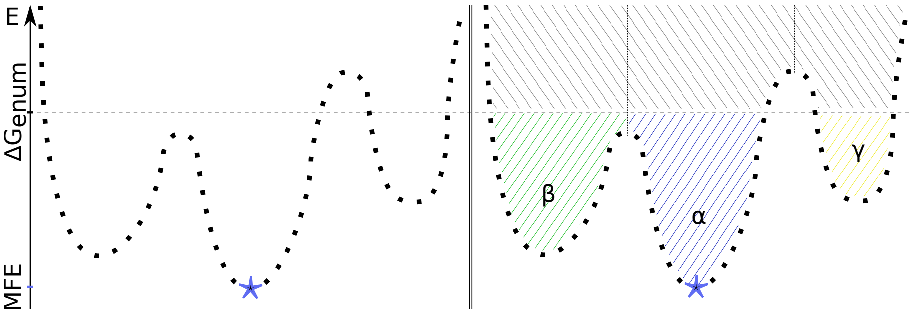


Figure 6: Schematic and strongly simplified visualization of an RNA energy landscape to which a coarse graining is applied. *left:* Individual structures are denoted as black squares, each being adjacent to its left and right neighbor. The y-axis displays their free energies. The MFE structure is marked with a blue asterisk. *right:* Coarse-grained version of the same energy landscape. The individual structures have been binned into gradient basins, denoted α, β, and γ (hatched in green, blue, and yellow, respectively). Due to the enumeration threshold ΔG_{enum} (horizontal, dashed line), the upper part of the landscape (hatched in gray) is not part of the coarse-grained representation. As a result, γ is disconnected from the remaining basins.

all $x, y \in X$. We also refer to these macrostates as (*gradient*) *basins* because of their "shape" in the energy landscape. Figure 6 shows the result of applying the described procedure to an example.

It should be noted that the definition of macrostates in terms of the gradient descent operator $\Gamma(\cdot)$ is not indisputable. As B. M. R. Stadler and P. F. Stadler (2010) point out, a gradient descent is not necessarily the most likely folding path. Instead of always following the steepest descent, they consider arbitrary *adaptive walks* – i. e., walks where the free energy of the current structure reduces with each step –, and calculate probabilities for the resulting trajectories. A structure can thus be assigned to the minimum it is most likely to refold into. This is desirable especially when considering that the Metropolis rule (cf. Section 2.3.3) assigns a rate coefficients of 1 to all transitions to adjacent structures of lower energy, regardless of how much lower their energy is.

There are also alternative approaches to coarse graining. For example, Giegerich, Voß, and Rehmsmeier (2004) describe an approach to represent secondary structures by more abstract classes referred to as RNA *shapes*. Depending on the selected granularity, this approach removes more or less details from the secondary structure, e. g., the exact number of consecutive base pairs in a stem, the size of a loop, bulges, interior loops etc., until only a coarse representation of important features remains. These shapes can conveniently be represented as a dot–bracket string just as usual secondary structures. This is a handy tool to visualize the basic structural features of a set of structures. A problem with such broad classes in the context of kinetic simulations, however, poses the required equilibrium within each macrostate discussed in the next section. While this assumption seems reasonable for gradient basins, the same shape class may contain structures separated by high energy barriers, e. g., two overlapping, mutually exclusive hairpin structures. Such a class will usually be far from equilibrium, and using such a state set may thus introduce significant errors into a simulation. Another method to reduce the number of simulation states, which is better suited for kinetic folding simulations, was proposed by Tang et al. (2008). Instead of the full ensemble, it considers only a Boltzmann sample, i. e., a random sample of structural conformations drawn with a probability proportional to their Boltzmann weight. To further reduce the number of transitions, only the k closest (with respect to the number of differing base pairs) neighbor conformation were considered. The transition rates are also computed using the Metropolis rate. Yet another approach is taken by the *basin hopping graph* framework (Kucharík et al., 2014). It also takes a Boltzmann sample from the ensemble, but applies gradient walks to the sampled structures to obtain a list of local minima. Direct paths between the minima are then computed both to discover more minima between the ones already known and to estimate the energy barriers for transitions between individual minima.

2.3.5 Transition rates for macrostates

To model RNA folding based on such a coarse-grained ensemble, it is necessary to lift the definition of transition rate constants for structures to basins of structures. For every pair of macrostates $\alpha, \beta \subseteq X$, the rate coefficient for the transition from α to β is thus computed as a weighted sum over the rate coefficients of each pair of microstates $x \in \alpha$ and $y \in \beta$:

$$\tilde{r}_{\alpha\to\beta} := \sum_{x\in\alpha}\sum_{y\in\beta} \Pr[x \mid \alpha]\ r_{x\to y},$$

where $\Pr[x \mid \alpha]$ is the probability that the Markov process L is in state x given we know that it is in the macrostate α to which x belongs. Assuming that basins are steep, the Markov process will approximately equilibrate within α before leaving the basin, justifying the approximation $\Pr[x \mid \alpha] = \mathrm{Z}[x]/\,\mathrm{Z}[\alpha]$. The approximation fails in particular for large, shallow basins, which are likely to appear in landscapes of sequences with extremely biased nucleotide distributions and unusually large fractions of unpaired bases in their ground state. The rate constants $\tilde{r}_{\alpha\to\beta}$ define a Markov process on the set of basins, which again can be seen as a graph $\tilde{L} := (\tilde{X}, \tilde{r}_{\cdot\to\cdot})$ whose neighborhoods are given by $\tilde{N}(\alpha) := \{\beta \in \tilde{X} \setminus \{\alpha\} \mid \exists y \in \beta :\ y \in N(x)\}$. Thus, two basins α, β are considered adjacent if and only if there are two adjacent structures $x \in \alpha$ and $y \in \beta$. The neighbor-generating function N thus naturally extends to gradient basins. Note that, just as in the microscopic case, the graph $\tilde{L}$ is connected when partitioning the *full* ensemble X. If, however, only a part of the ensemble is processed, e. g., when using Wuchty's algorithm to generate low-energy, paths connecting the remaining states may be removed and the graph falls apart into multiple components. In this case, the Markov process L is no longer ergodic; some basins can no longer be accessed. Since disconnected states are useless and hinder a further analysis, they should be removed before proceeding. This can easily be achieved by performing a breadth-first search on the graph $\tilde{L}$, starting at the MFE basin, i. e., the basin represented by the global MFE structure. Any basin that is not reached during the graph traversal can then be deleted.

Along with the basins and the corresponding rate matrix, *barriers* computes several statistics on the number of structures and the partition functions $\mathrm{Z}[\alpha]$ for the basins $\alpha \in \tilde{X}$.

2.3.6 Simulating first-order reaction kinetics for RNA folding

Assuming n different *species* of RNA $R_1, \dots, R_n$, representing the individual microscopic or macroscopic folding states of the molecule, the folding reaction can be expressed as a system of elementary reactions $R_i \longrightarrow R_j$ for all $i, j \in \{1, \dots, n\}, i \neq j$. Elementary reactions during which, at a time, a single molecule of a single reactant is converted to the reaction product are said to be of *first order*. The kinetics of such reactions are governed by the *law of mass action* and obey the first-order ordinary differential equation $[\dot{R_i}] = \frac{d}{dt}[R_i] = -r_{R_i \to R_j}[R_i]$, where $[R_i]$ is the concentration of species R_i, and $r_{R_i \to R_j}$ is the rate coefficient for the elementary transition from species R_i to species R_j. Thus, the change in concentration of species R_i over time due to its conversion into species R_j is indirectly proportional to its current concentration. Of course, species R_i can, in general, refold into many different states. Also, other species are refolding into species R_i. To fully describe the change in concentration for R_i, all these individual changes need to be summed over:

$$[\dot{R_i}] = \sum_{j \neq i} (r_{R_j \to R_i}[R_j] - r_{R_i \to R_j}[R_i])$$

By setting the (undefined) rate coefficient $r_{R_i \to R_i}$ to $-\sum_{j \neq i} r_{R_i \to R_j}$, the equation can be simplified to

$$[\dot{R_i}] = \sum_{j} r_{R_j \to R_i}[R_j].$$

It is commonly referred to as a *master equation* and can be interpreted as a Markov process as follows (Kampen, 2007).

The rate coefficients $r_{\cdot \to \cdot}$ are set to either microscopic rates for individual secondary structure simulation, or to macroscopic rates to simulate gradient basin transitions. Given a distribution $p(0)$ of initial states and the rate matrix $\mathbf{R} = (r_{ij})$ composed of the respective rate coefficients $r_{x_j \to x_i}$ or $r_{\alpha_j \to \alpha_i}$, the distribution after some time τ is simply given by $p(\tau) = \exp(\tau \mathbf{R})$. It can be computed using standard methods of linear algebra, requiring in essence the diagonalization of $\mathbf{R}$. The software *Treekin* (Wolfinger et al., 2004) is the implementation used for this work. It can easily process rate matrices of a dimension up to a few thousands. The resulting output gives, for each time step, the population density of each state. The user can choose the length of the simulated time period, which corresponds to setting the transcription rate.

Treekin expects a connected state space $(X, r_{\cdot \to \cdot})$. As noted above, the full ensemble of secondary structures is always connected, but this is not necessarily the case after the truncation of the landscape $(X, N, \Delta G)$ to low-energy structures. The author's software package *BarMap-QA* (Section 3.4) contains the script `barriers_keep_connected` that truncates any disconnected state from the rate matrix and generates *Treekin*-compatible input. As an alternative, heuristic methods such as `findpath` from the *ViennaRNA* package (Lorenz, Bernhart, et al., 2011) or *RNAEAPath* (Li and Zhang, 2012) could be used to estimate energy barriers and, thus, approximate transition rates between the components of a disconnected landscape. However, such heuristics introduce errors into the simulation, and increasing ΔG_{enum} (if computationally feasible) is thus a safer method to include relevant, disconnected states.

2.3.7 A model for cotranscriptional folding

A simulation as performed by *Treekin* is well capable to capture the dynamics of RNA folding. However, the rate matrix characterizing the transition rates and number of states – and thus many other parameters the simulation depends on – are assumed to be fixed. These parameters include the temperature, ion concentrations, and the sequence of the RNA molecule itself. Additionally, the binding of other molecules may change the free energy of certain structures significantly. One possibility to include this variation into a kinetic simulation was introduced by Hofacker, Flamm, et al. (2010) in their *BarMap* framework. The underlying idea is to overcome the limitation of having to use a single, fixed energy landscape for the entire *Treekin* simulation by allowing a *sequence* of coarse grained energy landscapes $\tilde{L}_1, \dots, \tilde{L}_n$ and their associated transition rate matrices as input instead, each of which may be using a different set of parameters and states. To this end, a set of maps $\mu_1, \dots, \mu_{n-1}$ is constructed such that μ_i assigns each state of landscape $\tilde{L}_i$ to a state of landscape $\tilde{L}_{i+1}$. The details of this construction process are described below. For now, we assume that the maps and landscapes are readily available. Figure 7 shows a simple example of a single mapping step between two energy landscapes.

Given these ingredients, the simulation starts with in $\tilde{L}_1$ with user-defined populations p_0 and runs until reaching a specific end time. Then, μ_1 is employed to map each state's population to the corresponding state in $\tilde{L}_2$. If multiple states are mapped to the same state in the successor landscape, their population is summed up. In general, the initial populations of landscape $i = 1$ are

$$\mathbf{x}_j^{(1)}(t_1^0) = p_0$$

and, for $i = 2, \dots, n$,

$$\mathbf{x}_j^{(i)}(t_i^0) = \sum_{k \in \mu_{i-1}^{-1}(j)} \mathbf{x}_k^{(i-1)}(t_{i-1}^{\infty}),$$

where $\mathbf{x}_j^{(i)}(t)$ is the population of the j-th state of landscape i at time t, t_i^0 and t_i^∞ are the simulation start and end times for landscape i, respectively, and $\mu_{i-1}^{-1}(j)$ is the preimage of state j from $\tilde{L}_i$ in $\tilde{L}_{i-1}$, i. e., the set of all states in $\tilde{L}_{i-1}$ that are mapped to state j. For $t \in (t_i^0, t_i^\infty]$, the populations $\mathbf{x}_j^{(i)}(t)$ are determined by running *Treekin* on the rate matrix of $\tilde{L}_i$ as usual. This process continues in the following landscapes until reaching $\tilde{L}_n$, where the simulation terminates at time t_n^∞.

For simulations of cotranscriptional folding, the sequence of landscapes is naturally generated from all possible prefixes of the input sequence such that L_i models folding reaction of the first i nucleotides. Of course, elongating the sequence by more than one nucleotide per landscape is also possible, but at the cost of accuracy of the simulation.

To construct the map $\mu_i : \tilde{X}_i \longrightarrow \tilde{X}_{i+1}$, where $\tilde{X}_i$ is the set of macrostates of $\tilde{L}_i$, *BarMap* considers the representative minimum x of each basin $[x] \in \tilde{X}_i$. It then appends an unpaired nucleotide to the end of x to obtain $x' \in X_{i+1}$ and sets $\mu_i([x]) := [x'] \in \tilde{X}_{i+1}$, i. e., basin $[x]$ is mapped to the basin represented by the local minimum $y = \Gamma(x')$ in the next energy landscape.

In practice, the structure y does not necessarily represent a basin in $\tilde{X}_{i+1}$, e. g., because the low-energy part $X'_{i+1} \subseteq X_{i+1}$ was not enumerated up to

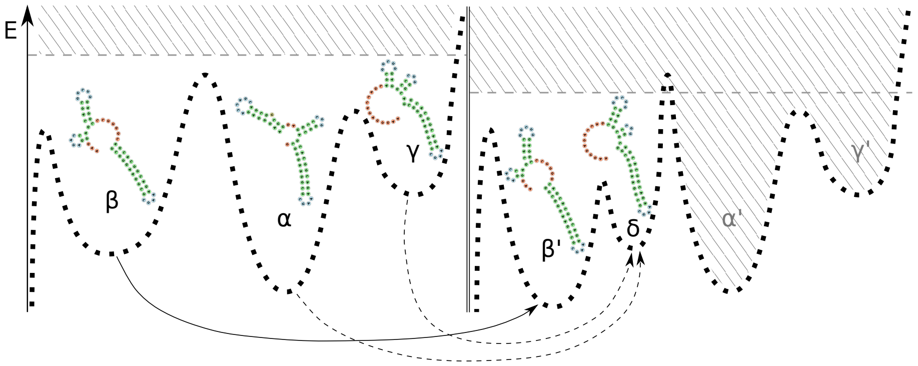


Figure 7: Mapping of some exemplary basins for two coarse-grained, partially enumerated RNA energy landscapes. The left landscape is mapped to the right one. The dotted lines, again, indicate the RNA structures of the individual landscape, and each Greek letter denotes a basin. The structure of each basins' minimum is plotted above. Solid arrows mark exact mappings while dashed arrows are approximate ones. Basin β from the left-hand landscape is mapped to the equivalent basin β' in the right-hand landscape, which has a lower local minimum due to the extension of the $3'$ hairpin by one base pair. Since the enumeration threshold (grey, horizontal, dashed line) of the right landscape is lower, the saddle between α' and β' is not enumerated. Thus, α' and γ', the equivalent basins of α and γ, are now disconnected and removed from the landscape. Instead, α and γ are mapped to a new basin δ due to their low base pair distance to the local minimum of γ.

sufficiently high energies to recover all previously connected minima, cf. Figure 7. *barriers* may also apply heuristics to remove small shallow minima. In both cases, $y \notin X'_{i+1}$ may be the consequence. To handle these degenerate conditions, *BarMap* uses an approximate approach and maps $[x]$ to another $[z] \in \tilde{X}_{i+1}$ such that the *distance* of minima x and z is minimal among all macrostates in $\tilde{X}_{i+1}$. To this end, the *base pair distance* $d_{\text{bp}}(x, z) = |(x \setminus z) \cup (z \setminus x)|$ is used, i. e., the number of base pairs present in one structure, but not in the other. If multiple structures with minimal distance exist, a lexicographic comparison is used as a tie breaker. While this rule ensures that μ_i is defined on every state of $\tilde{L}_i$, there is no guarantee that the mapping is an adequate choice. For example, a minimum containing two adjacent hairpins may be mapped to a structure containing a single, central hairpin if the former structure is not enumerated in the successor landscape. As a basic quality indicator, *BarMap* therefore characterizes its mappings either as *exact* or *approximately*, depending on whether the base pair distance between the minima of the mapped and the target basin is zero or greater. In the following sections, we will reconsider this definition and relax it to make it more applicable in practice. Also, the problem to quantify the quality of the generated landscapes, maps, and simulations will be treated much more thoroughly.

Note that in the *BarMap* framework the choice of the landscapes $\tilde{L}_0, \dots, \tilde{L}_n$ is, in principal, arbitrary as long as suitable mappings $\mu_1, \dots, \mu_{n-1}$ are provided. Apart from cotranscriptional folding, the approach may as well be used other scenarios like, e. g., varying temperature, transcription rates, structural constraints; or a mixture of any of these.

2.3.8 Stochastic simulations of folding kinetics

For the sake of completeness, we will briefly discuss an alternative class of approaches to the simulation of RNA folding kinetics based on stochastic sampling. They can be based on the same notions and models as the previously described, global approach. In contrast to them, however, they compute individual *trajectories* through the space of RNA structures, i. e., a finite sequence of structures $x_0, \dots, x_n$, such that x_i is a neighbor of x_{i-1} for all $i = 1, \dots, n$. Often, moves much more complex than the elementary opening and closing of single base pairs are used, e. g., insertion or removal of entire helices, allowing rapid simulations even for bigger molecules at the cost of precision. The initial structure x_0 can be specified by the user or chosen randomly. The next structure in the sequence is determined by randomly choosing one of all possible neighbors. To obtain a realistic prediction and maintain the property of detailed balance, the sampling procedure needs to correctly consider the probabilities to transition to any possible neighbor and accordingly select the next structure. This can be achieved by utilizing the Gillespie algorithm (Gillespie, 1977). In each step and for n possible transitions with rate coefficients $r_1, \dots, r_n$, the algorithm randomly chooses one transition to perform, and the probability of choosing the i-th transition is $r_i / \sum_j r_j$ (i. e.,, it is proportional to the respective transition rate). Note that this algorithm also keeps track of the (continuous) time τ that passes with each transition. Specifically, at each transition, τ is increased by $-(\sum_j r_j)^{-1} \log(\varrho)$, where ϱ is a random number sampled uniformly from the interval $[0, 1]$. Finally, the computation of the trajectory is terminated according to one of many possible criteria, e. g., a given simulation time, a given number of steps, the first (or k-th) passage of a specified target structure etc. While a single trajectory may not be very informative, a bigger number of them will likely draw an accurate picture of the folding process.

There are numerous examples of algorithms and programs to compute folding trajectories. Examples include *kinfold* (Flamm, Fontana, et al., 2000), *kinefold* (Xayaphoummine, Bucher, and Isambert, 2005), *kinwalker* (Geis et al., 2008) and others. They differ in the type of transitions they allow (e. g., insertion or deletion of single base pairs in *kinfold*, or addition of entire helices in a single step in *kinwalker*), the type of permitted structures (e. g., *kinefold* also allows structures containing pseudo-knots), and the employed sampling procedure (e. g., rejection or Gillespie sampling).

A general advantage of these methods is that, in comparison to global simulations like *Treekin*, they simulate more explicitly the behaviour of an RNA as it happens in the living cell. The computed trajectory directly corresponds to an actual folding pathway of the molecule. This way, funnels and kinetic traps in the energy landscape can be detected efficiently. They are also trivial to parallelize as the individual, computed trajectories are independent of each other. The performance for each individual trajectory, however, strongly depends on the transitions allowed and the length of the sequence. Changing individual base pairs may be the most precise transition, but quickly becomes infeasible as the sequence length increases. More coarse transitions like helix insertions, on the other hand, model the folding process only approximately and thus less precisely. It is, in theory, also easy to implement structural constraints on the intermediate structures.

On the downside, one needs a huge amount of trajectories to make valid claims about the behaviour of the input RNA molecule. Additionally, one needs sophisticated methods to generate interpretable results from a set of computed trajectories. Another problem is that a sensible stop criterion needs to be defined, and the trajectories generated may strongly depend on that choice.

All in all, trajectory-based methods are an interesting alternative to other folding models, but require the choice of some delicate, additional parameters and an increased amount of post-processing to obtain interpretable results.

2.4 The RNA design problem

The previous sections described various models for the folding process of a given RNA sequence. The motivation was the observed emergence of biological function from the structure of these molecules. Being able to predict their structure thus often allows to answer the question for their biological function and is therefore a valuable tool for the analysis of novel genes. One can, however, also reverse this question and ask how a sequence needs to be composed if it is supposed to preferably fold into a given structure (and thus perform a specific function). This problem is referred to as *inverse folding problem* for RNA. While obtaining an optimal solution for it is **NP**-hard (Schnall-Levin, Chindelevitch, and Berger, 2008), practice has shown that, for the most instances, sufficiently good approximations can easily be found, e. g., by using a program like *RNAinverse* (Hofacker, Fontana, et al., 1994) that starts at a random sequence and changes it by applying single-nucleotide mutations until a good result is achieved. The reason that such simple procedures can effectively solve this complex problem is elaborated in a study of (Schuster et al., 1994). Its key observations include that, at a given sequence length, the number of possible sequences is much higher than the number of possible secondary structures, and that huge *neutral networks* are embedded into the sequence space, i. e., networks of adjacent structures that fold into the same structure. This also means that sequences with specific structures can easily arise as a result of evolutionary processes if the resulting biological function is advantageous to the organism.

Inverse folding can be generalized to the task of finding a sequence that exhibits arbitrary properties, a highly interesting problem known as *RNA design*. It is long known that bistable sequences, i. e., sequences with two stable structural conformations, can easily be found (Flamm, Hofacker, Maurer-Stroh, et al., 2001). While there are always sequences that can fold into any two given structures, this is not necessarily the case for three or more structures. In the same work, these authors present a result that precisely characterizes the cases in which compatible sequences exist for a given set of target structures. To this end, a single graph G representing all target structures is constructed, and a compatible sequences exists if and only if G is bipartite. A uniform sampling procedure for the space of compatible sequences was implemented in the software *RNAdesign* Höner zu Siederdissen et al. (2013) and was improved especially for cases with many complex constraints in the library *RNAblueprint* (Hammer, Tschiatschek, et al., 2017). Still, the computational complexity of these methods grows exponentially with the sequence lengths. Later, Hammer, Ponty, et al. (2018) have shown that this complexity is inevitable when counting all sequences compatible with the given constraints, which is a requirement for uniform sampling. Instead, they followed a different strategy and implemented a Boltzmann-weighted sampling for generalized energy models in the *RNARedPrint* framework, which permits efficient sampling from sequence populations with specific free energies or GC contents.

For specific applications, an explicit construction of suitable candidate sequences according to a set of rules and patterns may be an alternative strategy. For example, to design cognate transcriptional riboswitches from a given *aptamer* sequence known to specifically bind the ligand theophylline, Wachsmuth, Findeiß, et al. (2013) devised a design approach that used such patterns to ensure the resulting sequences exhibit the required biological features. As the capability of an aptamer to bind its ligand tends to be sensitive to mutations, changing its sequence was not desirable. Furthermore, a transcriptional riboswitch requires a small, stable hairpin followed by a poly-U stretch to be able to trigger transcription termination, and this hairpin must overlap with the aptamer to make the terminator formation dependent on the binding of the ligand. These constraints left so few degrees of freedom that an explicit construction of the riboswitch candidates with only a few random spacer and loop sequences became possible. Some of the candidates were then successfully tested a laboratory experiment.

To conclude, RNA design can be a challenging problem in the presence of many complex constraints, but can also be solved efficiently for many cases that often arise in practice. A set of feature-rich, flexible and performant software tools is available to tackle these problems, and allow for exciting applications in the field of synthetic biology.

CHAPTER 3

Assessing the Quality of RNA Folding Models

Contents

3.1 Coverage: representative subsets of structures **30**
3.2 The coverage of low-energy bands **31**
  3.2.1 Methods 31
  3.2.2 Results 31
  3.2.3 Discussion 32
3.3 Canonical energy landscapes **32**
  3.3.1 Background 33
  3.3.2 Methods 33
  3.3.3 Results 34
  3.3.4 Discussion 34
3.4 *BarMap-QA*: cotranscriptional folding with quality assurance **35**
  3.4.1 Background 36
  3.4.2 Theory 38
  3.4.3 Implementation 40
  3.4.4 Application 46
  3.4.5 Discussion 51
3.5 Conclusion **53**

In many cases, the structure of a transcript and its dynamic rearrangements are crucial properties that mediate the function of the molecule in the cell. In Chapter 2, many powerful methods to simulate RNA folding using both thermodynamic and kinetic approaches have been introduced. It has also been mentioned in various situations that exact computations on the full ensemble are usually infeasible especially for kinetic folding simulations, and that various heuristics have to be employed to reduce the number of individual simulation states. The effect of these simplifications is analyzed and discussed in this chapter, which focuses on the evaluation of quality measures for given folding models. The author's software *BarMap-QA*, a comprehensive package providing quality measures for the simulation of cotranscriptional folding, will be presented.

3.1 Coverage: representative subsets of structures

The number of possible secondary structures grows exponentially with the length of an RNA (Stein and Waterman, 1979). As a consequence, even moderately sized RNA molecules have so many possible structures that it becomes infeasible to construct them all, let alone performing costly computations on them. The probability of a structure in equilibrium, however, decreases exponentially with its Gibbs free energy. Thus, while a sequence's ensemble is usually huge, most of the structures are so unstable they are biologically irrelevant at least in equilibrium. In practice, it is therefore both necessary and reasonable to only consider structures that are "stable enough" in folding simulations or thermodynamic analyses of the ensemble. While it easy to enumerate suboptimal structures (cf. Section 2.2.6), it is not immediately clear how many structures are required to adequately describe the full ensemble. To this end, we introduce the notion of ensemble coverage.

Let X be the structure ensemble of a fixed RNA sequence, and $Y \subseteq X$ an arbitrary subset of structures. As explained in Section 2.2.4, the probability that an equilibrated RNA is folded into any structure $y \in Y$ is determined by the fraction of their respective partition functions:

$$\Pr[Y] = \frac{\mathrm{Z}[Y]}{\mathrm{Z}},$$

where $\mathrm{Z} = \mathrm{Z}[X]$. We also call this probability the *coverage* of Y with respect to X, because it measures the fraction of the probability mass of X that is preserved when only considering Y instead of the full ensemble. Note also that an efficient computation of this probability is often possible in practice, since Z can be determined in $\mathcal{O}(n^3)$ using partition function folding (McCaskill, 1990). The much smaller set Y can often be explicitly constructed, e. g., by using Wuchty's algorithm (Wuchty et al., 1999), such that the Boltzmann weights of the individual structures can be summed over. Alternatively, Y could also be modelled using structural constraints, and the partition function could then be computed using dynamic programming as implemented in the *ViennaRNA* package (Lorenz, Bernhart, et al., 2011). Scaling the individual energies using the MFE as described in Section 2.2.5 helps to avoid numerical instabilities.

While computing probabilities of subsets of the ensemble is not very exciting by itself, the application as a measure for completeness of this subset is very useful and available in many situations. The concept of coverage will be used throughout this section.

3.2 The coverage of low-energy bands

One of the simplest possible applications of the concept of coverage is to measure the completeness of a low-energy band of structures as generated by Wuchty's algorithm. This enumeration method has an enumeration threshold ΔG_{enum} as single main parameter, which determines up to which energy value all structures should be generated. Note that in the actual implementation, *RNAsubopt* from the *ViennaRNA* package, this parameter is given as an *energy range* instead, i. e., as the difference to the global MFE. Here, however, we will assume ΔG_{enum} to be the difference to the open chain.

3.2.1 Methods

We denote by $X_{\leq \Delta G_{\text{enum}}} = \{x \in X \mid \Delta G(x) \leq \Delta G_{\text{enum}}\}$ the subset of all structures of the ensemble X with a free energy at most ΔG_{enum}. *RNAsubopt* emits all structures $x \in X_{\leq \Delta G_{\text{enum}}}$ in dot–bracket notation as well as their respective free energies in the specified energy range. The order of the structures is determined by the scheme used to backtrack the dynamic programming matrices, and thus in general not sorted. Since we only aim to sum up the Boltzmann weights $\mathrm{Z}[x]$, this is not a concern. That is advantageous compared to other methods like *barriers*, which require a prior sorting of the structure list. By summing up the $\mathrm{Z}[x]$, we obtain $\mathrm{Z}[X_{\leq \Delta G_{\text{enum}}}]$. Since the full partition function Z can be efficiently computed using dynamic programming, e. g., using the *RNAlib* library of the *ViennaRNA* package, the coverage $\Pr[X_{\leq \Delta G_{\text{enum}}}]$ can readily be computed. A technical peculiarity that had to be considered was the fact that *ViennaRNA* internally applies a smoothing of the energy function during the partition function computation. This makes the partition function differentiable with respect to the parameters of the used energy model and thus allows, e. g., to incorporate experimental data into the computational structure prediction (Washietl, Hofacker, P. F. Stadler, and Kellis, 2012). This smoothing, however, introduces a small bias into the computation of Z that produces spurious results when used together with summed, non-smoothed Boltzmann weights. Thus, a new option `pf_smooth` was added to the `md` (model detail) class of the library interface, which allows the user to disable smoothing if required. This contribution of the author is now part of the official release and thus also available to other users. The computed partition function value can then be used to precisely determine the probability $\Pr[X_{\leq \Delta G_{\text{enum}}}]$.

3.2.2 Results

The coverage achieved by enumerating random sequences of varying length is shown in Figure 8. While an energy range of $10\,\mathrm{kcal\,mol^{-1}}$ is sufficient to achieve high coverages of 96–100% for sequences of lengths up to 160 nt, the coverage rapidly drops for longer sequences. For sequences up to length 80 nt, comparable coverage values can already be achieved by only enumerating up to $8\,\mathrm{kcal\,mol^{-1}}$. While enumerating $10\,\mathrm{kcal\,mol^{-1}}$ is a matter of seconds for sequences up to 100 nt, it may become computationally challenging for sequences

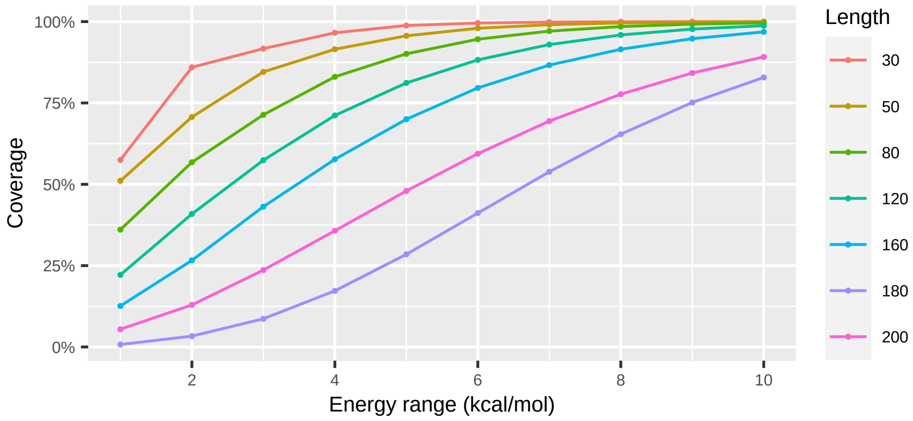


Figure 8: Coverage of low-energy bands for varying sequence lengths. The energy range is the difference between the enumeration threshold ΔG_{enum} and the sequences' MFE. The coverage achieved with a given energy range generally reduces with growing sequence length. For the sequence of 180 nt, however, the coverage is worse than for the longest sequence, which is 200 nt long.

as long as 200 nt, depending on the specific sequence. The computation for the chosen sequence of length 200 took about 11 minutes but, contrary to the expectation, 7 hours where required to enumerate the 180 nt long sequence.

3.2.3 Discussion

An enumeration of 10 $\mathrm{kcal\,mol^{-1}}$ usually achieves high coverages for sequences up to 160 nt and is recommended. For shorter sequences, a smaller energy range may be sufficient, but since enumeration usually fast anyway in these cases, a range of 10 $\mathrm{kcal\,mol^{-1}}$ may serve as a good default value.

The required computation times impressively demonstrate that the number of structures may vary dramatically even for sequences of the same length, and that specific sequences may have a much lower coverage than expected. In the challenging cases, increasing the enumeration threshold to compensate for a low coverage is thus not always an option. The application of Wuchty's algorithm to extract a significant part of the ensemble is therefore limited to RNAs of a length of approximately 160 nt. Since the following processing steps in a kinetic folding simulations are usually much slower than the generation of the secondary structures, a quick analysis of the coverage of the selected energy band gives useful information concerning a good choice of ΔG_{enum} at almost no additional cost and should therefore be a standard procedure.

3.3 Canonical energy landscapes

This section deals with the concept of canonical structures, a heuristic to significantly reduce the number of structures in the ensemble of an RNA by neglecting supposedly unstable conformations. As the structure ensemble is huge even for short sequences, this method is of great interest when performing kinetic folding simulations. The effect of the restriction to canonical structures on selected sequences is demonstrated, and a score to estimate its impact on folding simulations is discussed.

This section is based on the following literature:

S. Findeiß, S. Hammer, M. T. Wolfinger, F. Kühnl, C. Flamm, and I. L. Hofacker (2018). "In silico design of ligand triggered RNA switches". In: *Methods* 143. Methods and advances in RNA characterization and design, pp. 90–101. DOI: `10.1016/j.ymeth.2018.04.003`.

It will not be cited individually in the text.

3.3.1 Background

The great number of secondary structures renders their complete analysis infeasible even for small sequences. Coarse graining techniques, cf. Section 2.3.4, significantly reduce the number of states, but usually require the computation of the microstates (i. e., the individual structures) at first. Wuchty's algorithm, cf. Section 2.2.6, still emits very many structures depending on the choice of ΔG_{enum}. Additional methods to reduce the number of structures are therefore very useful and allow faster analyses, or the analysis of bigger molecules using the same computational resources.

The `--noLP` option of *RNAsubopt* achieves a considerable speed-up by neglecting structures containing so-called *isolated* or *lonely* base pairs, i. e., base pairs which are not directly surrounded by – or surround themselves – another base pair (Bompfünewerer et al., 2007). Formally, for some structure x, $(i, j) \in x$ is a lonely pair if and only if $\{(i-1, j+1), (i+1, j-1)\} \cap x = \emptyset$. Put differently, this option enforces a minimal helix length of two base pairs. The biological motivation of this optimization is that lonely base pairs usually destabilize a secondary structure and thus would open up again quickly. Structures not containing any lonely pairs are called *canonical* structures.

Considering only canonical structures significantly reduces the resources required for conducting the analysis, but may also bias its results. Therefore, when analyzing a new sequence, the question arises whether applying this heuristics will, in this specific case, yield accurate results or not. Here, we derive a measure that helps to answer this question on a per-sequence basis. More statistical analysis on the subject of canonical structures is also presented in Section 4.2.

3.3.2 Methods

To compute the coverage of canonical structures for a given sequence with ensemble X, we first calculate the partition function Z of the full ensemble using partition function folding (McCaskill, 1990). Like in the previous section, the set of all structures $X_{\leq \Delta G_{\text{enum}}}$ with energy at most ΔG_{enum} can be enumerated efficiently and, along with it, also the associated partition function $\mathrm{Z}_{\leq \Delta G_{\text{enum}}}$. For each enumerated structure, one can easily check whether it is canonical and, if so, sum over their Boltzmann weights to obtain $\mathrm{Z}^{\text{can}}_{\leq \Delta G_{\text{enum}}}$. As ΔG_{enum} is increased, $\mathrm{Z}_{\leq \Delta G_{\text{enum}}}$ approaches Z and the coverage of $X_{\leq \Delta G_{\text{enum}}}$ approaches 1. Thus, it becomes possible to estimate the coverage of *all* canonical structures

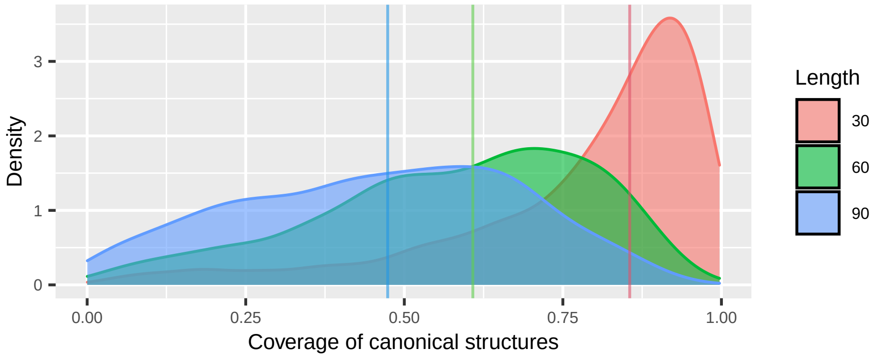


Figure 9: Coverage of canonical structures within the full ensemble for random sequences of different length. The median of each group is denoted by a vertical line. The coverage decreases significantly as the sequence length increases.

X_{can} in the ensemble by the coverage of the *enumerated*, low-energy canonical structures, i. e., $\Pr[X_{\text{can}}] \approx \Pr[X^{\text{can}}_{\leq \Delta G_{\text{enum}}}]$.

To analyze the distribution of canonical coverage, random RNA sequences of lengths 30, 60, and 90 have been generated, 1000 of each length. For each of the sequences, its partition function Z as well as all structures $X_{\leq \Delta G_{\text{enum}}}$ within an energy band of $\Delta G_{\text{enum}} = 10\,\text{kcal}\,\text{mol}^{-1}$ have been enumerated using *RNAsubopt*. All sequences for which the ensemble coverage $\mathrm{Z}[X_{\leq \Delta G_{\text{enum}}}] / \mathrm{Z}$ was smaller than 99% have been excluded from the analysis. For the remaining sequences, $X_{\leq \Delta G_{\text{enum}}}$ can be considered as representative for the entire ensemble, covering at least 99% of its probability mass.

3.3.3 Results

Density plots of the coverage of canonical structures, grouped by sequence length, can be seen in Figure 9. Two obvious trends can be observed when increasing the sequence length: firstly, the coverage of canonical structures decreases, and secondly, the variance increases. While for sequences of length 30, the canonical ensemble of every other random sequence has more than 85% coverage, this number drops to only 47% for sequences of length 90. For this length, the observed coverage values range from 0.02% to 95%. Even for shorter sequences, the range is so big that one cannot sensibly make predictions by only looking at the length.

3.3.4 Discussion

The coverage of the canonical ensemble varies greatly with the sequence length, but also for multiple sequences of the same length. The longer the sequence, the more uniform is the distribution of the coverage. This implies that this parameter should be calculated for any sequence for which a – kinetic or thermodynamic – folding analysis limited to canonical structures is to be performed. Though a considerable speedup may be achieved by limiting the analysis to canonical structures, this should only be done if the excluded fraction of probability mass is not too big, or a significant error may be introduced.

As a technical side node, it is arguable that, instead of the lengthy enumeration process described, the ensemble energy of the canonical ensemble could be directly computed using partition function folding via `RNAfold -p --noLP` from the *ViennaRNA* package. However, due to current technical limitations, the returned (canonical) ensemble energy is only an upper bound of the actual value and may dramatically over-predict the fraction of canonical structures. This was also confirmed by personal communication with the leading developer of the *ViennaRNA* package. As an example, we consider a 30 nt sequence[1], which has a full ensemble energy of $\Delta G(\mathrm{Z}) = -8.62\,\mathrm{kcal\,mol^{-1}}$ and a canonical ensemble energy of $\Delta G(\mathrm{Z_{can}}) = -8.37\,\mathrm{kcal\,mol^{-1}}$, and so $\Pr[X_{\mathrm{can}}] = \mathrm{Z}[-8.37 + 8.62] \approx 67\%$. *RNAfold*, on the other hand, reports a significantly higher canonical ensemble energy of $\widehat{\mathrm{Z}}_{\mathrm{can}} = -8.60\,\mathrm{kcal\,mol^{-1}}$. Consequently, $\Pr[\widehat{X}_{\mathrm{can}}] \approx 96\%$, i. e., it erroneously predicts the fraction of canonical structures to be 29% higher than the explicit summation method. While the difference in the prediction is usually much smaller than in this rather extreme case, this still demonstrates that reliable results cannot be achieved using the current implementation of partition function folding in *ViennaRNA* for this specific application.

Due to the enormous number of structures even for small sequences, this fact may seem like a major obstacle. In practice, however, despite the required explicit construction of low-energy structures, the required computation time is usually small compared to any following kinetic analysis. For thermodynamic analyses, it is sufficient to enumerate the energy range that contributes significantly to the ensemble of the sequence. Additionally, when enumerating only a part of the ensemble, the coverage of this subset of structures may be more relevant than the coverage of *all* canonical structures.

The described measure provides effective means to avoid situations in which highly probable structures are accidentally excluded from further analyses when restricting the ensemble to canonical structures. The proposed check is easy to perform and the required computational effort is modest. The method is thus a practical addition to the bioinformatician's tool set and should be a standard procedure whenever the exclusion of non-canonical structures is considered.

3.4 *BarMap-QA*: cotranscriptional folding with quality assurance

Structural changes in RNAs are an important contributor to controlling gene expression not only at the post-transcriptional stage but also during transcription. A subclass of riboswitches and RNA thermometers located in the $5'$ region of the primary transcript regulates the downstream functional unit – usually an ORF – through premature termination of transcription. Such elements not only occur naturally but they are also attractive devices in synthetic biology. The possibility to design such riboswitches or RNA thermometers is thus of considerable practical interest. Since these functional RNA elements act already during transcription, it is important to model and understand the dynamics of folding and, in particular, the formation of intermediate structures concurrently with transcription. Cotranscriptional folding simulations are therefore an important

---

[1] `GGCCCUACGCCACGCAAUAGUUGAGGCGUG`

step to verify the functionality of design constructs before conducting expensive and labour-intensive wet lab experiments.

Section 2.3.7 describes all necessary components to model the dynamic, possibly cotranscriptional, folding process for small to medium-sized RNA molecules up to about 100 nucleotides using the *BarMap* framework. The quality of the obtained results, however, strongly depends on the threshold ΔG_{enum} used to enumerate the low-energy structures of each landscape. While it is fairly obvious that not considering relevant states and transitions will lead to imprecise or even completely wrong predictions, it is non-trivial to determine which values of ΔG_{enum} yield reasonably accurate results without raising the computational cost of the simulation to an unacceptable level. Using *BarMap*, ΔG_{enum} can be chosen for each landscape individually. As such a simulation run may easily involve more than fifty landscapes, it becomes obvious that a systematic approach of choosing ΔG_{enum} is necessary. The key to solving this problem is to efficiently measure how "complete" a partially enumerated landscape actually is and how good important states are mapped into the next landscape. This has been achieved by developing the software package *BarMap-QA*, which wraps the original *BarMap* scripts to aid the user in their application. To this end, we applied the idea of ensemble coverage to devise multiple quality metrics for the conducted simulation, and generalize the definition of exact and approximate mappings. Building on these concepts, *BarMap-QA* automates many steps of the folding analysis and constantly presents the computed quality scores to the user, who can then precisely adjust the simulation parameters to obtain the best possible results at minimal computational cost. It also provides powerful tools for post-processing and analyzing the generated output, including the generation of clean plots visualizing the entire simulation run. *BarMap-QA* is free and open source software, and provided to the user as a highly portable, ready-to-run Docker container hosted at Docker Hub, which can be installed using a single command.

This section is based on the following literature:

F. Kühnl, P. F. Stadler, and S. Findeiß (2019). "Assessing the Quality of Cotranscriptional Folding Simulations". In: *RNA Design*. Ed. by R. Lorenz. Methods in Molecular Biology. Manuscript accepted for publication. Berlin: Springer Nature.

It will not be cited individually in the text.

3.4.1 Background

RNA folding is an inherently dynamic process and the details of the folding trajectory are at least occasionally biologically relevant. Metastable states, for example, are sometimes the functional ones, as in the example of the Hok/Sok host killing system (Gerdes and Wagner, 2007). In living cells, a nascent RNA starts folding into stable structures while it is transcribed by an RNA polymerase. The structures formed in cotranscriptional folding have decisive functions, e. g., in the case of transcriptional riboswitches, where they decide whether the transcription is terminated or continued to produce a full-length RNA transcript. The structures formed cotranscriptionally are transient and will continue to refold as the transcript is elongated. The mechanisms underlying the biological function of such RNA elements thus can only be understood in terms of dynamics of the folding process.

Riboswitches and RNA thermometers that act on the level of transcription are of practical interest in synthetic biology. They form a class of fast-acting regulators for gene expression that can react to the presence of small molecules via suitable aptamer components or respond to environmental changes in temperature or ion strength that physically affect RNA structure formation. The rational design of such devices, however, requires not only a detailed understanding of the folding process but also computationally efficient methods to model and evaluate the folding dynamics. The size of RNA sequences and the time scales involved preclude 3D molecular dynamics simulations.

Since the standard Turner model assigns an energy to every RNA secondary structure, it is, in principle, possible to simulate the dynamics of RNA at this level (Flamm, Fontana, et al., 2000). The calculation and analysis of a large number of trajectories is, however, computationally expensive and is infeasible at this level when folding processes with time scales of seconds or even hours are to be studied. To be of practical use, e. g., in design applications, furthermore, computational models not only need to provide reasonably reliable predictions, but they also must be much faster and cheaper than experimental approaches. One such method is *barriers*, cf. Section 2.3.4. It generates a coarse-grained representation of the underlying high-dimensional energy landscape using the notion of gradient basins, and approximates the folding dynamics of the RNA by a dynamical system on the coarse-grained landscape.

However, cotranscriptional folding cannot be modeled by a single energy landscape since the underlying RNA sequence changes with the addition of each nucleotide. The idea of *BarMap* (Section 2.3.7) is to use a sequence of (coarse-grained) landscapes, one for each step of transcriptional elongation. The refolding dynamics between elongation steps are then modeled as a Markov processes on *fixed* landscapes. Upon elongation, the populations of the macrostates in the current landscape are transferred to the next one using specifically constructed maps. This yields an approximate time-course of the occupancy of coarse grained macrostates for cotranscriptional folding without the need for expensive simulations of individual trajectories. The separation of computations on individual landscapes and the transition between them, furthermore, makes it easy to explore the effects of variations in the speed of transcription on the formation of intermediate structures.

The *BarMap* software accompanying the publication of the method (Hofacker, Flamm, et al., 2010) is a collection of scripts to facilitate the kinetic analysis of several related RNA landscapes, for example under changing environmental parameters such as temperature, or, more relevant here, for a step-wise elongated transcript. It makes use of *barriers* (Flamm, Hofacker, P. F. Stadler, et al., 2002) to coarse grain the given RNA landscape, and *Treekin* (Wolfinger et al., 2004) to compute the folding dynamics at the level of macrostates represented by gradient basins of secondary structures. The script then stitches together the output of these tools to obtain a result for the full process. In contrast to other simulation tools, *BarMap* does not rely on the sampling of individual trajectories, but offers an analytical solution based on the enumeration of representative states.

Despite its flexibility and extensibility, the original version of *BarMap* is a proof-of-concept rather than a ready-to-use tool. Its major drawbacks are:

i) The user has to call multiple scripts to get initial results, making the method unattractive for batch processing even for short input sequences.

ii) The output is spread across multiple files, some of which are hardly human-readable, making the interpretation of the results a tedious and error-prone task.

iii) There is no feedback that would allow the user to judge the quality of the produced results or help to improve on them.

iv) Required software dependencies, in particular *barriers* and *Treekin*, are developed for Linux. Although they can be compiled for other operating systems, this excludes many potential users.

v) There is no comprehensive documentation that guides the user step by step through a typical run, so getting started with the software is unnecessarily complicated.

It therefore takes considerable effort to use *BarMap* in the context of cotranscriptional folding. Its practical applications there have remained limited.

The intention of this contribution is to alleviate all of the mentioned deficiencies. We provide a pre-configured working environment including all required software to immediately jump into a cotranscriptional folding analysis. It is deployed inside a highly portable Docker image that can be loaded with a single command and requires no setup apart from a standard Docker installation, which is available on Windows, macOS and Linux. The included scripts can produce preliminary output with just a single command and an input sequence, generating a full cotranscriptional folding simulation and post-processed plots in various graphics formats including PDF and SVG. In addition, quality statistics are computed that allow the user, together with powerful helper scripts, to semi-automatically improve the simulation results if necessary. Finally, an integrating command line viewer for the output files is available that greatly simplifies the interpretation of the results and, together with the generated plots, provides a deep insight into the folding process. Detailed instructions to analyze a real-world example that the reader can easily reproduce on an average desktop computer have been provided by the author (Kühnl, P. F. Stadler, and Findeiß, 2019).

3.4.2 Theory

Measuring simulation quality

A major problem of the original *BarMap* pipeline is that one cannot easily assess the quality of the produced output. To tackle this issue, our software provides rich statistics to evaluate …

i) the *ensemble coverage* for each generated coarse-grained ensemble $\tilde{X}_i$, $i = 1, \dots, n$, to ensure all highly probable (i. e., important) structures are contained in the simulation:

$$\Pr\big[\tilde{X}_i \,|\, X_i\big] = \sum_{\alpha \in \tilde{X}_i} \frac{\mathrm{Z}[\alpha]}{\mathrm{Z}[X_i]},$$

ii) the summed probability density of all exactly mapped (as defined in Section 3.4.2) macrostates – *exact coverage* for short – for each mapping step from $\tilde{X}_i$ to $\tilde{X}_{i+1}$ to ensure correctness of the mapping in the case of an equilibrated RNA:

$$\sum_{\substack{\alpha \in \tilde{X}_i, \\ \alpha \text{ is mapped exactly to } \tilde{X}_{i+1}}} \frac{\mathrm{Z}[\alpha]}{\mathrm{Z}[X_i]},$$

and

iii) the fraction of *exactly mapped population* for the mapping from $\tilde{X}_i$ to $\tilde{X}_{i+1}$ during the kinetics simulation to ensure highly populated states are mapped correctly to the next landscape independent of their free energy:

$$\sum_{\substack{\alpha \in \tilde{X}_i, \\ \alpha \text{ is mapped exactly to } \tilde{X}_{i+1}}} \mathbf{x}_{\alpha}^{(i)}(t_i^{\infty}).$$

Here, (i) measures the quality of the individual energy landscapes generated by *barriers*, (ii) measures the quality of the macrostate mapping generated by *BarMap*, and (iii) measures the quality of the entire kinetics simulation. The coverage of a macrostate α is $\Pr[\alpha]$. The total coverage of all enumerated basins is the ensemble coverage (i), which is directly controlled by the enumeration threshold of Wuchty's algorithm and serves as a measure of completeness of the coarse-grained landscape. The exact coverage of the mapping (ii) is the sum of coverage of all macrostates that are mapped exactly to the next energy landscape by *BarMap*. Low values indicate that probable structures present in the previous landscape have not been enumerated in the next energy landscape. In such a case, the enumeration threshold needs to be increased for the next landscape and the mapping has to be recomputed. Finally, the fraction of exact population (iii) of a mapping step is given by the sum of the populations of exactly mapped macrostates at the end of this simulation step. Here, a low value also shows that an important state was not enumerated in the next energy landscape, which has to be re-enumerated with a higher threshold. The difference is that the approximately mapped state could as well have a very low coverage, but is highly populated for another reason (e. g., because it is an important transient state, or a kinetic trap).

Definition of exact and approximate mappings

As explained previously (Section 2.3.7), *BarMap* cannot always map every macrostate $[x]$ from the current landscape to some macrostate $\mu[x] = [y]$ in the next landscape, e. g., because the local minimum y has been disconnected from the next landscape by one of the heuristics applied to keep the number of states tractable. In such cases, $[x]$ is then mapped to another macrostate $[y']$ such that the minima y and y' have minimal base pair distance $d_{\mathrm{bp}}(y, y') = d$. By default, *BarMap* characterizes mappings as *exact* if $d < 1$, and as *approximate* otherwise. For the proposed quality measures “exactly mapped coverage” and “exactly mapped population”, this characterization is critical to distinguish “good” from “bad” mappings.

However, in practical application it became obvious that this definition is too strict. The prime example is a minimum `..___.`, where `.` denotes an unpaired nucleotide and `___` is some arbitrary substructure. During mapping, a new unpaired nucleotide is appended at the $3'$ end and a gradient walk is performed on the resulting structure. Since `___` is already of minimal energy, the only option is to add either of the two base pairs `.(___).` or `(.___.)` denoted by matching pairs of parentheses. If both possible base pairs destabilize the structure, then `..___..` is still a minimum in the next landscape, but `((___))` may be, too. Now, the heuristics of *barriers* are likely to remove `..___..`, because `((___))` will usually be more stable and the barrier to the other minimum is often very small. Then, the mapping of minimum `..___.` to minimum `((___))` will be considered approximate even though this is the naturally correct mapping in this case. Another source of similar cases is due to the *dangling end* energy contributions in the Turner energy model: an additional unpaired nucleotide may stabilize an adjacent base pair. In effect, a previously unstable base pair at the second-to-last nucleotide may suddenly become energetically feasible, and the basin minimum changes slightly.

To account for such subtle changes in the structure that prevent exact mappings, but which do not correspond to an actual error, the definition of approximate mappings is relaxed. A mapping is called d-exact if the base pair distance of the mapped minimum to the target minimum is less than d. Exact mappings in the strict sense are thus 1-exact mappings.

In the scripts `barmap_exact_coverage` and `barmap_exact_population` shipped with *BarMap-QA*, the value of d can be set using the `-d` switch. In practice the choice $d = 3$ has proven to minimize the occurrence of both false mappings considered as exact and correct mappings considered as approximate, and is thus the default value in *BarMap-QA*.

3.4.3 Implementation

Parallelization

To speed up the possibly lengthy computations of *BarMap*, our extension *BarMap-QA* implements a parallelization of the computation of the energy landscapes, their associated rate matrices, and the diagonalization of the latter. To achieve this, several steps and adaptations in the pipeline were necessary.

To parallelize the coarse graining of the energy landscapes, each instance of *barriers* had to be run in its own subdirectory, because hard-coded file names like `rates.out` are used to store the generated rate matrix. Running multiple instances within the same directory would overwrite these files in an undefined manner. In individual subdirectories, however, the generation of the n individual energy landscapes can trivially be distributed to up to n worker threads. Note however that running multiple instances of *barriers* on long sequences requires a lot of main memory because of the huge hash data structure storing all encountered structures. We recommend keeping an eye on the spawned threads using a system monitor and setting a per-process memory limit if supported by the operating system.

The final folding simulation on the sequence of precomputed landscapes is a linear process, where the initial condition of the simulation on the next landscape depends on the results of the previous step. A parallelization is thus not possible. By default, however, the simulation on each landscape as performed by *Treekin* consists of two steps. First, the respective rate matrix $\mathbf{R}$ is diagonalized, i. e., its eigenvalues and the respective eigenvectors are determined, to allow for an efficient computation of the matrix exponential $p(\tau) = \exp(\tau\mathbf{R})$, and then the actual populations $p(\tau)$ are computed for the requested time τ. Indeed, the first step is independent of the initial population, and can thus be factored out. This is already supported by *Treekin* through the options `--dumpE` and `--recoverE`, which allow writing and reading of the computed eigenvectors and eigenvalues to files `evecs.bin` and `evals.bin`, respectively. The pipeline thus pre-computes the eigenvalues and -vectors and stores them along with the energy landscapes and rate matrices. Now, a patch was applied to the original *BarMap* code that creates file links `evecs.bin` and `evals.bin` to the stored eigenvalues and -vectors of the current landscape before *Treekin* is called with the `--recoverE` option. Thus, the diagonalization of the rate matrices can be parallelized along with the landscape computation, and it is also not repeated when the simulation is re-run on the same set of landscapes, e. g., with a different initial population.

In the implementation, the script `barmap_rebar` allows to set the number of threads to launch via the parameter `-t`. The same parameter is also supported by `barmap_gen_barmapfile`, which performs a fully automated, complete simulation run and passes the option on to all other scripts it is calling.

Libraries for input and output file handling

BarMap-QA is a collection of scripts that repeatedly have to parse, query or transform one or more of the following file types:

- coarse-grained energy landscapes computed by *barriers* (*barriers files*)
- rate matrices characterizing the transitions between the states computed by *barriers* (*rate files*)
- population data time series computed by *Treekin* (*Treekin files*)
- mappings between macrostates of consecutive energy landscapes computed by *BarMap* (*mapping files)*
- aggregated time series of population data from multiple landscapes computed by *BarMap* via multiple calls of *Treekin* (*multi-Treekin files*)

To avoid code duplication and ensure a consistent syntax across all parts of the package, it was thus required to outsource the functionality to handle these file types into external libraries. Since *BarMap-QA* is largely written in *Perl*, three *Perl* distributions have been developed:

`Bio::RNA::Barriers` provides support for parsing and writing *barriers* and rate files. The number of macrostates and their individual properties can be queried, e. g., the basin free energy, the father minimum, connectedness, barrier heights and many more. Minima can be pruned by restricting the their total number or by removing all disconnected states. Rate files can be converted between their binary and text representation; their rows and columns can be transformed and pruned with high-level functions, and of course querying individual rates for given states is possible. Removal of disconnected states is supported for rate matrices, too.

`Bio::RNA::Treekin` allows to parse and write both regular (single-) *Treekin* files and multi-*Treekin* files. The `Record` class represents a complete time series of population data, which can be queried for specific states or time points, for the maximum population of a specific state etc. It can be spliced to a certain set of minima, re-arranged in a specific order, or new states can be added. For multi-*Treekin* files, the `MultiRecord` wrapper class allows to retrieve individual record objects from a multi-*Treekin* file.

`Bio::RNA::BarMap` implements parsing and writing for mapping files. The mapped landscape files and their respective minima can be queried, and both files and minima can be mapped by one or multiple steps. Inverse mapping is also supported, i. e., for a given macrostate in a specific landscape, all macrostates in the previous landscape that map to it are reported. The package also properly manages the type (exact or approximate) of individual or sequences of mappings, as well as their conversion to and from the symbolic notation used by *BarMap*.

The libraries have been designed following an object-oriented paradigm and are implemented using the *Moose* object system. Beside the provided functionality, all three distributions include comprehensive unit test suites as well as a complete feature documentation. To make the libraries available to the scientific community, they have been uploaded to the Comprehensive Perl Archive Network (CPAN), and can thus be installed on any computer running *Perl* by using a single command (e. g., `cpanm Bio::RNA::Barriers` using the *Perl* package manager *cpanminus*). The package documentation is also available via MetaCPAN. The libraries are free and open source software distributed under GNU General Public License, and their source code is publicly available via GitHub.

Analyzing the simulation results

It has already been mentioned that it is hard to analyze the raw output of *BarMap* by hand. Here we will describe more precisely which difficulties arise and how *BarMap-QA* helps to overcome them.

The first important output file of *BarMap* is the *mapping file*, which stores all constructed maps μ_i, $i = 1, \dots, n-1$ for the n input landscapes. It is formatted as a plain-text, fixed-width table where the column j denotes the landscape, and each row i consists of (possibly empty) integer entries $e_{i,j}$ encoding the mapping of the macrostates as follows. Let α_k^j be the k-th basin of the j-th landscape. Then any two consecutive entries $e_{i,j}$, $e_{i,j+1}$ in one row of the mapping file mean that $\mu_j(\alpha^j_{e_{i,j}}) = \alpha^{j+1}_{e_{i,j+1}}$. For example, a row with entries $5, 9, 7$ in the first three columns means that minimum 5 of the first landscape is mapped to minimum 9 in the second one, and minimum 9 is mapped to minimum 7 in the third landscape.

This way of storing the maps has a number of issues. Firstly, the fixed-width format limits the maximum number of macrostates to 9999. Secondly, if no state is mapped to a basin $\alpha^j_{e_{i,j}}$, then all previous entries $e_{i,1}, \ldots, e_{i,j-1}$ in that respective line are empty. Due to the fixed-width format, they have to be filled up with space characters, which occupy a significant fraction of the total file size. Thirdly, the mapping file often contains a large fraction of redundant data. The reason is that the constructed maps are in general not injective, i. e., different basins α^j_k, $\alpha^j_{k'}$ in one landscape k could be mapped to the same basin α^{j+1}_ℓ in the next landscape $k+1$. In the mapping file, this will be encoded as entries $e_{i,j}, e_{i,j+1}$ and $e_{i',j}, e_{i',j+1}$ in two distinct rows i and i', such that $e_{i,j} = k$, $e_{i',j} = k'$, and $e_{i,j+1} = e_{i',j+1} = \ell$. But whenever two entries in the same column are equal, so are the remaining entries of the two rows of these entries, since otherwise the mapping would be ambiguous. Thus, $e_{i,j+2} = e_{i',j+2}, \ldots, e_{i,n} = e_{i',n}$. Since mappings of more than one state to the same target arise regularly, the mapping file is full of partially duplicated lines. Fourthly, the minima in each column are in general not sorted. Since, for a given set of maps, the first entry of each row determines the remaining entries of that row and the maps usually do not conserve the order of the mapped states, this is simply not possible with this encoding. As a consequence of the described issues, the mapping file of any non-trivial simulation is usually so bloated that it is almost impossible to analyze by hand.

The second important file generated by *BarMap* contains the concatenated output of the n individual runs of *Treekin*, i. e., the population data for each landscape. We refer to it as the *kinetics file*. Each *Treekin* output consists of some header lines containing metadata, and a time series of populations for each macrostate. The last line (i. e.,, time point) of each time series provides the populations at the end time of that respective simulation and serves as initial population for the next landscape.

What makes the kinetics file so hard to interpret manually is (i) the sheer number of macrostates and time points, (ii) that the populations are given in a plain-text table format, where the position of the value encodes the macrostate it describes, while the structures of the minima are stored in n different files, and (iii) that the order of the macrostates of each landscape depends on the free energy of their minima, and is thus not related to the mappings used to transfer the populations from one energy landscape to the next. To trace the population of a single structural conformation of interest through the course of the simulation, the user thus has to look up the index i of the macrostate of interest in the input file of the current landscape j, identify column i in the time series of landscape j in the kinetics file, track the numbers until the end of the time series, consult the mapping file on how to map state i in landscape j to the corresponding state i' in landscape $j+1$, and finally look up macrostate i' in the input file for landscape $j+1$ to see whether its structure has changed. This process has to be repeated for every added nucleotide. It is needless to say that it is impossible to quickly analyze the behaviour of the input RNA using such a laborious and error-prone procedure.

To alleviate these deficiencies, *BarMap-QA* provides a convenient viewer for the computed simulation run. The command line tool integrates the kinetics time

```
# '18.bar' [...]
#                   CAUGACUUGAACCCAUGG
 90.45% <= 61.86% 1 ((((.........)))).  [...]  -> 1  <= {1}
  6.12% <= 31.26% 2 ..................  [...]  -> 2  <= {2}
  1.25% <=  2.67% 4 .....(.((....)).).  [...]  -> 6  <= {4}
  0.71% <=  3.65% 5 ((.....)).........  [...]  -> 7  <= {3}
```

Listing 1: Shortened output of the results viewer for a folding simulation of RNA sequence `CAUGAC...`, showing the change in population of four states (labelled 1, 2, 4, and 5) in energy landscape `18.bar`. State 3 was filtered out because its population was too low. While the open chain (state 2) has a population of 31% at the beginning, it has only 6% population remaining at the end time of this landscape. A stable hairpin (state 1) gains up to 90% of population. The last columns show mapping information for each state. The last line, for instance, indicates that state 3 from the previous landscape is mapped to state 5 in this landscape, and state 5 is in turn mapped to state 7 in the next landscape.

series with the macrostate maps and the *barriers* files describing the landscapes, and thus can show population data along with structural conformations as well as mapping information for each state. It also offers filtering options to remove unimportant conformations. Used in conjunction with the kinetic plot, it is the main tool to analyze the results of the folding simulation. Exemplary output is shown in Listing 1.

Post-processing and visualization

According to a well-known saying, a picture is worth a thousand words. Thus, *BarMap-QA* includes functionality to generate a *kinetics plot* from the data of a given simulation run. These are line plots of the population time series in the multi-*Treekin*, where each line denotes the population (y-axis) of a macrostate during the course of time (x-axis). One line per macrostate is drawn. This type of plot provides an overview of the entire folding process. It allows to quickly judge the overall behaviour of the molecule, e. g., to answer whether there are one or multiple populated states at a given time, or at which points in time one macrostate dominates another one. Unfortunately, the plot does not contain the actual conformations of the plotted states. Since these are permanently changing with every added nucleotide, this is not easily possible in general. The kinetics plot therefore has to be analyzed in conjunction with the command line viewer discussed in Section 3.4.3, as the latter provides the structure and exact population of each macrostate. The kinetics plots are generated using the *Grace* data visualization package (The Grace contributors, 2015). The output of PDF files, SVG vector graphics, and lossless PNG pixel graphics is supported.

The raw kinetics file as generated by *BarMap* contains lots of data points, many of which are not relevant to the overall behavior of the RNA. To achieve pleasing results when plotting the data, it is therefore necessary to apply some post-processing steps to it. As a first step, macrostates with very low populations are removed. To this end, the user selects a threshold (by default, 0.5%), and any macrostate with a maximum attained population below the threshold is removed. This will significantly reduce the number of plotted lines and thus make the plot cleaner and easier to interpret. Next, the user can define an end point for the data if the quality metrics indicate unreliable results beginning at a specific point in time. Any data beyond the defined end point is then truncated.

Finally, a merging procedure can be applied that joins all population time series of the individual landscapes from the kinetics file into a single time series record. This solves the problem that the individual population data time series provide no information about the mapping of individual macrostates to the next landscape. When only considering the raw kinetics file, it is not possible to track the course of a single macrostate across multiple energy landscapes, and thus one cannot, e. g., assign a common color to all segments of the corresponding curve in all landscapes. After merging, this task becomes trivial. To join two consecutive population data time series, the procedure has to rearrange the columns of states that are mapped onto each other and then concatenate the time series. If the second time series contains a state that no states of the first time series are being mapped to, a new columns has to be created for it. Its population values are initialized with 0.

An example of a generated kinetics plot is shown in Figure 11 and discussed in Section 3.4.4.

Software

The following scripts are provided to the end user by our software package:

`barmap_gen_barmapfile` automatically performs a full cotranscriptional folding simulation for a given input sequence. Supports multi-threading.

`barmap_rebar` is used to re-enumerate individual energy landscapes up to a given energy threshold. It also re-renders the transition rate matrices and pre-computes eigenvalues and -vectors, cf. Section 3.4.3. Supports multi-threading.

`barmap_remap` recomputes the macrostate mapping for the (updated) energy landscapes and optionally re-runs the final kinetics simulation.

`barmap_filter_treekin` is a post-processing tool that filters the simulation data to remove states with low population, cf. Section 3.4.3.

`barmap_merge_treekin` is a post-processing tool that joins all records of the given multi-*Treekin* (simulation data) file into a single record to improve the plotting results, cf. Section 3.4.3.

`barmap_show_run` is a command line viewer for the generated kinetics file, which also integrates the data of the macrostate maps and the structural conformations, cf. Section 3.4.3.

`barriers_coverage` computes coverage and connectedness statistics for the *barriers* files of the energy landscapes, cf. Section 3.4.2.

`barmap_exact_coverage` computes exactly mapped coverage for each generated map between two consecutive energy landscapes, cf. Section 3.4.2.

`barmap_exact_population` computes the fraction of exactly mapped population for each mapping step between two consecutive energy landscapes, cf. Section 3.4.2.

`barmap_plot_treekin` generates the kinetics plot from the (post-processed) multi-*Treekin* file, cf. Section 3.4.3.

The software accompanying this work is deployed inside the Docker image `xilef1337/barmap-qa` hosted on Docker Hub. It can thus be downloaded by issuing the command `docker pull xilef1337/barmap-qa`. The only prerequisites are an installation of Docker (Community Edition) and at least 8 GB of main memory. Docker currently supports Microsoft Windows 10 (Pro, Enterprise and Education edtitions), Apple macOS as well as many Linux distributions (e. g., Ubuntu, Debian, Fedora, etc.). As an additional convenience, a wrapper script `run_barmap-qa.sh` for the *Bash* shell is provided, which takes care of updating the local Docker image, running the container and mounting the working directory. It is available via the official *BarMap-QA* homepage[1].

Alternatively, a source distribution for Linux is offered for download. It has been tested on the distributions Fedora 27 and 30 as well as Ubuntu 18.04 LTS. The user is responsible for providing all external dependencies as explained in the `README` file. Afterwards, the shell script `install.sh` performs all necessary setup steps. We highly recommend to run the test suite by executing the command `make test` inside the distribution directory after the setup script completed successfully to ensure the environment properly configured.

3.4.4 Application

Typical analysis work flow

Assume that a cotranscriptional folding analysis is to be conducted for a given sequence. Here, we will describe shortly how to proceed to do so with *BarMap-QA*. The following steps are to be executed in order, unless explicitly stated otherwise, cf. Figure 10.

i) Run `barmap_gen_barmapfile` to perform a fully automated cotranscriptional folding simulation for the given input sequence, including elongation, energy landscape generation, rate computation, landscape mapping, and output file filtering. A low enumeration threshold should be chosen (e. g., $\Delta G_{\text{enum}} = 5\,\text{kcal}\,\text{mol}^{-1}$) to quickly get preliminary results. This step supports multi-threading.

ii) Assess the quality of the simulation run:

    a) Use `barriers_coverage` to check for sufficient coverage of the individual energy landscapes.
    b) Use `barmap_exact_coverage` to check for a sufficiently exact mapping of coverage.
    c) Use `barmap_exact_population` to check for a sufficiently exact mapping of populations during the folding simulation.

    If the simulation quality is good, go to (v).

iii) Employ `barmap_rebar` to re-enumerate individual energy landscapes up to a higher enumeration threshold $\Delta G'_{\text{enum}} > \Delta G_{\text{enum}}$. This also re-renders

---

[1] `https://www.bioinf.uni-leipzig.de/Software/BarMap_QA/`

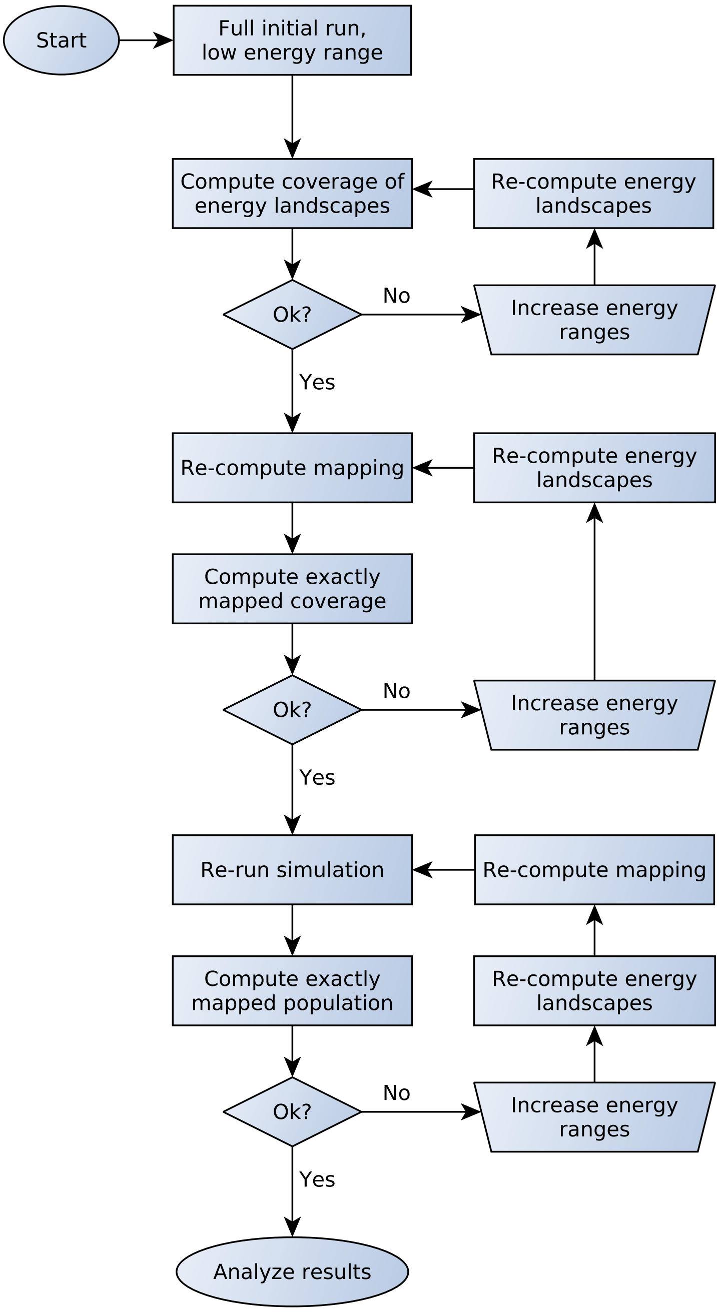

Start
Full initial run, low energy range
Compute coverage of energy landscapes
Re-compute energy landscapes
Ok?
No
Increase energy ranges
Yes
Re-compute mapping
Re-compute energy landscapes
Compute exactly mapped coverage
Ok?
No
Increase energy ranges
Yes
Re-run simulation
Re-compute mapping
Compute exactly mapped population
Re-compute energy landscapes
Ok?
No
Increase energy ranges
Yes
Analyze results

Figure 10: Flowchart of a typical work flow for performing a cotranscriptional folding analysis with *BarMap-QA*. After a quick, initial simulation run with a low energy range parameter, the three quality criteria landscape coverage, exactly mapped coverage and exactly mapped population are computed for each landscape and map, respectively. If any value is found to be too low, the user can adjust the enumerated energy ranges for the problematic energy landscapes. The model is then regenerated and the quality scores are re-computed. This is repeated until the model performs well. Now, the results of the folding simulation may be evaluated.

the transition rate matrices of the updated landscapes, and re-computes their eigenvalues and -vectors. This step supports multi-threading.

iv) Use `barmap_remap` to re-compute the macrostate mappings and, optionally, to re-run the folding simulation after the energy landscapes have been changed in the previous step. Go back to (ii).

v) Analyze the generated output using the command line viewer and the generated kinetics plot.

vi) If necessary, use `barmap_merge_treekin` with adapted filter settings to improve the kinetics plot. Re-compute the plot with `barmap_plot_treekin`.

Already the first step runs the entire simulation and creates all required files. The choice of a low enumeration threshold ensures that this initial run terminates quickly. However, the results will usually be bad. Now the analyst can iteratively refine individual landscapes by increasing the respective enumeration thresholds until the quality is sufficiently high. If the coverage of an energy landscape is low, only step (iii) has to be performed before re-evaluating the quality score. If the fraction of exactly mapped coverage is low, both step (iii) and step (iv) have to be executed, but re-running the folding simulation is not necessary until the quality metric has improved.

As soon as all quality scores are sufficiently high, the results of the simulation can be evaluated and the folding kinetics of the RNA may be be studied. Optionally, the simulation can be repeated with varying simulation time parameters to assess the effect of a varying transcription rate to the folding process. This can be achieved efficiently by running `barmap_remap` with the option to disable the re-computation of the macrostate mappings. Changing the energy landscapes, rate matrices and mappings is not required in this case.

Output files

A table describing all output files generated by a full run of *BarMap-QA* can be found in Appendix B.

Usage example

To demonstrate the usage of *BarMap-QA* for a specific example, the cotranscriptional folding kinetics of a riboswitch shall be analyzed. Riboswitches in general are described in more detail in Chapter 5. Wachsmuth, Findeiß, et al. (2013) engineered multiple synthetic, transcriptional riboswitches responding to the ligand theophylline, and verified their *in vivo* functionality by transfection into *Escherichia coli* (*E. coli*). Of six tested candidates, the 68 nt long sequence RS10[2] performed best, showing a 3-fold activation ratio in in an ONPG test (cf. Section 5.1.4). A cotranscriptional folding analysis of RS10 should thus show an interesting switching behavior between at least two dominant states: (i) the binding-competent aptamer state, which is able to sense the ligand theophylline, and (ii) the terminator state, which disrupts the aptamer conformation and triggers transcription termination.

---

[2] 5′-`AAGUGAUACCAGCAUCGUCUUGAUGCCCUUGGCAGCACUUCAGAAAUCUCUGAAGUGCUGUUUUUUUU`-3′

```
file_name [ ... ] fulRangeNrg EnsembleNrg [ ... ] fulCovr lowCv
  [ ... ] [ ... ]
   21.bar [ ... ]   -1.654210   -1.665671 [ ... ]  98.16% **
   22.bar [ ... ]   -2.003425   -2.050692 [ ... ]  92.62% ****
   23.bar [ ... ]   -2.003425   -2.061808 [ ... ]  90.96% ****
   24.bar [ ... ]   -2.003425   -2.074347 [ ... ]  89.13% *****
   25.bar [ ... ]   -2.003425   -2.458591 [ ... ]  47.78% *****
   26.bar [ ... ]   -5.368760   -5.372944 [ ... ]  99.32% *
  [ ... ]
```

Listing 2: Energies of the enumerated range (`fulRangeNrg`) and the full ensemble (`EnsembleNrg`), and the corresponding coverage values (`fulCovr`) for the first 21 to 26 nt of RS10. Asterisks (`*`) mark landscapes with low coverage values.

Initial landscape generation. After running and connecting to the *BarMap-QA* Docker container using the command `./run_barmap-qa.sh .`, the software provided by *BarMap-QA* becomes available and the current working directory "`.`" is mounted. In the following, it is assumed that the sequence of RS10 is available in a variable of the same name. Next, the initial sequence of energy landscapes can be generated by executing the command `barmap_gen_barmapfile -t 8 -E 5 $RS10`. Here, multi-threading with up to eight threads (`-t 8`) is combined with a small energy range of 5 kcal mol^{-1} (`-E 5`) to quickly finish this first run.

Coverages of the landscapes. The coverage of the generated energy landscapes is automatically computed, but can also be recomputed using the script `progname`, as shown in Listing 2. Besides other statistics not shown here, the ensemble free energy (`EnsembleNrg`), the total energy of all explored basins (`fulRangeNrg`), and the resulting coverage values are shown for each energy landscape `n.bar`, where $\texttt{n} \in \{21, \ldots, 26\}$ is the current length of the RNA molecule. While the coverage is as high as 98% for `21.bar`, it starts to decrease in the following landscapes and drops to only 48% in `26.bar`. The next landscape has a high coverage of 99% again, and its ensemble energy has more than doubled, indicating that a new low-energy conformation became available with the addition of the 26th nucleotide. These abrupt changes in coverage once more stress the importance of quality criteria to ensure a reliable model performance.

To improve on the low coverage detected in some of the landscapes, the explored energy range needs to be increased. In this example, it suffices to increase the range to 8 kcal mol^{-1} for *all* energy landscapes by re-running the `barmap_gen_barmapfile` script with argument `-E 8`. The re-evaluation of the resulting energy landscapes now shows that each has a coverage of at least 97%, i. e., the structures disregarded by the analysis correspond to less than 3% of the probability mass of the entire ensemble.

Exactly mapped coverage. We now proceed to the evaluation of the quality of the "bar map", i. e., the mapping from each coarse-grained energy landscape to the next one. As before, the quality measures for the mapping are generated automatically by `barmap_gen_barmapfile`. To manually recompute the scores, the script `barmap_exact_coverage -a` may be used. The option `-a` enables

```
from_file to_file #exact #basin %exactBsn %%exactCov %totalCov %exactCov lowCv
--------------------------------------------------------------------------------
[ ... ]
25.bar ->  26.bar     21     61    34.43%     50.44%     99.97%    50.42% *****
    43.89%:    1 (((((((......)))).)))....    -1.70    0    8.00  saddle:  -1.70
         ~>    4 ..........................    0.00    1    2.70  dist:     7 bp
     3.95%:    4 ...((((......))))........     0.10    1    0.40  saddle:   0.50
         ~>    4 ..........................    0.00    1    2.70  dist:     4 bp
[ ... ]
```

Listing 3: Approximately mapped minima of the mapping between landscapes `25.bar` and `26.bar`. Minimum 1 of `25.bar`, represented by a hairpin loop containing a bulge, is mapped to the open chain in `26.bar` despite a distance of seven base pairs. The reason is that the saddle structure at $-1.7\,\mathrm{kcal\,mol^{-1}}$ was not enumerated in landscape `26.bar`, and the – now disconnected – corresponding minimum was thus removed. This results in 44% of the coverage of landscape `25.bar` to be mapped to the wrong state. Another 4% of coverage is lost due to the erroneous mapping of minimum 4 to the open chain.

the output of individual miss-mapped minima. An excerpt of the result is shown in Listing 3.

The first significant problem is encountered when mapping landscape `25.bar` to `26.bar`. Specifically, minimum 1 of `25.bar` is mapped to the open chain in `26.bar` despite a distance of seven base pairs. The reason is that the saddle structure at $-1.7\,\mathrm{kcal\,mol^{-1}}$ was not enumerated in landscape `26.bar`, and the – now disconnected – corresponding minimum was thus removed. This results in 44% of the coverage of landscape `25.bar` to be mapped to the wrong state. Again, this and similar deficiencies can be solved by increasing the explored energy range for all landscapes; this time we choose a range of $9\,\mathrm{kcal\,mol^{-1}}$, which is still a modest value for the given RNA length. As a result, the fraction of exactly mapped coverage increases up to at least 97% for all mapping steps.

Exactly mapped population. Now, the third quality criterion measuring the fraction of exactly mapped population can be optimized for the given landscapes. The computation of the scores is performed by `barmap_exact_population -a`. Again, only the first significant issue reported is shown, cf. Listing 4.

The result looks quite similar to Listing 3, as the same conformation, this time corresponding to minimum 4 in landscape `26.bar`, is miss-mapped to the open chain in the next landscape. This minimum, however, no longer has a significant coverage in its landscape, because other, more stable states now dominate the equilibrium distribution. It is thus not detected by the criterion based solely on the coverage of the mapped states. But due to the dominance of the corresponding conformation in the previous landscape, a high fraction of population is mapped to minimum 4 anyway, and remains there due to a high energy barrier of $5.2\,\mathrm{kcal\,mol^{-1}}$. The exact population criterion successfully detects this mapping error and demonstrates that an enumeration up to $3.5\,\mathrm{kcal\,mol^{-1}}$ is necessary to include the saddle structure required to connect the corresponding minimum to the global minimum in landscape `27.bar`. Otherwise, a loss of more than 54% of the population will occur, and the simulation results should be considered unreliable.

```
from_file to_file %exactPop lowPop
-----------------------------------
[ ... ]
26.bar -> 27.bar      40.02 *****
      54.80%: 4 (((((((......)))).))).....  -1.70  1  5.20  saddle: 3.50
           ~> 4 ...........................  0.00  1  2.70  dist:   7 bp
       4.87%: 6 ...((((......)))).........   0.10  4  0.40  saddle: 3.50
           ~> 4 ...........................  0.00  1  2.70  dist:   4 bp
[ ... ]
```

Listing 4: Approximately mapped population of the simulation between landscapes `26.bar` and `27.bar`. Minimum 4 of `24.bar`, represented by a hairpin loop containing a bulge, is mapped to the open chain in `27.bar` despite a distance of seven base pairs. While this situation seems similar to the one in Listing 3, it shows that the exact population criterion successfully detects relevant miss-mappings during the simulation even if the coverage of the mapped minimum is low in equilibrium.

As a remedy, one may ponder to repeat the previous strategy and simply enumerate *all* energy landscapes up to the saddle point at 3.5 kcal mol^{-1}. When considering the very low global MFE of -27.6 kcal mol^{-1} for the full-length sequence, however, it becomes apparent that this is not a feasible option. A – quite lengthy – run of *RNAsubopt* enumerating all structures in the energy range of 31.1 kcal mol^{-1} above the MFE reports a total of about 4.9 billions of structures; by far too many to serve as input to *barriers*. But, as the population of an unstable minimum is going to decrease during the simulation, an alternative strategy is to only increase the enumerated energy range for landscape `27.bar` and the following ones, e. g., up to `30.bar`. A partial re-enumeration is achieved by using the script `barmap_rebar` on selected. In this vein, the energy ranges of the remaining landscapes is carefully increased step by step until proper quality scores are achieved. The full details of the procedure for this sequence are presented to the reader in a dedicated publication (Kühnl, P. F. Stadler, and Findeiß, 2019).

Evaluating the simulation results. The kinetics plot of the final simulation is presented in Figure 11. It can be analyzed in conjunction with the text-based output of the result viewer `barmap_show_run`, and shows three distinct phases during the folding process. These are *(A)* the transcription and formation of the theophylline aptamer, *(B)* the ligand sensing phase, and *(C)* the terminator formation. If the sensing phase was too short, the ligand could not bind well to the aptamer and thus not switch on the gene expression. If it was too long, in contrast, the formation of the terminator would not happen in time and it could not stop the RNA polymerase from transcribing the RNA even in absence of the ligand.

3.4.5 Discussion

The simulation of RNA folding kinetics is a powerful tool for the *in silico* analysis of these biomolecules. Models incorporating cotranscriptional folding are especially well suited to study RNAs whose function critically depends on the timing of their structural rearrangements after transcription like, e. g.,

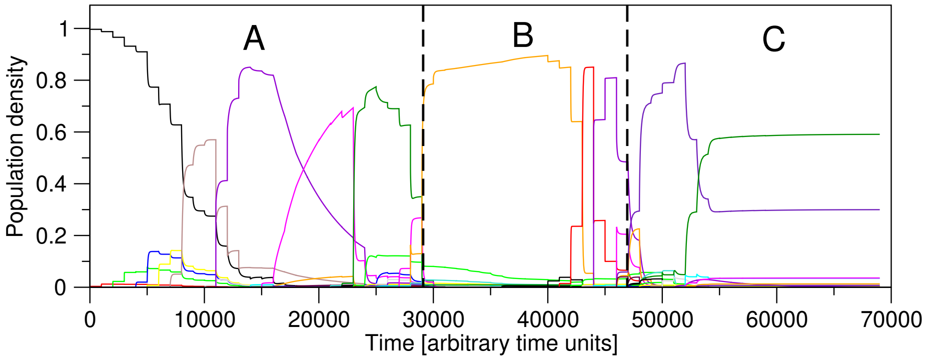


Figure 11: Final results of the cotranscriptional folding simulation of RS10 with *BarMap-QA*. The line plot can be split into three functional phases: *(A)* the transcription and formation of the theophylline aptamer, *(B)* the ligand sensing phase, and *(C)* the terminator formation. Although curves may appear and disappear within the same phase, it can be verified using the result viewer that they represent similar conformations, here the binding-competent aptamer conformation at the $5'$ end, or the terminator hairpin at the $3'$ end of the transcript in *(B)* and *(C)*, respectively. The switching between ligand sensing and terminator formation happens at time 47 000 (given in arbitrary units).

transcriptional riboswitches. *BarMap* is an – in theory – elegant framework to simulate a folding process on a sequence of dynamically changing energy landscapes. In practice, however, its application is difficult and requires a lot of manual steps. Analyzing the output is cumbersome and it is not easily possible to verify the integrity.

Our software *BarMap-QA* dramatically improves on this situation. The entire process of constructing energy landscapes for the transcription intermediates of the input sequence, associated transition rate matrices, the computation of the macrostate mappings, and finally the orchestration of the simulation run can now be performed using a single command. We provide theoretically well-grounded criteria to assess the simulation quality of the different components of the model. This allows the analyst to perform a guided improvement only of those parts of the simulation where it is really necessary, which helps to avoid unnecessary computations. Still, the computational cost of this method is high and may quickly become very challenging for RNAs longer than 100 nt.

Apart from the length of the input sequence, the feasibility of a *BarMap* analysis also depends on the sequence's folding characteristics. An RNA molecule with a smooth energy landscape with a dominant global minimum will be much easier to simulate than bistable sequences exhibiting highly probable alternative conformations or "folding traps". The reason is that persistent intermediate folding states require an enumeration up to a constant energy level during the entire elongation process, leading to a combinatorial explosion of the number of secondary structures. In contrast, if the most likely folding paths quickly lead to the global minimum, only a small number of structures need to be enumerated to obtain high quality simulation results even for long RNA sequences.

BarMap is designed to be applicable to a broad range of problems and thus very flexible. Due to the semi-automatic architecture of *BarMap-QA*, this flexibility is transparently propagated despite the added functionality. For instance, the user can provide custom rate matrices accounting for ligand binding or changing environmental temperatures, and may still compute the quality statistics as described above. *pourRNA*, a memory efficient and parallelized alternative to *barriers*, has been published recently (Entzian and Raden, 2020). Integrating this new tool into *BarMap-QA* may be an interesting option for a future release.

Apart from facilitating cotranscriptional folding analysis by providing a highly portable Docker container of *BarMap-QA*, we also provide flexible yet robust *Perl* libraries to ease the processing of many file types commonly used for folding simulations. Thus, we hope to encourage our fellow scientists to write and publish their software building on the same powerful tool set, making the analysis of RNA folding even easier and better.

3.5 Conclusion

The simulation of their folding kinetics is a promising approach to study the structural rearrangements of RNAs at a resolution which is hardly accessible through wet lab experiments. The combinatorial explosion of the number of possible structures with increasing sequence length, however, precludes modelling the folding process at the level of microscopic transitions on the full ensemble. Thus, an array of heuristics, simplifications and coarse graining approaches has to be employed to be able to analyze even moderately sized molecules. Their application has to be paid with the price of a reduced model accuracy. While an overwhelming majority of all possible structures is irrelevant to the biological function of any RNA and can safely be excluded from a folding analysis, it is not obvious *a priori* how to precisely and automatically tell important and unimportant conformations apart, or more generally, how to judge the impact of any such heuristic that reduces the number of states or transitions in an RNA energy landscape.

Therefore, this chapter was dedicated to the development of methods for assessing the quality of various models for RNA folding. A key idea was to interpret the probability of a set of structures in thermodynamic equilibrium as the coverage of that set with respect to the ensemble. This concept was then be applied to various different cases. The coverage of low-energy bands as simplest case was studied for various enumeration thresholds. Such an analysis is quickly performed for given sequence, and the obtained results are very informative. It should therefore be a standard procedure before continuing with the construction of an energy landscape.

Furthermore, the concept of coverage was applied to canonical structures to analyze the impact of neglecting structures containing isolated base pairs. Surprisingly, it was found that a significant fraction of non-canonical structures only has isolated base pairs that are in fact stable within the Turner energy model. The idea to generally exclude non-canonical structures from analysis should therefore be considered very carefully only be an option if it is really necessary.

The theoretically elegant *BarMap* framework has proven to be difficult to apply in practice. Therefore, the software package *BarMap-QA* has been developed, which does not only simplify and automate the application of *BarMap* for cotranscriptional folding, but also introduces several novel criteria to assess the quality of the folding simulation. Again, the concept of coverage was employed to judge the reliability of the different components of the model. While cotranscriptional folding simulations remain computationally challenging tasks, the availability of these quality scores allows a guided adjustment of the enumeration threshold for each individual energy landscape and thus helps to avoid unnecessary calculations. The easy installation of *BarMap-QA* via a portable Docker image significantly lowers the burden to apply the method in practice. It can thus be expected to have a significant impact on the ability of the research community to conduct high quality cotranscriptional folding analyses.

CHAPTER 4

Statistics of Free Energies for Sequences, Structures, and Gradient Basins

Contents

4.1 The distribution of minimum free energies of random sequences . . . **56**
  4.1.1 Methods . . . 56
  4.1.2 Results . . . 58
  4.1.3 Discussion . . . 62
4.2 The statistics of canonical structures . . . **63**
  4.2.1 Methods . . . 64
  4.2.2 Results . . . 64
  4.2.3 Discussion . . . 65
4.3 The statistics of low-energy minima . . . **65**
  4.3.1 Methods . . . 66
  4.3.2 Results . . . 66
  4.3.3 Discussion . . . 67
4.4 The distribution of free energies within gradient basins . . . **68**
  4.4.1 Methods . . . 68
  4.4.2 Results . . . 69
  4.4.3 Discussion . . . 71
4.5 The distribution of probability mass in gradient basins . . . **71**
  4.5.1 Results . . . 72
  4.5.2 Discussion . . . 75
4.6 Conclusion . . . **76**

In Chapter 3, secondary structures have been formally described and energy landscapes have been introduced as a flexible and powerful model for RNA folding kinetics. Additionally, the coarse graining of energy landscapes based on gradient walks was explained, which partitions their states into a set of gradient basins, each of which is represented by a local minimum. This chapter presents statistical analyses of data generated by applying these models, characterizing their universal properties. Insights about the distribution of probability mass in the landscape are extremely helpful to predict the quality of simulation results and to guide the exploration of huge landscapes. This allows to effectively reduce the number of structures that have to be considered while controlling the error introduced by this simplification.

Various existing studies have analyzed statistical aspects of RNAs and their structures. Fontana, Konings, et al. (1993) analysed the properties of random RNAs up to a length of 100 nt and compared them to natural RNA sequences. Their findings include the mostly linear dependence of the mean number of loops as well as stacks from the sequence length, a quick convergence of the mean size of stacks and loops, and a linear increase of the number of components (i. e., closed substructures in the exterior loop) after an initial "lag phase" until a minimum sequence length required to fold into stable substructures is reached. Beside the biophysical alphabet $\{A, U, G, C\}$, these authors also analyzed sequences derived from other alphabets and found them to exhibit significantly different properties. On the other hand, it was shown that many of these properties are *not* sensitive to the specific choice of the energy model parameters or the folding algorithm (Tacker et al., 1996). Properties of GC- and AU-sequences were analyzed, e. g., the number of MFE structures for varying sequence length as well as their frequencies at fixed sequence lengths, which was shown to follow a power law distribution (Grüner et al., 1996). Another work shows that for almost all MFE structures, a sequence with that MFE can be found within a small distant to any initial sequence (Fontana, P. F. Stadler, et al., 1993). Wolfsheimer and Hartmann (2010) have analysed the distribution of minimum free energies for given sequence lengths and found the distribution to be slightly skewed. Some of their results will be revisited in the first section of this chapter.

4.1 The distribution of minimum free energies of random sequences

As a first step towards the subject, this section will discuss the distribution properties of minimum free energies of random RNA sequences of a given length. The results are useful in contexts where many sequences with specific energetic properties are analyzed, e. g., in the design of RNAs. A sound knowledge of the energetic properties of sequences also allowed to make smarter choices in the following analyses, e. g., because the expected MFE or their variance was known and considered.

4.1.1 Methods

For each of various sequence lengths ranging from 20 nt to 300 nt, 10 000 random sequences were generated using a custom Perl script, invoking a

Mersenne twister (Matsumoto and Nishimura, 1998) implemented in the module `Math::Random::MT::Auto` for random number generation. The MFE for each sequence was determined using *RNAfold* from the *ViennaRNA Package* (Lorenz, Bernhart, et al., 2011). Statistical analyses and visualizations were performed in *R* using several additional packages: `genefilter` for mode computations, `e1071` for skewness analyses, `truncnorm`, `sn`, `evd`, `ExtDist`, and `GeneralizedHyperbolic` for probability density and cumulative distribution functions of the truncated normal, skew normal, generalized extreme value, Burr, and generalized hyperbolic distributions, respectively; `goftest` for the Cramér–von Mises statistic, and `ggplot2` for plot generation. The `MASS` package was used to fit the data to the cognate distributions.

The sample skewness of sample $x_1, \dots, x_n$ with n observations has been calculated as $b_1 = m_3/s^3$, where $m_3 = n^{-1} \sum_i (x_i - \bar{x})^3$ with mean $\bar{x} = n^{-1} \sum_i x_i$ is the sample third central moment, and $s = \sqrt{(n-1)^{-1} \sum_i (x_i - \bar{x})^2}$ is the sample standard deviation.

The *adjusted* skewness of x was calculated by computing the skewness b_1 of the filtered vector $x_{\geq 2\bar{x}}$ instead, which contains only those values of x that are equal to $2\bar{x}$ or greater. The idea here is to symmetrize the values of x around their mean, such that the calculation of the third central moment only considers values in the left tail for which there are equally distant values in the right tail before the point of truncation. This way, the “missing information” due to truncation is not interpreted as skewness. Of course, this procedure underestimates the skewness and thus serves as a lower bound.

To determine the p-value for the skewness $b_1(x)$ of a sample x of size k, 10 000 samples $y^{(i)}$, $i \in \{1, \dots, 10\,000\}$, of size k were drawn from a normal distribution with the same mean and standard deviation as x. Their skewnesses $b = (b_1(y^{(1)}), \dots)^T$ have been calculated. These values follow a normal distribution, as has been verified using a Q–Q plot (data not shown). The p-value of the skewness of x was then estimated as $\Pr[X \leq b_1(x)]$, using the cumulative distribution function of a normal distribution with mean 0 (as the normal distribution is not skewed) and the sample standard deviation of the vector of observed skewnesses b.

Distributions were fitted using the maximum-likelihood approach, i. e., by maximizing their log-likelihood functions giving the observed data. The optimisation was carried out using the Nelder–Mead (Nelder and Mead, 1965), BFGS (Fletcher, 2008) and L-BFGS-B (Zhu et al., 1997) procedures for truncated distributions, and distributions with unbounded or bounded parameters, respectively.

The goodness of fit of the various distribution families was compared using both the Akaike information criterion (AIC) and the Cramér–von Mises (CvM) statistic. The AIC is defined as $2k - 2\ln(\max_\theta \mathcal{L}(\theta \mid x))$, where $\mathcal{L}$ is the likelihood of the (vector of) parameters θ of the distribution given the sample x, and k is the number of degrees of freedom of the distribution, i. e., the number of components of θ (Akaike, 1974). For comparing multiple sequence lengths, the difference of the AICs of each distribution family to the minimum AIC for this sequence length was computed. This was done because the AIC is a relative criterion, and the absolute values for models of different sequence lengths – fitted to different data – thus cannot be compared. The CvM statistic T for comparing a single, *ordered* sample $x_1 \leq \dots \leq x_n$

of n observations against a theoretical probability distribution function F is $T = (12n)^{-1} + \sum_i [(2i-1)/(2n) - F(x_i)]^2$ (Csörgő and Faraway, 1996).

4.1.2 Results

Especially for longer sequences of 100 nt or more, the distribution of MFEs looks – at a first glance – much like a normal distribution (Figure 12a), i. e., it is a continuous, unimodal, bell-shaped curve. As discussed below, it is mostly, but not completely symmetric. The mean and variance of the MFE linearly depend on the number of nucleotides, cf. Figure 12. To determine the precise relation of the quantities, linear models were fitted to the data, yielding

$$\mu = -0.3\,\mathrm{kcal\,mol^{-1}} \cdot \ell + 6.3$$

and

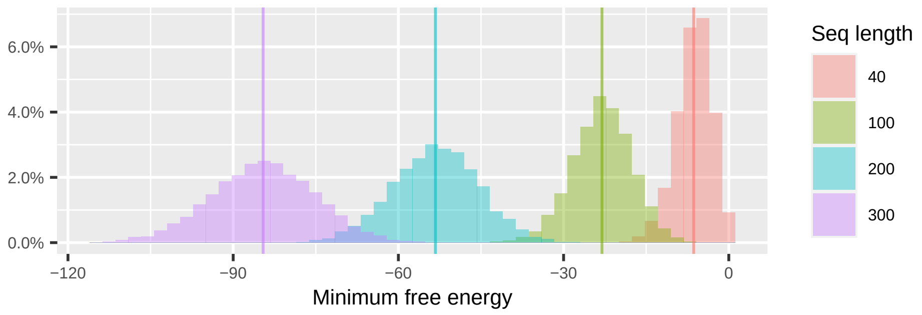
$$\sigma^2 = 0.31\,\mathrm{kcal\,mol^{-1}} \cdot \ell - 1.9,$$

where ℓ is the sequence length, μ is the mean and σ^2 is the variance. This shows that for every additional 10 nt of sequence length, the mean MFE decreases by about 3 kcal mol^{-1}, while its variance increases by the same amount (Figure 12b). This is consistent with previous estimates given by Wolfsheimer and Hartmann (2010). This functional relation of sequence length and expected stability can, e. g., be used to increase the efficiency of sequence sampling in an RNA design problem: if a specific stability of the RNA is required, a sequence length with a mean MFE close to the required value should be chosen. This ensures a high number of candidates and thus efficient sampling. The increasing variance, however, also means that, for longer sequences, no assumptions about stability should be made based solely on its length. It should also be noted that, for every sequence length, there are sequences with an MFE of 0 kcal mol^{-1}, e. g., a sequence consisting exclusively of adenine.

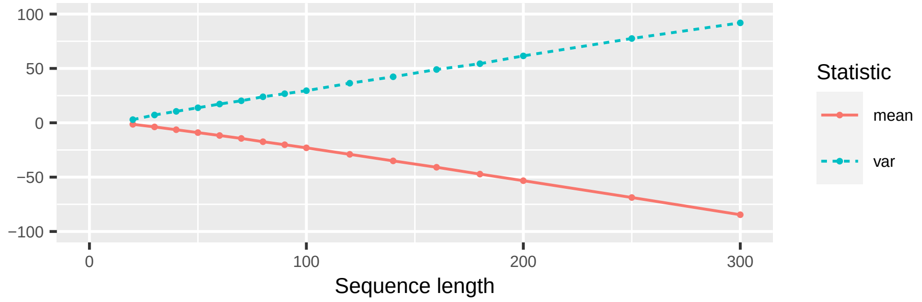
Despite the *seeming* normality observed for longer sequences, the shape of the distribution deserves further attention. By definition, the MFE of an RNA molecule cannot be higher than 0 kcal mol^{-1}, which is the free energy of the open chain. Therefore, especially short sequences, having a mean MFE just below zero, exhibit an obvious truncation of their right-hand tail at zero (Figure 13). For sequences longer than about 70 nt, no significant truncation can be observed anymore.

But even for longer sequences, the distribution is *not* perfectly normal, especially in its tails. This becomes apparent when considering a Q–Q plot of the data against the theoretical quantiles of the normal distribution (Figure 14). Both tails deviate significantly from the diagonal towards the theoretical quantiles, i. e., the right tail is shorter and the left tail is longer than expected.

In addition, a slight skewness can be observed, supporting the indication from the Q–Q plot. To verify this visual impression, the skewness b_1 of the MFE distributions has been computed for sequence lengths from 20 to 300 nt. All were negative, in a range from -1.47 to -0.075, were the skewness reduces as the sequence length increases. Of course, the truncation of the distribution at 0 kcal mol^{-1} is one obvious source of skewness. This, however, does not explain the observed skewness for longer sequences, where the truncated probability mass is close to zero. Additionally, an attempt has been made to to compensate



(a) Histograms of MFE samples. Vertical lines denote the sample mean.



(b) Mean and variance of sampled MFEs

Figure 12: Histograms of MFE samples and their means and variances for sequences of various lengths. As the sequence length increases, the mean MFE decreases and the variance increases linearly.

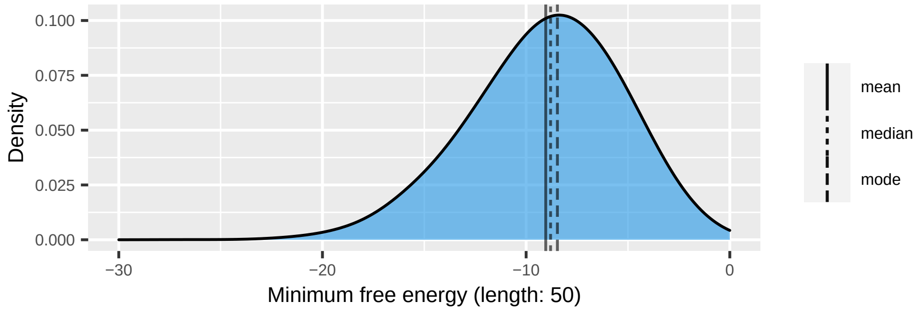


Figure 13: Density of MFEs for sequences of length 50. The distribution is truncated at 0 kcal mol^{-1} and also exhibits a slight skew to the left. The mean and median lie to the left of the mode. The density to the right of the mode is “thicker”, while the left-hand side is “thinner” and longer.

for the effect of the confounding truncation, yielding an *adjusted skewness* value for each sequence length, serving as a lower bound to the true skewness. As Figure 15 demonstrates, this significantly reduces skewness for short sequences, but skewness is still present. Starting from lengths of 70 nt, the regular and adjusted skewness are more and more approaching each other and are almost identical starting at 100 nt. This is congruent with the observed absence of truncation beginning at 70 nt.

Even though the observed skewness values are close to zero especially for longer sequences, they are significantly different from those expected for samples drawn from a normal distribution. The longest sampled sequences (300 nt) show the (absolutely) smallest skewness of -0.0751. For a sample of size 10 000 as in our case, the corresponding p-value is 0.001, which is an order of magnitude below the commonly expected significance threshold of 0.05.

Given the above results, it is clear that though a normal distribution may fit the data to a reasonable degree for many applications, a better fit may be achieved by using another distribution that can model the described properties. Any relevant distribution should be continuous, skewed, and have a support that is either unbounded, or bounded from either above or below. For distributions bounded from below, the sign of the energy values has to be changed to obtain non-negative energies. As cognate distribution families, the normal distribution as a baseline, the truncated normal distribution, the skew normal distribution, the gamma distribution, the generalized extreme value distribution (which generalizes the Weibull, Gumbel, and Fréchet distributions), the Burr (type XII) distribution, and generalized hyperbolic distribution were selected. The observed MFE values were fitted to the distributions using maximum likelihood procedures. Then, the goodness of fit was assessed using two independent criteria. The AIC is based on the likelihood of the observations and additionally accounts for the number of parameters in the model. The CvM statistic considers the quadratic differences of the cumulative distribution function of the fitted distribution to the empirical distribution function of the sample. Together, they provide a complete picture of the quality of the models to the observed data.

The goodness of fit statistics of the different distribution families for different sequence lengths are shown in Figure 16. Generally, the distributions fit the data of very short sequences below 50 nt worse than for longer sequences. The generalized extreme value distribution shows the worst overall fit. The Burr distribution works well in the range of 50–140 nt, but from there is inferior to the family of normal distributions. In the same range, however, the skew normal distribution performs better still, and from 140 nt on it is level with the normal distribution. The truncated normal distribution performs equal to the normal distribution starting at 60 nt, and is better below. Explicitly modelling the cut-off at $0\,\mathrm{kcal\,mol^{-1}}$, it is the best distribution for very short sequences below 40 nt. Since the cut-off is known *a priori*, it has only two free parameters just as the normal distribution, compared to which it performs better. The skew normal distribution does, however, clearly outperforms the truncated normal distribution, though it has one more free parameter. Finally, the generalized hyperbolic distribution shows the best fit of all distributions, except for the shortest sequences of length below 40 nt. However, the difference to the next-best contestant, the skew normal distribution, is not too big, and it has four free parameters. Generally, the more parameters a model has, the more susceptible it is to overfitting. Thus, among multiple models with comparable

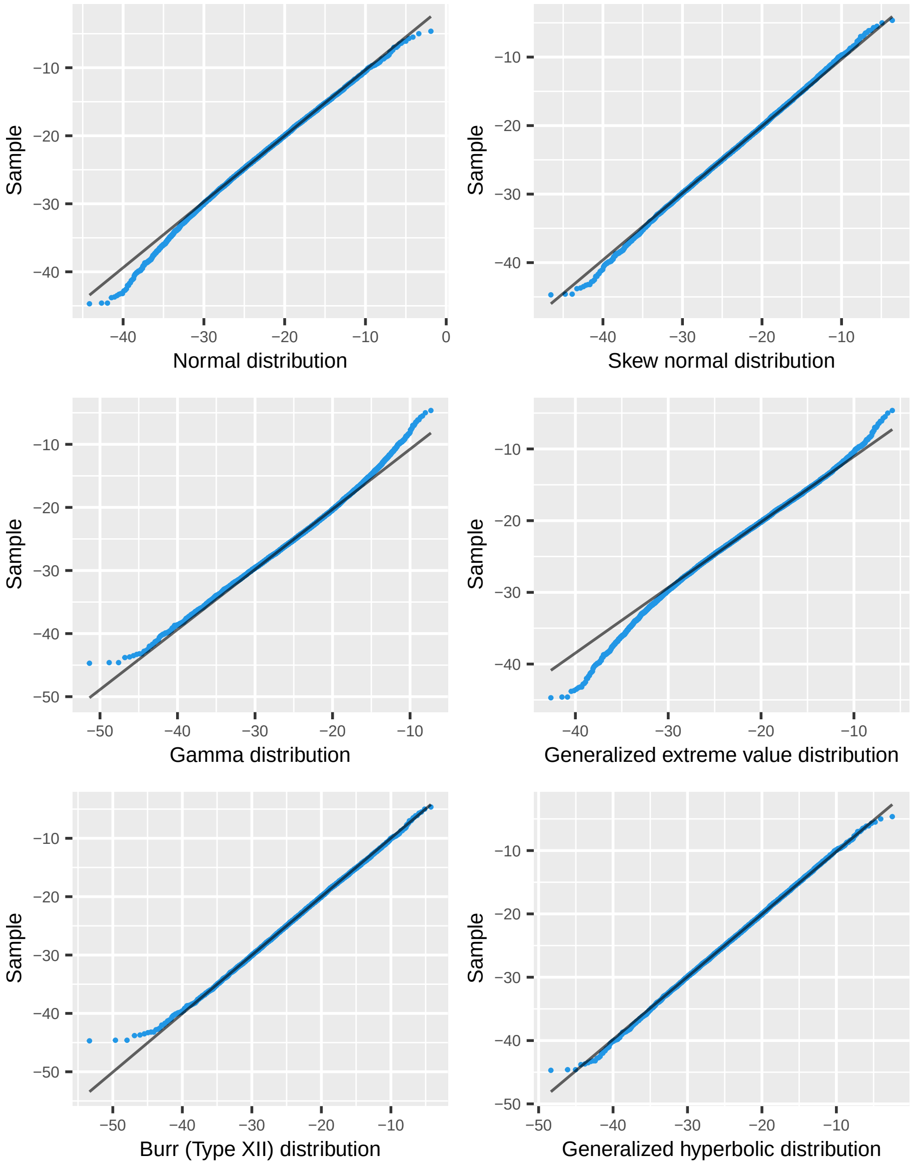


Figure 14: Q–Q plots of the MFE sample for sequences of length 100 against the theoretical quantiles of various fitted distributions. The tails of the normal, gamma and generalized extreme value distributions deviate significantly from the sample quantiles. The skew normal and the Burr distributions fit the data better, and the best fit is obtained when using the generalized hyperbolic distribution.

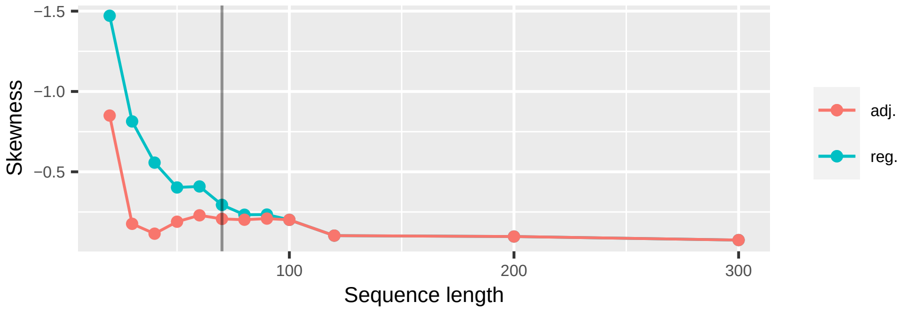


Figure 15: Regular (*reg.*) and adjusted (*adj.*) skewness of MFE distributions, depending on the sequence length. A black, vertical line marks a sequence length of 70 nt, at which the energy distributions no longer exhibit an obvious truncation of their right tails. From there, both skewness measures approach each other closely. All observed skewness values are negative. After a significant drop in the beginning, which can be attributed to the truncation at 0 kcal mol^{-1}, the skewness decreases only very slowly with increasing sequence length.

performance, the model with the lowest number of free parameters is usually the best choice.

4.1.3 Discussion

While the distribution of MFEs of long sequences can be assumed to be approximately normal, this assumption is less precise for shorter sequences, which exhibit a slightly negative skewness. The deviation from the normal distribution occurs mostly in the tails, especially in the left one. If a very tight fit is required, a generalized hyperbolic distribution is a good choice, as it shows the best performance of all candidates. The skew normal distribution is a good alternative, having one free parameter less. For sequences above a length of 140 nt, a normal distribution with two free parameters may be used as well. Only for very short sequences below 40 nt, the truncated normal distribution should be preferred over the other options. One should consider, however, that the precision of MFE folding models in general is limited for very long sequences, thus the limit of 300 nt for the sequences examined here.

From the length of a sequence, a reasonable estimate for its expected MFE can be made. With increasing sequence length, however, the linearly increasing variance widens the MFE distribution, and more and more outliers have to be expected. Also, there are sequences with an MFE of zero for any length.

An aspect that was not considered explicitly in this analysis was the effect of other sequence properties beside the length. To name the most prominent example, the GC content of a sequences, i. e., the fraction of G and C bases w. r. t. all nucleotides, is known to significantly impact the stability (Washietl, Hofacker, and P. F. Stadler, 2005). A higher GC content increases the stability of the sequence, as GC and CG pairs are usually more stable than other base pairs. Also, varying dinucleotide contents have an affect on stability for the same reason. Tools like *RNAz* (Gruber et al., 2010) therefore judge the stability of sequences by comparison to typical values for sequences having similar properties.

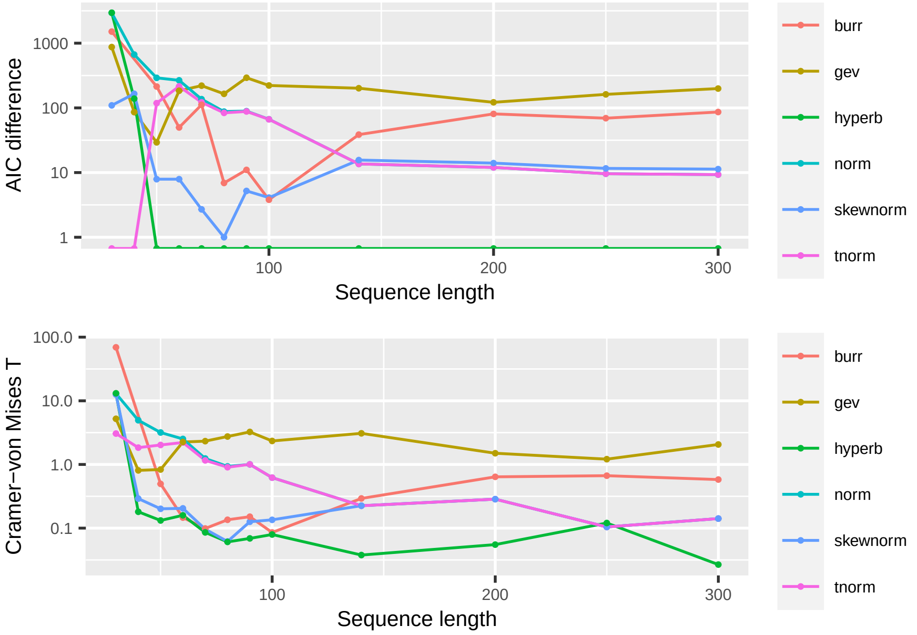


Figure 16: Goodness of fit of various distribution families and sequence lengths (30–300 nt), measured using the difference in AIC (*top*) and the CvM statistic (*bottom*). Both y-axes are logarithmic. For both criteria, "lower is better", i. e., a lower value means the fitted distribution is better-suited to describe the observed energy values. The abbreviated distribution names in the legend stand for the Burr, generalized extreme value, generalized hyperbolic, normal, skew normal, and truncated normal distributions, respectively. The hyperbolic distribution describes the data best; the normal, skew normal, and truncated normal distributions also show a good fit especially for longer sequences.

Taken together, these results provide insights about the statistics of MFEs, which can be used as a guide when sampling random sequences subject to a specific stability constraint. This facilitates an efficient design process providing many suitable candidates for further processing steps.

4.2 The statistics of canonical structures

In Section 3.3, canonical structures were introduced, i. e., RNA secondary structures that do not contain any isolated base pairs. In that section, the concept of coverage was applied to assess the relevance of canonical structures in the structure ensemble of a sequence. Here, we will revisit canonical structures and analyse their frequencies for different sequence lengths. We will also distinguish the subclass of *stable* canonical structures. Furthermore, the analysis will focus on the low-energy part of the structure ensemble, which contains the stable structures most relevant for kinetic simulations.

4.2.1 Methods

Random sequences of lengths of 20–100 nt in steps of 10 nt, and sequences of lengths 100–200 nt in steps of 20 nt have been sampled using a custom Perl script, invoking a Mersenne twister (Matsumoto and Nishimura, 1998) implemented in the module `Math::Random::MT::Auto` for random number generation. For each length, five sequences were selected for further processing. While five sequences per length is a quite small sample size, many different sequence lengths have been tested, and the observed overall trends seem quite stable. For each of these sequences, the energy range containing the lowest 10 000 gradient basins has been determined using an iterative approach: from an initial flooding level of 2 kcal mol^{-1}, the level was increased by 1 kcal mol^{-1} per iteration. In each iteration, all the secondary structures within the current flooding level above the MFE have been computed using *RNAsubopt*. These were partitioned using the coarse graining tool *barriers*, which computes a list of local minima representing the gradient basin macrostates. If at least 10 000 minima were found, the procedure was stopped. The procedure was also stopped if no new minima were found in two iterations to properly handle sequences having less than 10 000 minima.

The determined flooding levels were then used to enumerate all structures within, and for each it was determined whether it was canonical or non-canonical, and in the latter case, also if it was *stable* or *unstable*. For some structure x, a *base pair* $(i, j) \in x$ is referred to as unstable if $\Delta G(x \setminus (i, j)) < \Delta G(x)$, i. e., the free energy of x decreases when removing (i, j), otherwise it is stable. A canonical *structure* is called stable if all of its isolated base pairs are stable. The fractions of structures with these properties were then calculated.

4.2.2 Results

For all sequences consisting of at least 60 nt, more than 10 000 minima were found. The required flooding levels were 8–17 kcal mol^{-1} for sequence lengths 60–100 nt, and 4–9 kcal mol^{-1} for lengths 90–200 nt. This effectively limits the analysis to very stable structures with equilibrium probabilities.

While the overall fraction of non-canonical structures is increasing from about 75% to over 90% for short sequences from a length of 20–40 nt, it is then constantly dropping with increasing sequence length, until reaching about 65% for sequences of length 200 nt, Figure 17. Notably, with the length, the range of the observed values increases: sequences of this length had 44–98% of non-canonical structures. So for longer sequences, depending on the sequence, totally different fractions of lonely pairs have to be expected.

The fraction of unstable and stable non-canonical structures has been computed, where the question of stability of the individual lonely base pairs has been assessed with respect to their free energy contribution in the Turner energy model (Turner and Mathews, 2010) as implemented by the *ViennaRNA* software package. As Figure 17 shows, in many cases, only about half of the non-canonical structures consist solely of unstable base pairs. Only for short sequences of 60 nt or less, most non-canonical structures fall in this class. On the other hand, the number of *stable* non-canonical structures is, with less than 2%, very small for short sequences, but constantly increasing up to about 13%

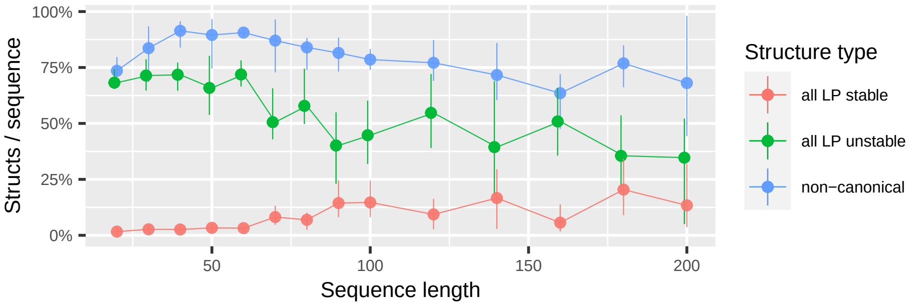


Figure 17: Fractions of different types of non-canonical structures with respect to all enumerated structures. The blue line represents *all* non-canonical structures, whereas the green (red) line shows the fractions of structures that are non-canonical *and* for which all lonely base pairs are unstable (stable). Vertical error bars denote the observed range of values at each point, and the lines run through the means of the observations.

of all structures for sequences of length 200 nt. This corresponds to about 20% of the non-canonical structures.

As expected, most of the stable lonely pairs are CG or GC pairs, which contribute the highest energy bonus of all possible pairs in the Turner energy model. There are, however, also stable lonely AU pairs, and even stable GU wobble pairs can be observed in an isolated position, though their frequency is significantly lower.

4.2.3 Discussion

The concept of canonical structures, i. e., the exclusion of lonely pairs, is an optimization that is supposed to prune the structure ensemble of a given sequence by removing structures which are not occurring in practice. The rationale is that many isolated base pairs are unstable and open up again immediately, and thus need not to be considered (Bompfünewerer et al., 2007). In the light of the results of this section, this assumption is at least debatable. While the restriction to canonical structures does significantly reduce the number of possible structures, only half of the non-canonical structures consist exclusively of unstable base pairs. And even worse, a constantly increasing fraction of the non-canonical structures does not have any unstable lonely pairs at all. Excluding them from a simulation or analysis cannot reasonably be justified with the argument above. Therefore, care has to be taken when applying this heuristic. Since it can massively speed up computations, it may enable the analysis of larger molecules which would otherwise be out of reach with the available resources, but it should not become a default that is used all the time even if it is not necessary.

4.3 The statistics of low-energy minima

As the sequence length increases, the number of structures grows exponentially and quickly becomes inaccessible to analyses relying on explicit construction of the ensemble (Stein and Waterman, 1979). Even when employing sensible heuristics like a restriction to low-energy structures and a gradient-based coarse graining, the huge number of macrostates makes simulations with medium-sized RNA molecules up to 100 nt challenging, and usually their number has to be restricted, too. Therefore the question arises to which extent the energy landscape is still covered when limiting the maximum number of macrostates to a fixed number. Here, 10 000 minima will be considered, as this number still permits an explicit diagonalization of a quadratic transition rate matrix, a technique used, e. g., by the program *Treekin* to analyse the reaction kinetics of the specified system.

4.3.1 Methods

Random sequences of lengths of 20–100 nt in steps of 10 nt (1 000 sequences per length), and sequences of lengths 100–200 nt in steps of 20 nt (400 sequences per length) have been sampled using a custom Perl script, invoking a Mersenne twister (Matsumoto and Nishimura, 1998) implemented in the module `Math::Random::MT::Auto` for random number generation. For each of these sequences, the energy range containing the lowest 10 000 gradient basins has been determined as described in Section 4.2. The energy differences between the global and the 10 000th minimum of each sequence have then been used to conduct the statistical analyses. This has been done with *R*; the package `evd` was used for fitting the generalized extreme value (GEV) distribution, and `ggplot2` has been used to create the visualizations.

The correlation has been computed according to Pearson's definition (Becker et al., 2016), i. e., for two paired samples $(x_1, y_1), \dots, (x_n, y_n)$ of size n, the correlation coefficient was calculated as

$$r_{xy} = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}, \tag{4.1}$$

where $\bar{x}$ and $\bar{y}$ are the respective sample means. The skewness has been computed as in Section 4.1.

4.3.2 Results

For sequences of a fixed length, the distribution of energy ranges of the first 10 000 minima is positive, unimodal and asymmetric. It has a long right-hand tail, and skewness values ranging from 0.3 to 0.9 have been observed for different samples. There is, however, no trend in skewness with respect to the length of the observed sequences. A GEV distribution fits the data very well, Figure 18.

Given the above characteristics, the median and standard deviation seem to be good descriptors for the distribution of energy ranges at a given sequence length. Figure 19 displays their values for lengths from 20 to 200 nt. Its axes are logarithmic, so a straight line indicates an exponential change in value. Starting at 50 nt (40 nt), the median (standard deviation, respectively) decreases exponentially with the sequence length. The observed median energy ranges span values from 13 to 4.7 $\mathrm{kcal\,mol^{-1}}$, with standard deviations from 2.8 to 1 $\mathrm{kcal\,mol^{-1}}$. The initial, exponential increase happens because small sequences simply do not have 10 000 minima; the range computed as described in the methods section is thus small, even though the entire ensemble has been observed.

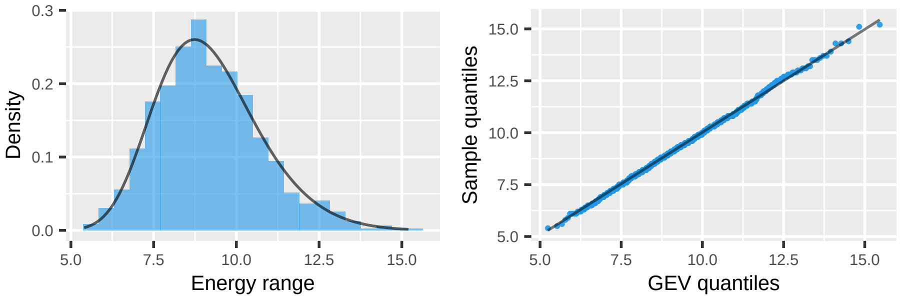


Figure 18: Distribution of energy ranges of the first 10 000 minima for a sequence length of 80 nt. A GEV distribution has been fitted to the data. Both a histogram of the observed values with a curve depicting the fitted density (*left*), and a Q–Q plot of the sample quantiles against the theoretical quantiles of the fitted distribution (*right*) proof that it describes the data very well.

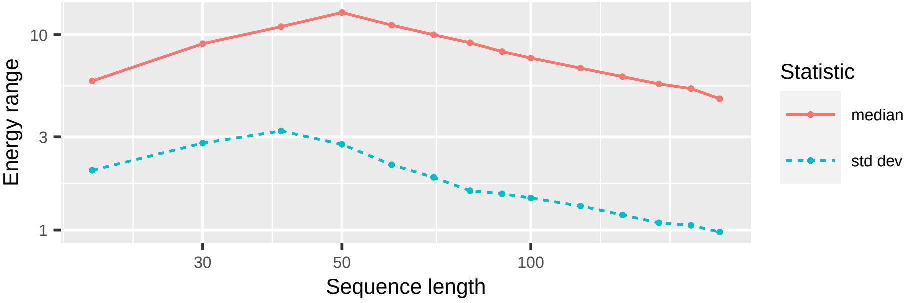


Figure 19: Median and standard deviation of the energy range (in kcal mol^{-1}) covered by the first 10 000 minima for samples of random sequences of various lengths. Both axes are logarithmic. Starting at 50 nt (40 nt), the median (standard deviation, respectively) decreases exponentially with the sequence length. The initial increase happens because small sequences simply do not have 10 000 minima; the computed range is thus small.

As shown before, the expected MFE of a random sequence decreases linearly with its length. Therefore the question arises whether the observed decrease in covered energy range of the first 10 000 minima is directly linked to the decrease of the MFE. However, as Figure 20 demonstrates, the correlation between the MFE and the covered energy range reduces rapidly with increasing sequence length, from a value of -0.86 at 50 nt to only -0.33 at 200 nt.

4.3.3 Discussion

These results indicate that the – well-known – exponential increase of the number of minima w. r. t. the sequence length cannot be circumvented by exploring only the lower parts of the landscape. Especially for longer sequences, the number of minima does not seem to depend so much on their MFE, but mostly on their length. Also, the deviation of the ranges from their expected value becomes smaller and smaller, so for long sequence lengths, the landscapes of all sequences are more or less equally hard to explore.

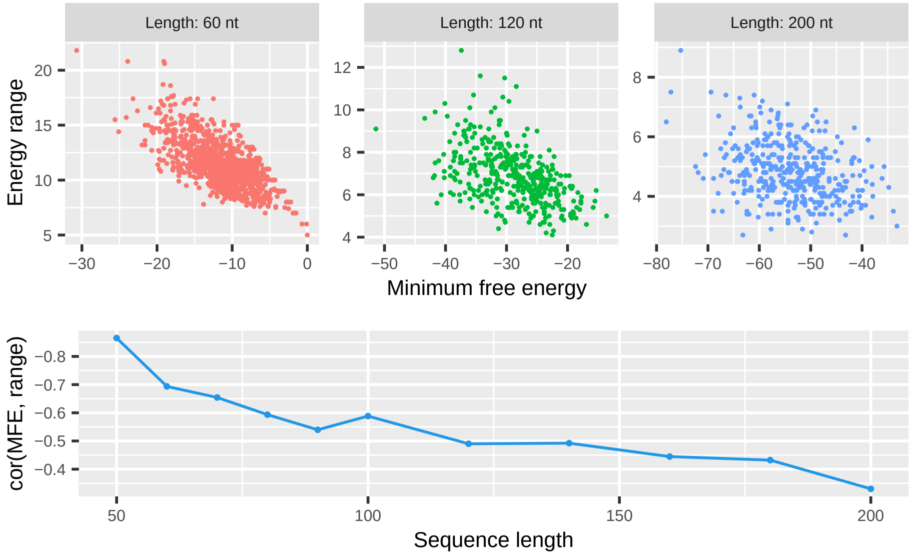


Figure 20: Scatter plots (*top*) and correlation (*bottom*) of MFEs and energy ranges covered by the first 10 000 minima for samples of various sequence lengths. While both properties are strongly negatively correlated for short sequences, the correlation drops rapidly as the sequences get longer.

The consequence is that, for very long sequences, it is not even worth trying to enumerate their energy landscapes even when using a coarse graining approach based on gradient walks. Instead, one has to resort to other methods to perform a kinetic analysis.

4.4 The distribution of free energies within gradient basins

In this chapter, the distribution of *global* MFEs of random sequences as well as the distribution of (low-energy) gradient basins have been analysed. Here, these results are complemented by a study of the distribution of free energies of the structures *within the same basin*. In this context, we will also derive properties of the *local* MFE of the basin, the corresponding structure of which is usually considered as the representative of the basin. These results will serve as a foundation for the results concerning the distribution of Boltzmann weights in gradient basins in the next section.

4.4.1 Methods

The open source program *pourRNA* (Entzian and Raden, 2020) implements a local flooding algorithm for RNA energy landscapes. The source code was modified to allow the output of all enumerated structures within a single basin, and to significantly increase the memory efficiency. The latter was achieved by, firstly, removing a hash data structure storing all encountered structures, which is not necessarily required, and secondly, by applying a space-efficient ternary encoding for dot–bracket strings of secondary structures (“structure packing”) as implemented in the *ViennaRNA* package (Lorenz, Bernhart, et al., 2011). Using the modified version of *pourRNA*, basins of several random RNA sequences have been enumerated. The energy distribution of the structures generated has been analyzed in *R*, the visualizations have been generated using the package `ggplot2`.

The k-th order statistic of an *ordered* sample $x_1 \leq \cdots \leq x_n$ of size n is simply the k-th value x_k. The first order statistic is thus the minimum. Expected values for *normal* order statistics, i. e., order statistics of samples drawn from a (standard) normal distribution, can be obtained naively by numerical integration of the density function, or better by using specialized algorithms (Royston, 1982). There are also effective approximations; here, the heuristic

$$E(X) \approx \Phi^{-1}\left(\frac{1-\alpha}{n-2\alpha+1}\right),$$

as proposed by Blom (1958) was used to obtain the expected first order statistic for a given sample size n . It uses a constant $\alpha = 0.375$ and resorts to the quantile function Φ^{-1} of the standard normal distribution. The rationale is that the lowest value of a sample of size n will, on average, lie close to the $\frac{1}{n+1}$-th quantile of the distribution.

In her course notes, Jenny Baglivo[1] gives an approximation for the variance of (arbitrary) k-th order statistics as

$$\operatorname{Var}(X_{(k)}) \approx \frac{p(1-p)}{(n+2)(f(\theta))^2},$$

where $p = k/(n+1)$, $\theta = F^-1(p)$ is the p-th quantile, and f is the probability density function of the distribution. To obtain the variance of the first order statistic, we set $k = 1$. Of course, an estimate for the standard deviation of the respective order statistic can be obtained by taking the square root of the estimated variance.

4.4.2 Results

The distribution of free energies in gradient basins

As Figure 21 shows, the distribution of free energies within a gradient basin is mostly normal. There is a small deviation especially in the left tail as observed for other energy distributions in this chapter, but the overall fit of a normal distribution is relatively good. Since the choice of the normal distribution will greatly simplify further the analysis of the distribution of structure probability mass in the basin, we will settle with this small compromise here.

The expected minimum free energy of gradient basins

In the previous section, we have shown that the free energies in a basin approximately follows a $\mathcal{N}(\mu, \sigma^2)$ distribution, i. e., a normal distribution with mean μ and variance σ^2. Since the support of the density of the normal distribution is $\mathbb{R}$, there is a non-zero probability of observing arbitrarily small free energies when drawing from this distribution. In practice, however, there is a clearly defined minimum free energy; and also only a finite number of structures n within a

---

[1] Mathematical Statistics, Notebook 4. Boston College, Fall 2017 Course.

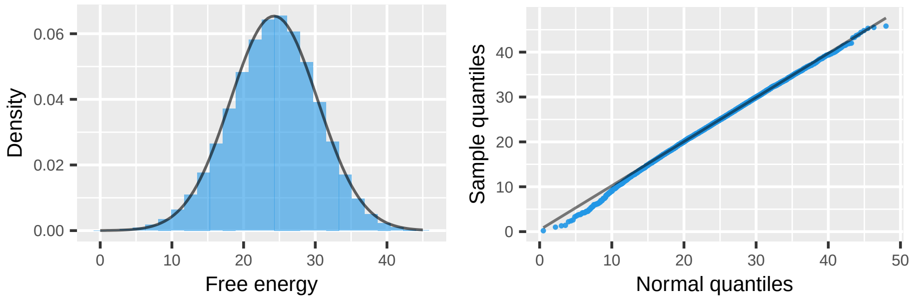


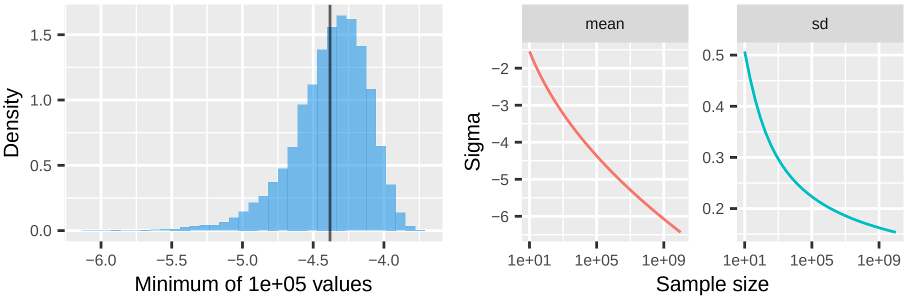
Figure 21: Distribution of free energies of a single gradient basin for a sequence length of 50 nt. A normal distribution has been fitted to the data. Both a histogram of the observed values with a curve depicting the fitted density (*left*), and a Q–Q plot of the sample quantiles against the theoretical quantiles of the fitted distribution (*right*) show a good fit, though especially the left tail deviates slightly from the theoretical quantiles.

basin. Thus, the question arises how exactly the underlying distribution of free energies is related to the basin's MFE.

To shed light on this matter, we change our perspective and consider the set of all structures in the given gradient basin to be a *sample* of size n from the underlying normal distribution. This view reflects the fact that secondary structures represent selected structures from a much larger, three-dimensional space. From this perspective, the local MFE X becomes the minimum of a sample of size n, drawn from a $\mathcal{N}(\mu, \sigma^2)$ distribution. This is, again, a random variable; and one can thus ask for its expected value $E(X)$ and variance $\text{Var}(X)$.

X is also referred to as the *first order statistic* (Becker et al., 2016). Its cumulative distribution function is given by $F_X(x) = 1 - (1 - F_{\mathcal{N}}(x))^n$, i. e., the probability that the smallest of n energies is at most x equals the probability that not all of n drawn energies are greater than x. The probability density function can readily be obtained by differentiating F_X; it is given by $f_X(x) = n f_{\mathcal{N}}(x)(1 - F_{\mathcal{N}}(x))^{n-1}$. As a statistic of a random sample is, again, a random variable, one can also compute expected values and variances of (normal) order statistics. Approaches to do so are described in the methods section.

A distribution of first order statistics as well as plots of the mean and standard deviation for various basin sizes have been computed. The results are given in Figure 22. The distribution of first order statistics is continuous, unimodal and skewed to the left. For basins of of size 100 000, the local minimum is, on average, located -4.4σ to the left of the mean of the underlying normal distribution (Figure 22a). The average MFE and its standard deviation for basin sizes ranging from 10 to 10^9 structures have been analysed (Figure 22b). Basins with 1 000 to 10^6 structures have an average local MFEs of -4σ to -5σ, and very big basins with 10^6 to 10^9 structures exhibit local MFEs as low as -6σ. The standard deviation drops rapidly with increasing structure counts, and is as low as 0.3 even for small basins with only 1 000 structures. With 10^6 structure, standard deviation reduces to 0.2 and approaches 0.15 for huge basins with 10^9 structures.



(a) **(b)**

Figure 22: Distribution and moments of the first order statistic of a standard normal distribution. *(a)* Histogram of minima, obtained from normally distributed samples of size 100 000. The mean is marked with a vertical bar. The distribution is highly skewed to the left. *(b)* Mean and standard deviation (*sd*) of the normal first order statistic for various sample sizes (logarithmic scale), given in multiples of the standard deviation (*sigma*) of the underlying normal distribution.

4.4.3 Discussion

While the mean MFE stays relatively stable over a large range of structure counts, the standard deviation drops quickly to very small values. For a normal distribution, more than 99% of the probability density lie within a range of $\pm 2\sigma$ around the mean. This means that, for example, more than 99% basins of size 10 000 (mean: -4σ, standard deviation: 0.25σ) will have an MFE that is located at $(-4 \pm 0.5)\sigma$, relative to the mean of the underlying basin distribution. As this example illustrates, quite precise predictions about the expected local MFE of a basin can be made using this approach. These insights are also interesting by themselves because they reveal a tight connection of the underlying distribution of a gradient basin with the seemingly independent local MFE and the number of structures in a basin. They may thus be of great value for further analyses.

4.5 The distribution of probability mass in gradient basins

Closely related to the distribution of free energies is the question for the distribution of *probability mass* of the structures within a gradient basin. In the gradient basin abstraction, the distribution of structures within each basin is assumed to be in equilibrium. Furthermore, the equilibrium is assumed to follow a Boltzmann distribution, and thus the probability mass of each structure is given by its Boltzmann weight, Section 2.2.4. Hence, the distribution of probability mass is a (scaled) product of the distribution of energies and the Boltzmann distribution. This distribution is of high interest for the exploration of a basin in a landscape, because it gives valuable information about which structures are influential for the behaviour of the basin in a kinetic simulation.

Depending on the sequence and the selected minimum, a gradient basin may contain thousands or even millions of structures. Their free energies usually span a range of tens of kilocalories. When enumerating such a basin, e. g., to compute transition rates for a kinetic simulation, usually only structures with a high Boltzmann weight (i. e.,, with a low energy) contribute significantly to the result. On the other hand, there are many more structures with a higher

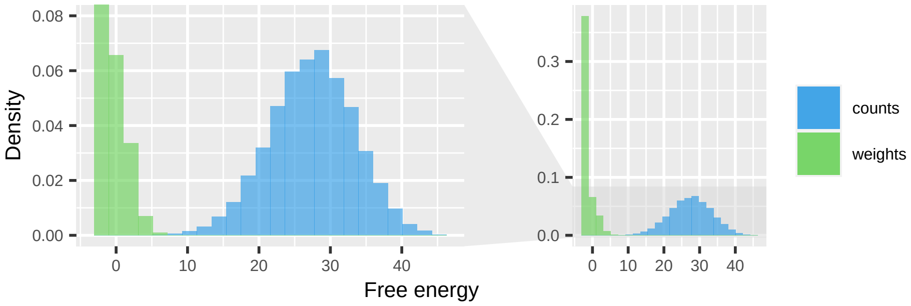


Figure 23: Combined histogram of the counts (*blue*) and the Boltzmann weights (*green*) of the structures of a single gradient basin. The left panel zooms in on the gray area marked in the full data panel on the right-hand side. The selected basin belongs to a sequence of length 50. Almost the entire Boltzmann weight is located closely to the minimum at $-2.4\,\mathrm{kcal\,mol^{-1}}$. Thus, the vast majority of the (blue) structures do not contribute significantly to the partition function.

energy, so it is not clear *a priori* whether their greater number may outweigh their lower individual Boltzmann weight. This section therefore aims to analyze the contribution of structures of a given free energy to the partition function.

4.5.1 Results

Before evaluating the statistics of Boltzmann weights in a basin, a few numerical considerations should be made. The explicit computation of partition functions involves the summation of a great number of Boltzmann weights, and due to the exponential change in their value with respect to their energy, this may lead to numerical instabilities. As a counter measure, a re-scaling with the local MFE as described in Section 2.2.5 should be performed. This ensures that the local minimum has a Boltzmann weight of 1. As computers perform calculations with a limited precision, there is a smallest value ϵ with $1 + \epsilon > 1$ that can be stored exactly by a program. ϵ is also commonly referred to as machine epsilon, e. g., in the `float.h` header of the *C* programming language. Thus, there is also a maximum energy ΔG_{max} such that $1 + \mathrm{Z}[\Delta G_{\mathrm{max}}] > 1$, where $\mathrm{Z}[\Delta G_{\mathrm{max}}]$ is the Boltzmann weight of energy ΔG_{max}. When computing the partition function with double precision according to the "IEEE Standard for Floating-Point Arithmetic" (2019), $\epsilon = 2^{-53}$, and consequently $\Delta G_{\mathrm{max}} \approx 22.64\,\mathrm{kcal\,mol^{-1}}$. Therefore, $22.7\,\mathrm{kcal\,mol^{-1}}$ may serve as a – very coarse – upper bound for the flooding of any basin.

Using *pourRNA*, the structures of several gradient basins have been enumerated and analysed. In Figure 23, both the number of structures and their (accumulated) Boltzmann weight is shown for the range of energies within a single gradient basin. The green bars mark energies with significant contributions to the partition function. This is only a tiny fraction of the full energy range; only the left-most tail of the distribution of structures is relevant. The results are similar for other basins (data not shown).

The presented results indicate that the exponential decrease in Boltzmann weight with increasing free energies within a gradient basin dominates the increase in the number of structures in the higher-energy range. Only a small, low-energy tail of the distribution of free energies is required to be enumerated to cover the partition function to a high degree. In the next section, this observation is put on a solid theoretical basis.

Theoretical distribution of probability mass in gradient basins

In Section 4.4, it was shown that the free energies within a gradient basin are approximately normally distributed. Thus, if X is the free energy of a randomly chosen secondary structure of some gradient basin, then $X \sim \mathcal{N}(\mu, \sigma^2)$, i. e., X is a random variable following a normal distribution with expected value $E(X) = \mu$ and variance $\mathrm{Var}(X) = \sigma^2$. Its probability density function is thus

$$f_{\mathcal{N}}(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right). \tag{4.2}$$

It was also mentioned that structures *within* a gradient basin are assumed to be equilibrated (cf. Section 2.3.5) and their probabilities thus follow a Boltzmann distribution. To model the distribution of the *Boltzmann weight* in a basin, one thus needs to multiply the density of the energies with their respective Boltzmann weight $\mathrm{Z}[x] = \exp(-\beta x)$, where $\beta = (RT)^{-1}$ is the inverse temperature. By doing so, we obtain the following result.

Theorem 1. *Let Y be the energy of a secondary structure from a gradient basin with $\mathcal{N}(\mu, \sigma^2)$-distributed free energies, chosen randomly according to their equilibrium distribution. Then*

$$Y \sim \mathcal{N}\left(\mu - \beta\sigma^2, \sigma^2\right).$$

Proof. We construct the cognate density f_Y of Y by multiplying the density of the normal distribution (Equation 4.2) with $\mathrm{Z}[x]$, and then show that it is indeed a proper probability density function:

$$\begin{aligned} f_Y(x) &= c \cdot \mathrm{Z}[x - c_{\Delta G}] \cdot f_{\mathcal{N}}(x) \\ &= c \exp\left(\beta(c_{\Delta G} - x)\right) \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right) \end{aligned} \tag{4.3}$$

for some normalization constant c and an energy scaling constant $c_{\Delta G}$. Obviously, the product of two strictly positive functions is positive. Next, we consider the indefinite integral of f_Y:

$$\int f_Y(x)dx = \frac{c}{2} \exp\left(\beta(\frac{\beta\sigma^2}{2} - \mu + c_{\Delta G})\right) \operatorname{erf}\left(\frac{x - (\mu - \beta\sigma^2)}{\sigma\sqrt{2}}\right)$$

using the Gauß error function $\operatorname{erf}\colon \mathbb{C} \longrightarrow (-1, 1)$. Since $\lim_{x\to\pm\infty} \operatorname{erf}(x) = \pm 1$, we can easily calculate the value of the improper integral

$$\int_{-\infty}^{\infty} f_Y(x)dx = c \exp\left(\beta(\frac{\beta\sigma^2}{2} - \mu + c_{\Delta G})\right) \overset{!}{=} 1,$$

which converges to a constant. f_Y is thus a proper probability density. By solving for c, inserting it into Equation 4.3 and rearranging terms, we finally obtain

$$f_Y(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left(\frac{x - (\mu - \beta\sigma^2)}{\sigma}\right)^2\right),$$

i. e., the density of Y is normal with a mean of $\mu - \beta\sigma^2$ and a variance of σ^2. □

As a direct consequence of the theorem, the cumulative distribution function of Y can be written as

$$F_Y(x) = \Pr(Y \leq x) = \int_{-\infty}^{x} f_Y(u)\,du = \frac{1}{2} + \frac{1}{2}\operatorname{erf}\left(\frac{x - (\mu - \beta\sigma^2)}{\sigma\sqrt{2}}\right).$$

The probability $\Pr(Y \leq x)$ is exactly the probability that a structure from the equilibrated gradient basin has an energy of at most x. It can thus be interpreted as the fraction of the partition function covered when enumerating the basin up to an energy value of x.

Also, note that the energy scaling constant $c_{\Delta G}$ was cancelled and is not part of the final form of f_Y. Consequently, the partition function can be rescaled freely without changing the distribution of Boltzmann weight within a basin.

Predicting the partition function of a gradient basin

Theorem 1 has further implications: it tells us that the mean of the weight distribution – which is equal to the median and mode in the case of normal distributions – lies $\beta\sigma^2$ to the left of the mean μ of the underlying distribution of the free energies. For physiological temperatures from 0 to 50 °C, β is in the range of 1.842 to 1.557 mol kcal^{-1}, and at 37 °C, $\beta \approx 1.623$ mol kcal^{-1}. From Figure 22, we recall that basins with at most 10^9 (i. e., all that are feasible to explore) have an expected MFE not lower than $\mu - 6\sigma$. Thus, for all basins with $\sigma > 6/1.557 \approx 3.85$, the mode of the Boltzmann weight distribution is smaller than the expected MFE of the basin, since in that case $-\beta\sigma^2 < -6\sigma$. This assumption holds for the most basins of non-trivial sizes even for short sequences of 30 nt; for sequences of 50 nt, basins with 10^6 to 10^7 usually have a σ of 6 to 6.5 (data not shown).

It follows that the bigger part of the (full) Boltzmann weight density f_Y is actually located outside of the basin. Since the partition function sums over the secondary structures of the basin, the part of f_Y that is left of the basin MFE has to be truncated, and only the right tail has to be kept to properly model the partition function. The density of the truncated distribution can be calculated as

$$f_Y^{\geq\theta}(x) = \begin{cases} \frac{f_Y(x)}{1-F_Y(\theta)} & x \geq \theta, \\ 0 & x < \theta \end{cases}$$

for a cut-off point θ by rescaling the truncated density to 1. The truncated cumulative distribution function is obtained similarly as

$$F_Y^{\geq\theta}(x) = \begin{cases} \frac{F_Y(x)-F_Y(\theta)}{1-F_Y(\theta)} & x \geq \theta, \\ 0 & x < \theta. \end{cases}$$

An example, where this approach has been applied to predict the coverage of a basin with about 5 million structures with a MFE of -7.3 kcal mol^{-1}, a mean energy of 24.3 kcal mol^{-1} and a standard deviation of 6.1 kcal mol^{-1}, is presented in Figure 24. There, the value of $F_Y^{\geq\theta}(x)$ is compared to the first 40 *partial partition functions* Z_k, $k = 1, \dots, 40$, where $\mathrm{Z}_k := \mathrm{Z}[\{x_1, \dots, x_k\}]$ is the partition function of the k lowest energies $x_1 \leq \dots \leq x_k$ of the basin. Starting at $k = 15$, the values of Z_k and $F_Y^{\geq\theta}(x_k)$ differ by less than 1%.

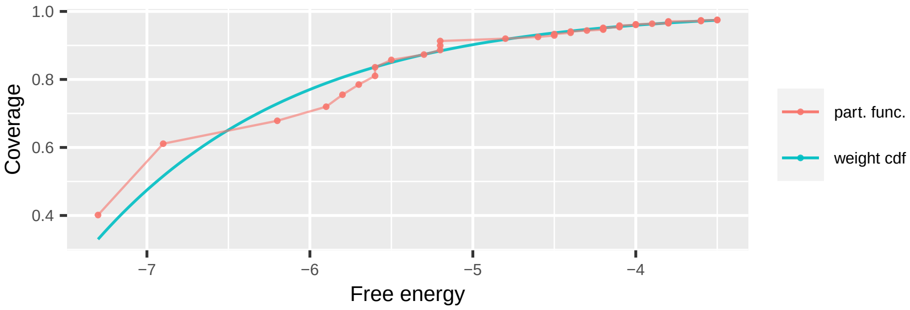


Figure 24: Comparison of the coverage predicted using the cumulative distribution function $F_Y^{\geq\theta}(x)$ of the Boltzmann weight distribution (*weight cdf*) with the values obtained by summing up all structures' Boltzmann weight up to the given energy level (*part. func.*). The curves converge against each other starting at the 15th structure at an energy of $-4.8\,\mathrm{kcal\,mol^{-1}}$. θ was chosen to be $0.4\,\mathrm{kcal\,mol^{-1}}$ smaller than the basin's MFE.

A remaining difficulty is the choice of the cut-off point θ. Choosing the basin's MFE x_1 leads to a slight underestimate of the coverage since $F_Y^{\geq x_1}(x_1) = 0 < \mathrm{Z}_1$. In the above example, $\theta = x_1 - (x_2 - x_1)$ was chosen to compensate for this effect, i. e., θ was shifted to the left by the distance of x_1 to the second-smallest energy x_2, corresponding to $0.4\,\mathrm{kcal\,mol^{-1}}$ in this specific case. But even when $\theta = x_1$, the values of $F_Y^{\geq x_1}(x_k)$ and Z_k converge quickly, allowing to use this choice to obtain a lower bound on the coverage of the partial partition functions.

4.5.2 Discussion

As mentioned in the beginning of this section, the distribution of probability mass within gradient basins is of high relevance for the efficient exploration of energy landscapes. The reason is that the transition rates between the macrostates are usually defined in terms of weighted sums over the contained structures, where the weights are the equilibrium probabilities of the structures in their respective basin, which in turn is determined by its Boltzmann weight and the partition function of the basin (Wolfinger et al., 2004). If the distribution of Boltzmann weights in the basin was known *a priori*, an informed choice of an energy threshold for energy landscape exploration by the means of local flooding could be made, allowing to significantly reduce the number of enumerated structures with a precise knowledge of the error introduced.

It has been shown that the structures contributing significantly to the partition function are, as expected, located at the low-energy part of the basin. This is similar to the behavior of the entire structure ensemble of the sequence, where a small fraction of low-energy structures as computed using Wuchty's algorithm (Wuchty et al., 1999) accounts for almost the entire partition function. It is still interesting to see how tiny the relevant fraction of structures really is, and that this behaviour is universal to gradient basins, even when their local minima lie high above the global MFE. As a consequence, the exploration of huge basins would remain feasible for much longer sequences when restricting enumeration of structures to those with a relevant contribution to the partition function. To estimate the observed fraction of the partition function while flooding up to a specific energy threshold, however, required a deeper understanding of the underlying distribution of the Boltzmann weight.

This goal was achieved by Theorem 1. It provides a powerful tool to estimate the coverage of a basin when enumerating it up to arbitrary energy values once the parameters of the underlying distribution of free energies are known. This allows to efficiently enumerate a specific quantile of the partition function, skipping over the vast majority of statistically unimportant structure and thus dramatically improving the performance of local flooding procedures.

The last remaining obstacle that prevents the application of the presented results in practice is that μ and σ are unknown. Thus, to use the proposed methods, both parameters need to be estimated first. Of course, enumerating all structures in a basin is impracticable for huge basins, and for small basins the partition function can be computed explicitly. One possible approach would be to sample structures from the basin first to estimate μ and σ, and then to use them to predict the partition function. It is, however, non-trivial to obtain unbiased samples from a gradient basin, as usually only the local MFE structure is known *a priori*, and other structures are generated iteratively from it by explicitly constructing its neighbor structures. The structures generated by this approach are thus mostly located in the energetically lower part of the basin, too. Designing effective strategies to solve this problem will be an interesting field for future work.

4.6 Conclusion

Though the energetic properties of RNA sequences and structures vary greatly, they often follow specific patterns and rules. These have been studied in this chapter, and the results provide deep insights that lay the foundation for solving more specific tasks like the design of RNA sequences or the kinetic analysis of a given molecule efficiently. The distribution of MFEs of random sequences has been analyzed and shown to be mostly, but not entirely, normal. Linear models to predict the mean and variance of MFEs for a given sequence length have been given, which can guide the choice of the right number of nucleotides for a design problem when a specific stability is required. The other sections focus on various aspects relevant in the context of RNA energy landscapes and folding kinetics. The effect of the restriction to canonical structures was analyzed, and was shown to be a possible source of error especially for longer sequences, for which many non-canonical structures actually consist exclusively of stable base pairs. The energy range of a coarse-grained landscape that is accessible when restricting the number of gradient basins to a fixed number was analysed. Both its median and standard deviation were shown to decrease exponentially with increasing sequence length, making computations of folding kinetics for longer sequences a hard task, and showing that, at high lengths, almost all sequences are equally hard to analyze. It was shown that the free energies of the structures within a single gradient basin are very well described by a normal distribution. From this normal distribution and the number of structures, expected values and standard deviations for the basin's MFE can be derived. Finally, the distribution of Boltzmann weights within a gradient basin has been analyzed. The probability mass has been shown to be located exclusively in the leftmost tail of the distribution of free energies. Theorem 1 formally describes this distribution as similar to that of the underlying distribution of free energies, but shifted to left by $\beta\sigma^2$, where β is the inverse temperature.

CHAPTER 5

The Design of Artificial Cotranscriptional Riboswitches

Contents

5.1 Biological background **80**
  5.1.1 Structure and function of riboswitches 80
  5.1.2 Classes of riboswitches 82
  5.1.3 Identifying aptamers for novel ligands 83
  5.1.4 Measuring riboswitch activity 83
5.2 Design and analysis of transcriptional neomycin-dependent riboswitches **86**
5.3 Materials and methods **86**
  5.3.1 Biochemical experiments 86
  5.3.2 Probabilities of RNA conformations 86
  5.3.3 Simulation of cotranscriptional folding 87
  5.3.4 Design of transcriptional riboswitches 88
  5.3.5 Designing a decoupling leader sequence 89
5.4 Results **89**
  5.4.1 Design of artificial neomycin riboswitches 90
  5.4.2 Riboswitch N1M7-D requires a 5′ hairpin structure 92
  5.4.3 Destabilizing the leader hairpin decreases reporter gene expression 93
  5.4.4 Upstream sequence context may impair riboswitch function 94
5.5 Discussion **95**

This chapter describes the design, optimization, and experimental validation of transcriptional riboswitches. Though the work is centered on constructing riboswitches responding to the antibiotic neomycin, the described techniques are universal and can be applied to any small molecule that could potentially bind to a specific part of an RNA molecule.

This chapter is based on the following literature:

C. Günzel*, F. Kühnl*, K. Arnold, S. Findeiß, C. E. Weinberg, P. F. Stadler, and M. Mörl (2020). "Beyond Plug and Pray: Context Sensitivity and in silico Design of Artificial Neomycin Riboswitches". In: *RNA Biology*, pp. 1–11. DOI: `10.1080/15476286.2020.1816336`.

It will not be cited individually in the text.

5.1 Biological background

Riboswitches are small *cis*-regulatory sequences located in the 5′-UTR of some protein-coding genes, predominantly in prokaryotes (Nahvi et al., 2002; Serganov and Nudler, 2013). They control the expression of the gene located downstream in response to the presence of a specific small molecule called the *ligand*. It may be of extra- or intracellular origin and may belong to any of such different groups of molecules such as nucleobases, amino acids, antibiotics, metal ions and many more (Montange and Batey, 2008; Wallis et al., 1995). This allows *in principle* to construct orthogonal switches that, for instance, force a gene to respond to any cell-permeable and non-toxic substance (Etzel and Mörl, 2017) and thus to create functional biosensors that respond to both intracellular signals and environmental conditions. The relatively small size of riboswitches facilitates their design and optimization (Fowler, Brown, and Li, 2008; Weigand, Sanchez, et al., 2008), which makes them attractive for numerous applications in synthetic biology. The practicality of such an approach, however, finally depends on how easily engineered riboswitches can be embedded in an arbitrary sequence context without resorting to a labor-intensive trial-and-error procedure to ensure their functionality.

In this section, the common properties of riboswitches will be revisited to introduce the reader to this matter to a degree necessary to understand the following steps to design and validate artificial riboswitches. There is also an enormous variety of naturally occurring riboswitches (Roth and Breaker, 2009) in most living organisms, but their description is beyond the scope of this work.

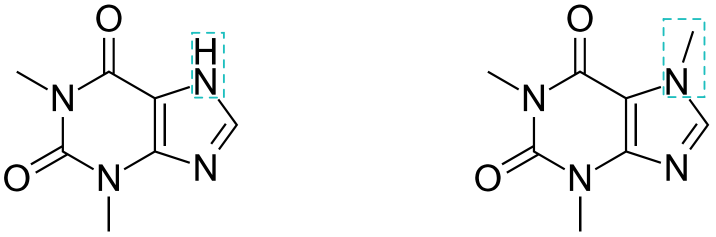
5.1.1 Structure and function of riboswitches

Riboswitches consist of two overlapping sequence parts, which are referred to as the *sensor* and the *actuator* domain, respectively (Findeiß et al., 2015). The sensor is capable to specifically recognize and bind the ligand molecule. The actuator domain mediates the regulatory effect of the riboswitch depending on the state – bound or unbound – of the sensor domain.

The sensor domain is usually a so-called *aptamer*, i. e., a defined sequence that a given ligand can bind to. To do so, the aptamer folds into a specific,

---

*The authors share first authorship.



(a) Theophylline **(b)** Caffeine

Figure 25: Skeletal formulas of theophylline and caffeine. Both molecules are identical up to the replacement of a hydrogen by an additional methyl group (*blue, dashed box*) bound to the seventh nitrogen atom (N) of caffeine.

Adapted from:
`https://en.wikipedia.org/wiki/File:Theophylline.svg`
`https://en.wikipedia.org/wiki/File:Caffeine_structure.svg`
on 2021/07/23.

binding-competent structure, which often exhibits a pocket-like shape and wraps around the ligand molecule upon binding (cf. (Jenison et al., 1994; Jucker et al., 2003)). Some aptamers are capable to recognize their ligand with extraordinarily high affinity and specificity. For example, a theophylline aptamer reported by Jenison et al. (1994) has a dissociation constant of only 100 nM, but is still capable to distinguish its actual ligand from the similar metabolite caffeine, which only differs by an additional methyl group attached to the seventh nitrogen atom, cf. Figure 25.

The *actuator* domain can take many different forms and determines the mechanism of action of that specific riboswitch. *Transcriptional* riboswitches regulate gene expression by stopping the transcription process. Specifically, intrinsic (ρ-independent) termination is triggered as the nascent, transcribed actuator forms a hairpin structure by intra-molecular base pairing, which is then interacting with the proximal RNAP. Additionally, a uridine-rich region follows downstream of the hairpin structure. This so called *poly-U stretch* is known to stall RNAP and thus constitutes a transcription pause site (Gusarov and Nudler, 1999; Peters, Vangeloff, and Landick, 2011). In conjunction, these two sequence elements force RNAP to release the DNA template strand (Wilson and Hippel, 1995). *Translational* riboswitches, in contrast, modulate the accessibility of the RBS located in the expression platform of the switch such that the ribosome binds the messenger RNA (mRNA) transcript with a higher or lower affinity, thus regulating the expression of the gene. The modulation of the accessibility is, again, mediated by intramolecular base pairs within the transcript, e. g., by forming a roadblock close to or including the RBS (Picard et al., 2009). Yet another type of actuator domains are *self-cleaving ribozymes* like the hammerhead ribozyme (Wieland and Hartig, 2006), i. e., sequences that mediate the decay of their own transcript by self-cleavage. Eukaryotic riboswitches fall into yet another class as they mostly control their regulated gene by altering the splicing process (Wachter, 2010).

Importantly, the sensor and the actuator domains are overlapping in a way that the binding of the ligand to the sensor enables or disables the actuator and thereby couples the regulatory effect on expression to the ligand concentration. Mechanistically, this happens because, firstly, the binding of the ligand stabilizes the aptamer's secondary structure, and secondly, each nucleobase can only engage in a regular Watson–Crick base pair with at most one other nucleobase. Thus, the overlap of both domains leads to a competition between mutually exclusive structural conformations, and the binding of the ligand shifts their balance depending on its stabilizing effect and its concentration (Breaker, 2012). The stability granted to the aptamer's structure by binding the ligand can be quantified by assigning it a free energy value. This value can be determined experimentally by procedures such as isothermal titration calorimetry (Jones, Piszczek, and Ferré-D'Amaré, 2019).

5.1.2 Classes of riboswitches

Since the term "riboswitch" is used for many different regulatory elements, it is often useful to classify them by their properties. There are various approaches to do so:

i) by origin: *natural* or *synthetic*. While natural riboswitches have evolved in living organism, synthetic riboswitches were designed and synthesized in a laboratory.

ii) by taxon: *eukaryotic*, *prokaryotic*, *bacterial* etc. Here, we are mostly concerned with bacterial riboswitches.

iii) by ligand: riboswitches responding to the same ligand(s) are considered to belong to one class. For example, there are entire families of riboswitches responding to the metabolite S-adenosyl-L-methionine (SAM), making it the most abundant ligand class of all natural riboswitches (Price, Grigg, and Ke, 2014).

iv) by response: *on* or *off*. With *on* riboswitches, the expression of the regulated gene is *increased* in the presence of the ligand, and *reduced* in its absence. The opposite is the case for *off* riboswitches. Riboswitches responding to multiple ligands may model any of various logical functions such as AND and NAND (e. g. Sharma, Nomura, and Yokobayashi, 2008) or (negated) implications (e. g. Ausländer et al., 2014).

v) by mechanism: *transcriptional* or *translational*, ribozymes, switches altering the splicing process

Depending on the context, each way of classification is useful, and all of them are used in this work.

A related extension of regular riboswitches are the so-called *tandem riboswitches*. They consist of several independent, transcriptional riboswitches that are arranged sequentially within a single $5'$ UTR. Since the transcription is inhibited if and only if any of the chained riboswitches is inhibiting it, a tandem riboswitch acts like a logical conjunction (i. e., an AND gate) for *on* switches, and like a NOR gate for *off* switches, on the response of the individual switches. Examples include a natural tandem riboswitch from *Bacillus clausii* (*B. clausii*) composed of two individual *off* riboswitches responding to SAM and coenzyme B_{12}, respectively, (Sudarsan et al., 2006) as well as a synthetic tandem riboswitch created by combining a theophylline riboswitch and a tetracycline riboswitch in *E. coli* (Domin et al., 2017). Note that tandem riboswitches differ from multi-ligand *AND* riboswitches in that they are composed of consecutive but independent single-ligand riboswitches whereas, in the multi-ligand switch, the domains of the riboswitch are interleaved and interact with each other (e. g. Mandal et al., 2004; Sharma, Nomura, and Yokobayashi, 2008).

5.1.3 Identifying aptamers for novel ligands

While many substances have a natural binding affinity for RNA, finding an RNA sequence capable of specifically identifying and tightly binding a given ligand *in silico* is challenging . A promising *experimental* procedure, however, is systematic evolution of ligands by exponential enrichment (SELEX) (Ellington and Szostak, 1990; Tuerk and Gold, 1990). It is an iterative *in vitro* procedure that determines aptamer sequences for a given ligand by starting with a pool of random RNA molecules, subsequently removing unsuitable candidates from it, and then amplifying the remaining ones. This procedure is repeated until only high-affinity aptamers remain. The initial RNA pool contains a vast number of unbiased random oligonucleotides of a length of about 100 nt. The removal of unsuitable candidates is then achieved by applying an *affinity chromatography*: the ligand is fixated onto a solid medium (the affinity column), and the solution containing the RNA pool is added. A buffer is then used to wash off unsuitable candidates while binding-competent RNAs remain bound to their ligand and thus on the column. Then, an elution buffer is used to dissociate and recover the RNAs from their ligands. A reverse transcriptase is used to obtain complementary DNA strands of the remaining aptamer candidates, which can then be amplified using several cycles of polymerase chain reaction (PCR). Adding an RNAP, the amplified DNA is transcribed into RNA again, which can be used for another cycle of the procedure.

Wallis et al. (1995) applied the described procedure to the antibiotic neomycin. Building on these results, Weigand, Sanchez, et al. (2008) further analyzed the identified aptamer N1 with varying stem sequences, and demonstrated its ability to act as a translational roadblock if neomycin is bound. Using the shortest stem M7, we selected the aptamer N1M7 as foundation for our designs of transcriptional riboswitches. M7 consists of five base pairs and comprises the P1 helix of the aptamer. It is followed by an interior loop, which acts as the binding pocket for a single neomycin molecule. From that loop region, another hairpin loop branches off, forming the P2 helix of the 27 nt long aptamer (Figure 26). N1M7 has a dissociation constant of $K_d = (9.2 \pm 1.3)\,\mathrm{nM}$, corresponding to a stabilizing Gibbs free energy contribution of $\Delta G = (-11.4 \pm 0.1)\,\mathrm{kcal\,mol^{-1}}$ (Weigand, Schmidtke, et al., 2011).

5.1.4 Measuring riboswitch activity

Due to the enormous complexity of living organisms, an error-free prediction of the exact behaviour of an arbitrary synthetic biomolecule is impossible. Therefore, it was inevitable to experimentally validate the riboswitch candidates we designed here to be able to make reliable claims about their functionality. There are several ways to do so, and the advantages and disadvantages of these choices will be discussed in this section.

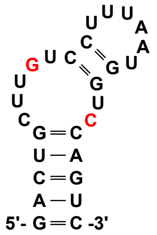


Figure 26: Sequence and secondary structure of the aptamer N1M7 (Weigand, Sanchez, et al., 2008). Note that the MFE structure predicted by the *ViennaRNA* package contains an additional GC pair at position 9 and 22 (colored red, energy difference $\Delta\Delta G = -0.7\,\mathrm{kcal\,mol^{-1}}$).

On *in vivo* and *in vitro* experiments

When experimenting with biomolecules in a laboratory, there are two fundamentally different approaches to do so. The first one is to observe the molecule in the living cell, which is referred to as an *"in vivo"* experiment, while the second one is to analyze the molecule in an artificial buffer solution, which is commonly called an *"in vitro"* setting. While *in vitro* analyses exclude the effects of known and unknown confounders and thus allow a more precise study of a specific aspect of the system, only an *in vivo* experiment draws a realistic image of the actual behaviour of the molecule in the cell. In this study, we opted for the latter option, as explained in the following section. Therefore, the presented results are not only of theoretical interest, but can be readily applied in practice, which stresses their high relevance.

A suitable host organism

To test the constructed riboswitch candidates *in vivo*, a suitable host organism is required. Of course, *transcriptional* riboswitches are only found in prokaryotes (Wachter, 2010), where the DNA is located directly in the cytoplasm and, thus, the nascent mRNA transcript can immediately be accessed by RNAP. We thus opted to use *E. coli*, a gram-negative bacterium and one of the most widely used model organisms of all prokaryotes, as a host for our constructs.

The ligand neomycin, which our aptamer is sensitive to, is as antibiotic, i. e., it is toxic to many bacteria including *E. coli* (Waksman and Lechevalier, 1949). Its mechanism of action is the binding to the 30S subunit of the bacterial ribosome, interfering with translation and thus stopping the synthesis of proteins in the cell, which ultimately leads to its death (Mehta and Champney, 2003). Given the fact that the ribosome consists of about 65% ribosomal RNA (rRNA) (Kurland, 1960), this explains the potential of neomycin to act as a riboswitch ligand: it has a high affinity to bind RNA.

The obvious drawback, however, is that a neomycin-resistant strain of *E. coli* is required to test the synthetic riboswitch candidates. Specifically, we used the *E. coli* SQ171 containing the plasmids ptRNA67 and pKK3535 (A1408G), derived from *E. coli* MG1655 (Quan et al., 2015), which was kindly provided by Kurt Fredrick, Columbus, OH. The resistance is achieved by the deletion of some of the seven *rrn* operons, which encode for the rRNA that neomycin binds to. While these modification have only a moderate impact on cell proliferation, they do not influence the neomycin concentration within the cell, as would be the case with a resistance mediated by an increased neomycin efflux.

Measuring on RNA or on protein level

For *translational* riboswitches, the natural method of analyzing their activity is to measure the amount of protein translated from the gene downstream of the riboswitch. Since the regulatory function of bacterial riboswitches is based on a local interaction with the polymerase or the ribosome, the regulated gene can be exchanged more or less freely without influencing the behaviour of the switch itself. Therefore, one can choose from many well-established reporter genes and methods of measurement to conduct the analysis. Examples of popular reporters include the enhanced green fluorescent protein (eGFP) (Green et al., 2014) or the yellow fluorescence protein (YFP) (Chen et al., 2013), and their emitted luminosity can be measured using image or flow cytometry. Another popular method is the ONPG test, which uses β-galactosidase as a reporter (Smale, 2010).

For *transcriptional* riboswitches, one can apply the same approach as for translational riboswitches. The measurement will, however, be more indirect, because the switch regulates the amount of mRNA available to the ribosome, and not the translation directly. On the other hand, when the aim is to assess the actual outcome of the regulatory effect on the entire cell, this is still the most reliable option. This is why this approach was chosen as the main method of measurement in our work.

Additionally, the activity of transcriptional riboswitches can be measured directly on RNA level. A classical method to proof the presence of a specific RNA is the northern blot (Trayhurn, 1996). It requires extracting the RNA from the cell, denaturing it and blotting it to a membrane after determining its size by applying a gel electrophoresis. Our collaboration partners used it in this work to verify that our riboswitch candidates are in fact regulating the amount of RNA in the cell, and not just the level of synthesized protein. A more direct approach is using so-called light-up aptamers (Ouellet, 2016). These systems consist of a specific RNA aptamer, often folded into a G-quadruplex structure, and a ligand binding to it. Once the ligand, which is also referred to as fluorogen, binds to its aptamer, it starts emitting light of a specific color. Thus, a light-up aptamer gives direct visual feedback about the amount of transcribed RNA it is located on. This would not only be easier, faster, and cheaper than a northern blot, but also allows to easily measure reporter gene expression, too. At the time of the experiments with our synthetic riboswitches, the available light-up aptamer systems were not yet bright enough for reliable *in vivo* detection, but this might change soon. Determining both protein and RNA levels of the same sample would yield valuable information about the designed candidates and may thus become an interesting option for future experiments.

Measuring riboswitch activity using fluorescence measurements

To measure fluorescence intensity to verify riboswitch activity, the riboswitch candidate is cloned into the 5′ UTR of a reporter gene encoding for a protein exhibiting fluorescent activity, e. g., eGFP. The resulting gene cassette is then transfected into the host organism, which is allowed to proliferate. Samples are taken once the culture reaches a specified optical density, making the signals of multiple experiments comparable. Both in the presence and in the absence of the ligand, the brightness of the emitted light after excitation with radiation of a specific wavelength is measured. By comparing the measured fluorescence intensity values, the difference in concentration of the reporter gene is estimated.

5.2 Design and analysis of transcriptional neomycin-dependent riboswitches

Based on the aptamer N1M7, we constructed a riboswitch that controls gene expression in *E. coli*. To this end, we focused on transcriptional regulation mediated by Rho-independent termination, a mechanism common in prokaryotes.

We used the *in silico* pipeline developed previously for designing theophylline and tetracycline riboswitches (Domin et al., 2017; Wachsmuth, Findeiß, et al., 2013) to integrate the N1M7 aptamer with artificial terminator hairpins. The designed neomycin riboswitches indeed increase the transcription level of the reporter mRNA in presence of neomycin *in vivo*. A detailed analysis of the constructs, however, shows that their function *in vivo* is context-dependent. In particular, we identify a $5'$ leader hairpin as essential element for the function of these neomycin riboswitches. On the other hand, leader sequences may also abrogate the function by interfering with the riboswitch. We demonstrate here that folding simulations predict the interference and thus can be used to identify functional constructs *in silico*.

5.3 Materials and methods

5.3.1 Biochemical experiments

The paper this chapter is based on was a joint work of the author, his supervisors, and collaborators from the Biochemistry Group of Leipzig University. Since the author of the thesis mainly contributed the bioinformatic methods and the evaluation of data, the details of the conducted laboratory experiments shall be omitted here, and only a short overview is given in the following.

The designed riboswitch constructs where tested *in vivo* in cells of bacterium *E. coli*. Since the aptamer N1M7 (Weigand, Sanchez, et al., 2008) used in the riboswitch candidates responds to the antibiotic neomycin, a special strain resistant to this bactericide (Quan et al., 2015) was used to conduct the experiments. Together with the *araBAD* promotor and one of the well-known reporters the enhanced green fluorescent protein gene (*egfp*) or the β-galactosidase gene (*bgaB*), the candidates were inserted into a plasmid and thus transferred into the cells. The expression of the constructs was then tested using either a fluorescence measurement (for eGFP), an ONPG test (for β-galactosidase (BgaB)), or a northern blot as described in the previous sections.

5.3.2 Probabilities of RNA conformations

As explained in Section 2.2.4, the probabilities of specific structures or sets of structures with specific properties can computed efficiently by the means of partition function folding algorithms. The *ViennaRNA* package provides a versatile way of specifying structural constraints and was thus used to compute partition functions and probabilities for RNA structures with the desired structural features, e. g., a terminator hairpin or a binding pocket for a certain ligand (Lorenz, Bernhart, et al., 2011; Lorenz, Hofacker, and P. F. Stadler, 2016).

5.3.3 Simulation of cotranscriptional folding

While initial folding intermediates of RNA form at time-scales of tens of microseconds, the formation of native hairpins appears at millisecond time-scales (Bevilacqua and Blose, 2008; Melnykov et al., 2015; Mohan et al., 2009; Pörschke, Uhlenbeck, and Martin, 1973), and the refolding of secondary structure elements may take even longer. In comparison, RNA is transcribed by *E. coli* RNAP with a rate of $30\,\mathrm{nt\,s^{-1}}$ to $90\,\mathrm{nt\,s^{-1}}$ (Ryals, Little, and Bremer, 1982; Vogel and Jensen, 1994). Thus, RNA folding forms intermediate structures long before the entire molecule is transcribed, i. e., while only part of the transcript is available to form structures. The structures formed initially thus may refold as transcription proceeds. This process of cotranscriptional folding (Lai, Proctor, and Meyer, 2013) plays an important role in particular for *transcriptional* riboswitches, since incomplete, metastable intermediate structures may be quite different from the thermodynamic ground state (Lutz et al., 2014). Cotranscriptional folding can be assessed either by stochastic sampling of folding trajectories (Flamm, Fontana, et al., 2000; Geis et al., 2008; Xayaphoummine, Bucher, and Isambert, 2005) or by analyzing the energy landscapes for each elongation step (Hofacker, Flamm, et al., 2010). We opted for the latter method.

For each length of the nascent transcript, we enumerated all secondary structures in an energy band above the ground state with *RNAsubopt* (Wuchty et al., 1999), also a component of the *ViennaRNA* package. Then we used *Barriers* (Flamm, Hofacker, P. F. Stadler, et al., 2002) to produce a coarse-grained representation of the energy landscape comprising the low energy minima as well as the saddle points between them. *Barriers* assigns each structure to a basin of attraction. The individual landscapes were then integrated using *BarMap* (Hofacker, Flamm, et al., 2010). In brief, *BarMap* determines the correspondence of energy basins in the landscapes of consecutive transcriptional elongation steps. This allows to efficiently simulate the folding dynamics. To determine the energy thresholds for the enumeration of structures with *RNAsubopt* in the individual landscapes, we used the quality scores for both the enumeration and the simulation provided by *BarMap-QA* (Kühnl, P. F. Stadler, and Findeiß, 2019). We set a simulation stop time of 1 000 au (arbitrary time units) for *BarMap*. Simulation results were plotted with Grace (The Grace contributors, 2015). Candidate riboswitches were screened to satisfy the following criteria: (i) the leader sequence does not interfere with the formation of the aptamer's binding pocket, (ii) the binding pocket is significantly populated during the transcription of the spacer and the 5′ part of the terminator, and (iii) the terminator hairpin dominates the structure ensemble as soon as it is fully transcribed. For a precise description of the preparation steps, quality metrics, and the post-processing, we refer to (Kühnl, P. F. Stadler, and Findeiß, 2019). To allow for easy reproduction, the entire simulation pipeline was packaged into a self-contained and publicly available Docker image.[1]

---

[1] `https://www.bioinf.uni-leipzig.de/Software/BarMap_QA/`

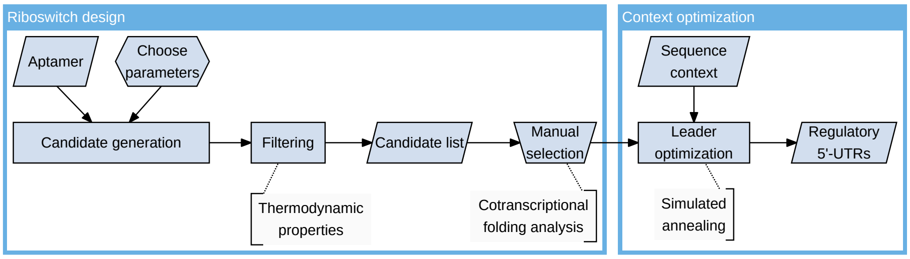


Figure 27: Flowchart of the design process used to create neomycin-dependent riboswitches and to include them into regulatory $5'$ UTRs. Starting from a given aptamer sequence and structure, riboswitch candidates are generated and selected as described in the text.

5.3.4 Design of transcriptional riboswitches

The design process consists of two separate phases. First, the riboswitch itself is designed from the given aptamer sequence. Second, for a set of designed riboswitch constructs, a decoupling leader (dL) is constructed, which isolates the riboswitches from the original leader (oL) sequence found upstream on the plasmid. This second step may become necessary if oL has the potential to interfere with the structure formation of the designed riboswitches (cf. Section 5.4.1). The full process is depicted as a flowchart in Figure 27.

To engineer the switch, a software pipeline similar to the one described by Wachsmuth, Findeiß, et al. (2013) including updated filter rules has been used. Briefly, this software is given the aptamer sequence and its ligand-binding structure as input, generates riboswitch candidates according to a specified pattern (*generation step*), and then applies a set of filters to remove unsuitable candidates (*filtering step*).

The generation step appends a random spacer sequence of varying length to the aptamer sequence. Then, the reverse complement of the last k nucleotides of the aptamer sequence is appended, forming a stable hairpin loop with the aptamer and disrupting the formation of the ligand-binding structure. The value of k varies from two to half of the aptamer length among the candidates. In addition, a U-stretch consisting of eight uracil residues is added, completing the structure of an intrinsic terminator (Ray-Soni, Bellecourt, and Landick, 2016).

To remove potentially faulty constructs, the filtering step applies a set of filters as described in Wachsmuth, Findeiß, et al. (2013), removing each sequence that violates any of the filter rules. The rules ensure correct terminator formation in the MFE structure and scan for possibly interfering intermediate structures by the means of thermodynamic folding simulations of a set of subsequences of the full switch. Additionally, we added a probability-based filter ensuring that, for the full sequence, the terminator structure forms with a probability of at least 95%. Also, it was ensured that at least two base pairs of the terminator hairpin can form despite the presence of the binding-competent aptamer structure. These *seed base pairs* facilitate the rapid formation of the terminator hairpin when the ligand is not present since the zippering of a helix happens at a higher rate than the nucleation of the first base pair (Mohan et al., 2009; Pörschke, Uhlenbeck, and Martin, 1973). Finally, promising candidates resulting from this selection process were analyzed using cotranscriptional folding simulations as described in the previous section.

5.3.5 Designing a decoupling leader sequence

To prevent the predicted interference of the oL located upstream in the $5'$ UTR with the designed riboswitches, a sequence insert that effectively decouples leader and riboswitch has been designed. To this end, we defined an objective function F as described below and optimized a sequence with respect to F. The sequence with the best score was then cloned into the vector immediately upstream of the riboswitch, cf. Figure 33 in Section A.1. To keep the required experimental effort as low as possible, only one optimized leader suitable for all riboswitch constructs was designed.

Let ℓ denote the oL, and $R = \{r_1, \ldots, r_n\}$ be a set of n riboswitches. The goal is to construct an insert $x_m \in \{A, U, G, C\}^m$ of length m that minimizes the objective function $F(x_m \mid \ell, R)$ measuring the interference of the upstream sequence ℓx_m with each of the riboswitches, subject to a constant length m. A natural measure for the isolation of ℓx_m and the riboswitches is the probability

$$p_{\text{unpaired}}(\ell x_m, R) = \prod_{r \in R} \frac{\mathrm{Z}[\ell x_m] \cdot \mathrm{Z}[r]}{\mathrm{Z}[\ell x_m r]},$$

that all base pairs occur either in the upstream part ℓx_m of the sequence (structures in $\mathrm{Z}[\ell x_m]$) or within the riboswitches (structures in $\mathrm{Z}[r]$), but not between the two substructures. Using $\Delta G(x) = -RT \ln \mathrm{Z}[x]$, it can be expressed equivalently in terms of Gibbs free energies, with the added benefit of numerical stability. We therefore use the following objective function:

$$F(x_m \mid \ell, R) = \sum_{r \in R} \Delta G(\ell x_m) + \Delta G(r) - \Delta G(\mathrm{Z}[\ell x_m r]) \longrightarrow \min .$$

The optimization of x_m was carried out using a standard simulated annealing procedure (Kirkpatrick, Gelatt, and Vecchi, 1983), starting at a random sequence and applying single nucleotide mutations to the insert to generate new candidates. A proposal sequence x'_m was *always* accepted if it performed better than the current state x_m (i. e., if $F(x'_m) < F(x_m)$), and otherwise with a probability of $\exp((F(x_m) - F(x'_m))/\tau)$, where the “annealing temperature” τ was slowly decreased with time. More specifically, we set the initial temperature to $\tau_0 = 1000$ and, after each mutation, cooled it down by letting $\tau_{n+1} = 0.97 \cdot \tau_n$. We used the rejection of $3m$ consecutive proposals as stopping criterion, where $3m$ is the number of neighbor sequences which can be obtained by applying a single point mutation to the candidate insert of length m. This ensures that, on average, about two thirds of its neighbors are sampled before terminating the optimization run. During the optimization, m was fixed because longer sequences generally have a higher potential to form stable structures fulfilling the objective, which could cause degenerate optimization runs with candidates of ever-growing sequence length.

5.4 Results

Many of our results are presented as bar plots of fluorescence intensity. The numerical values and raw data used to generate these can be found in Sec-

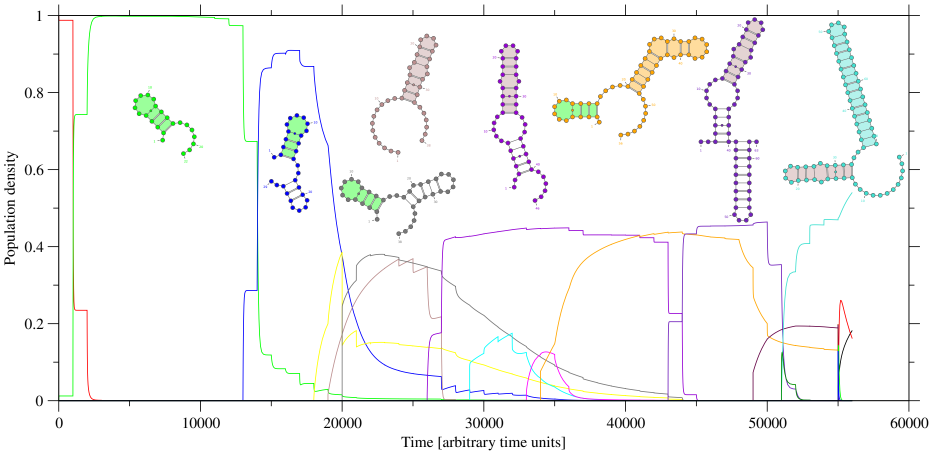


Figure 28: Influence of the 5′ oL, which is predicted to interfere with the used aptamer *in silico*, on the kinetics of riboswitch N1M7-D. The individual curves show the population of different macrostates during the elongation of the transcript. The most stable structure of highly populated states is shown in the respective color. Important sub-structures are shaded.
In essence, oL forms two distinct, stable structures: a small hairpin of four base pairs (green) that is compatible with the aptamer's binding pocket (orange), and a larger hairpin (brown) incorporating an interior loop that is extended later (light and dark purple) and interferes with the aptamer region. Both structural shapes are equally populated with about 40%, which means that the aptamer is not available for binding the ligand in about every other transcribed molecule. The terminator (cyan) forms immediately after it has been transcribed. This explains the permanent *off* state that has been observed for Lpl-N1M7-D experimentally, cf. Figure 36.

tion A.2, Table 1. Sequence data is available as a supplementary spreadsheet file accompanying the publication.

5.4.1 Design of artificial neomycin riboswitches

Using computational predictions of ligand-induced differences in RNA secondary structure formation, we designed a set of transcription-regulating riboswitches based on the well-characterized neomycin aptamer N1M7, Figure 26. These were evaluated *in vivo* in *E. coli* strain SQ171 using eGFP as reporter. This strain is neomycin-resistant, as all endogenous rrn operons were deleted and replaced by a plasmid-borne version carrying the mutated neomycin target site A1408G (Quan et al., 2015).

The plasmid used for assaying our riboswitch constructs contained a leader sequence we termed *oL* immediately downstream of the promoter. Since its effect on transcription was unknown, it was not attempted to delete it. *In silico* analysis of the constructs suggested, however, that the oL sequence interferes with the structure formation of the N1M7 aptamer and thereby prevents the correct folding of its ligand binding pocket, Figure 28. To remedy this issue, we designed a dL with a length of 15 nt that reduces the probability that base pairs form between oL and the riboswitch domain to less than 1%. It does so by forming the stable hairpin LH1, i. e., LH1 = oL + dL (cf. Figure 33).

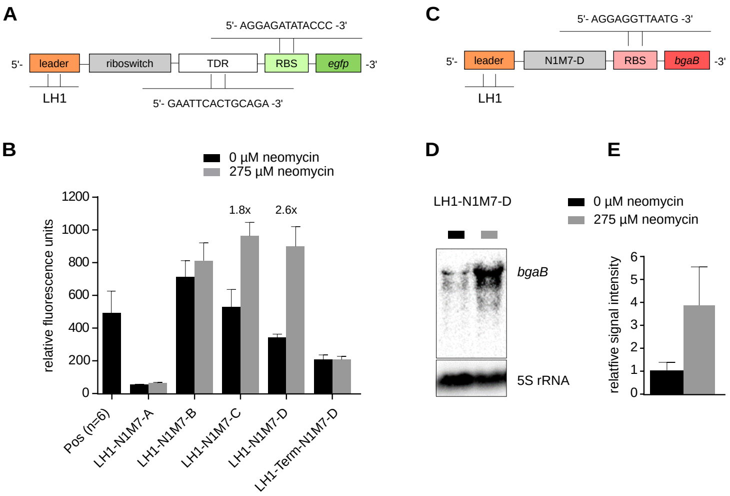


Figure 29: Design and analysis of N1M7 riboswitch constructs. (*A*) Schematic overview of the riboswitch constructs for the eGFP assay. The riboswitches are flanked upstream by leader hairpin (lH)1 and downstream by the terminator downstream region (TDR), RBS and *egfp*. (*B*) Fluorescence intensity of the measured N1M7 constructs in the absence (black) and presence (grey) of neomycin. The positive control, a transcript consisting of a single adenosine residue followed by a poly-U stretch and the TDR, shows the fluorescence intensity without leader and riboswitch sequence. The terminator efficiency was verified with construct lH1-term-N1M7-D. Here, the $5'$ part of the aptamer sequence that is not overlapping with the intrinsic terminator was deleted, such that the terminator forms irrespective of the presence of neomycin. If not indicated otherwise, measurements were performed using three independent replicates. (*C*) Schematic presentation of a neomycin riboswitch construct for northern blot analysis. Downstream of the riboswitch, the construct carries the BgaB reporter gene (*bgaB*) including the RBS published in Wachsmuth, Findeiß, et al. (2013). Due to the design strategy, this construct does not carry a TDR. (*D*) Northern blot with 10 µg of total RNA of *E. coli* strain SQ171 per lane regulated by LH1-N1M7-D, in the absence and presence of neomycin. 5S rRNA was used as internal standard. (*E*) Northern blot quantification of *bgaB* expression of three independent samples.

Four candidates prefixed with LH1 were selected for in-depth analysis, Figure 34. Fluorescence measurements of reporter gene expression (cf. Figure 29A–B) showed that LH1-N1M7-C and LH1-N1M7-D function as *on*-switches, i. e., the ligand causes up-regulation of the reporter. In contrast, LH1-N1M7-A remains in a permanent *off* state while LH1-N1M7-B exhibits a permanent *on* state. As a representative example, northern blots in the presence and absence of neomycin were used to validate that LH1-N1M7-D regulates the amount of transcripts in the cell, Figure 29C–E. Blotting experiments targeting *egfp* mRNA resulted in a low signal-to-noise ratio. Hence, the reporter gene was replaced by the well-established BgaB reporter that our collaboration partners have used successfully for detection in northern blot analyses of theophylline-dependent riboswitches before (Wachsmuth, Findeiß, et al., 2013).

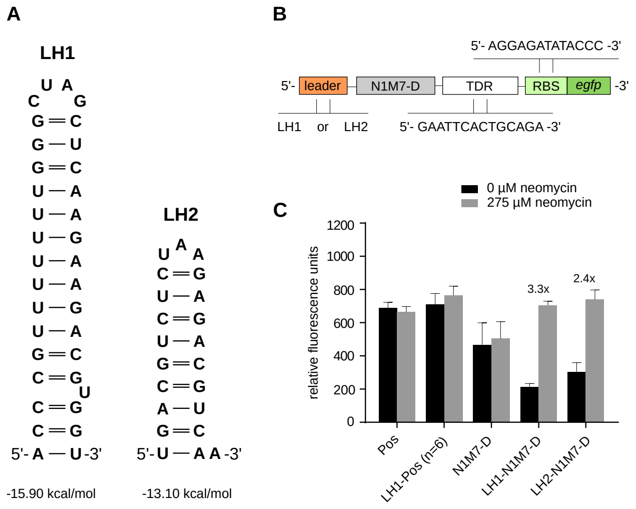


Figure 30: Fluorescence intensity of the riboswitch N1M7-D in conjunction with two different leader hairpins LH1 and LH2. (*A*) Sequence and secondary structure of LH1 and LH2. (*B*) Schematic overview of the constructs used in panel C. (*C*) Fluorescence intensity of the two leaders LH1 and LH2 placed upstream of N1M7-D. Legend as described in Figure 29.

5.4.2 Riboswitch N1M7-D requires a 5′ hairpin structure

Instead of designing a decoupling sequence for oL with a complicated optimization method as described in the previous section, the more obvious approach seems to simply remove oL from the plasmid and place the riboswitch candidates directly downstream of the transcription start site (TSS) represented by an adenosine residue. It turns out, however, that the removal of all leader sequences from the functional riboswitch LH1-N1M7-D yields a *non-functional* construct termed N1M7-D, which is no longer sensitive to neomycin (Figure 30C).

The riboswitch function was rescued by inserting a second, computationally designed leader hairpin LH2 (Figure 30A) resulting in the functional riboswitch LH2-N1M7-D, Figure 30C.

To rule out that the rescue is the consequence of a changed TSS, or a direct interaction of the leader hairpin with the riboswitch domain, we tested two short unstructured leader sequences U1 and U2, 12 nt and 14 nt in length, Figure 31A. In the computational design of U1 and U2, base pairing interactions with the riboswitch domain were avoided. Both constructs, U1-N1M7-D and U2-N1M7-D, were functional with a fold change of 1.6 and 1.4 (respectively), however, their fluorescence intensity was significantly reduced, resembling the *off* state of the functional construct LH1-N1M7-D, Figure 31C.

Next, the leader hairpin LH1 was re-added to the 5′ ends of these impaired riboswitches, Figure 31B. As a result, fluorescence activity similar to the original constructs was restored for both LH1-U1-N1M7-D and LH1-U2-N1M7-D, Figure 31C.

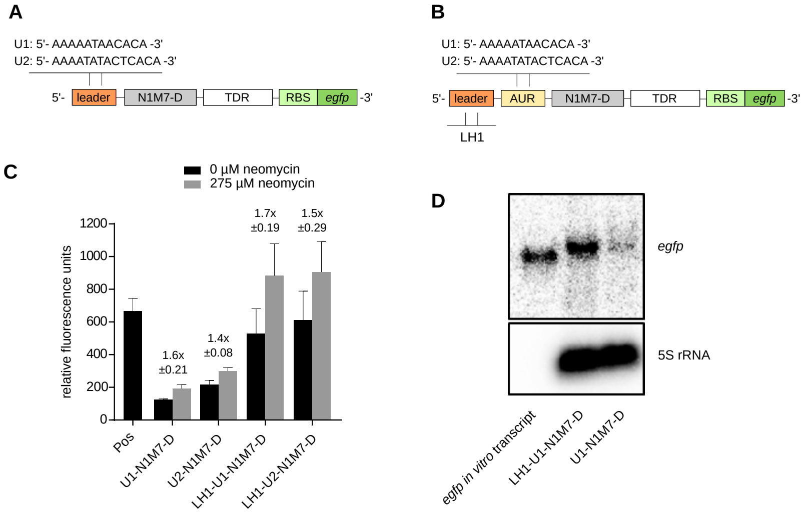


Figure 31: Fluorescence intensity of the N1M7-D riboswitch with short unstructured leaders. (*A*) Schematics showing the position of the unstructured sequences unstructured region (U)1 and U2 as leaders upstream of N1M7-D and (*B*) as aptamer upstream region (AUR) between lH1 and N1M7-D. (*C*) Fluorescence intensity of the constructs shown in (*A*) and (*B*). Legend as in Figure 29, *p*-values for the paired *t*-tests are given in Table 3. (*D*) Northern blot of 10 µg of total RNA per lane from *E. coli* strain SQ171 regulated by U1-n1m7-D or lH1-U1-n1m7-D, in the presence of 275 µM neomycin. Herein, *egfp* was used as reporter gene and 5S rRNA as internal standard. A 1100 nucleotide (nt) *egfp in vitro* transcript was used as positive control. Note that the band of *in vivo egfp* is located about 100–200 nt higher due to the additional rrnB T1 and T2 terminator length, which was not a factor in size calculation for the *in vitro* transcribed control.

5.4.3 Destabilizing the leader hairpin decreases reporter gene expression

Since unstructured leader regions abrogated the switching behavior, we investigated whether the stability of the leader hairpin has a systematic effect on the riboswitch. To this end, destabilizing point mutations were introduced into the leader hairpin LH1. Thus, two new leaders LM1 and LM2 were obtained (Figure 32A), each lacking two base pairs. The destabilization was reverted by compensatory mutations in LM1C and LM2C, restoring the stem of LH1. We found that none of these mutations significantly affected the functionality of the riboswitch N1M7-D. The activation ratios of these constructs range between 2.8-fold to 3.4-fold, comparable to LH1-N1M7-D, cf. Table 2.

There is, however, a trend in the overall fluorescence intensity: most stable hairpins lead to a higher eGFP expression, both in presence and absence of neomycin (Figure 32B). Constructs lacking the riboswitch domain, i. e., those consisting of a leader hairpin and the positive control (Figure 32C), show the same trend in the overall fluorescence intensity.

Similarly, the alternative leader hairpin LH2 was destabilized to obtain hairpins LM3, LM4 and LM5, Figure 35A. The construct LM3-N1M7-D, containing the least stable leader hairpin with MFE of $\Delta G = -5.5\,\mathrm{kcal\,mol^{-1}}$, exhibits a constitutive *on* state (Figure 35B) but still shows a moderate ac-

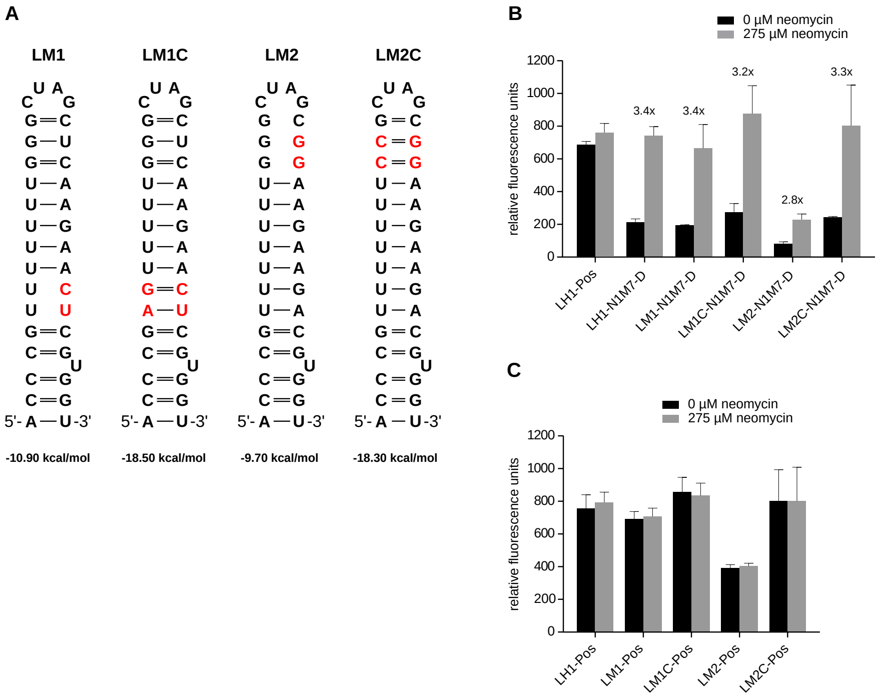


Figure 32: Impact of point mutations in the leader hairpin lH1. Fluorescence intensity of the modified leader LM1, LM1C, LM2 and LM2C (*A*) in conjunction with N1M7-D (*B*) or positive control (Pos) (*C*), along with the controls lH1-N1M7-D or lH1-Pos. Legend as described in Figure 29. Mutation details and MFE are given in Table 2.

tivation rate of about 1.5. Two compensatory mutations restore the hairpin (LM3C, $\Delta G = -13.9\,\mathrm{kcal\,mol^{-1}}$) to almost the same folding energy as LH2 ($\Delta G = -13.1\,\mathrm{kcal\,mol^{-1}}$) and rescue the function of the riboswitch. LM4-N1M7-D, with $\Delta G = -10.1\,\mathrm{kcal\,mol^{-1}}$, shows almost no difference in the activation rate or fluorescence intensity compared to LH2-N1M7-D or LM3C-N1M7-D. The fluorescence intensity of LM5-N1M7-D was decreased compared to LM4-N1M7-D, while the activation ratio virtually remained unchanged.

In summary, we observe that (i) a sufficiently stable leader hairpin appears to be required for functional constructs and (ii) the stability of the leader hairpin correlates with the constructs' fluorescence intensity.

5.4.4 Upstream sequence context may impair riboswitch function

As mentioned before, we predicted that oL interferes with the aptamer N1M7 using *in silico* simulations. To verify this finding experimentally, oL was inserted between the leader hairpins and the riboswitch domains, resulting in the constructs LH1-oL-N1M7-D and LH2-oL-N1M7-D. In addition, oL-N1M7-D was analyzed without any lH. All three constructs showed no neomycin-dependent regulation of eGFP expression *in vivo*, cf. Figure 36, even though the corresponding constructs without oL between lH1 and N1M7-D (namely

LH1-N1M7-D, LH1-U1-N1M7-D, and LH1-U2-N1M7-D) were functional, cf. Section 5.4.2.

The expression level of LH1-oL-N1M7-D or LH2-oL-N1M7-D resembles the *off* state of the functional switch LH1-N1M7-D with a fluorescence intensity of 200–400 RFU (relative fluorescence unit (RFU)). Without a leader hairpin upstream of oL, the fluorescence reached an intensity of about 150 RFU. These low fluorescence intensities combined with the disturbed switching behaviour is in accordance with the predicted interaction of the oL with the neomycin binding pocket of the aptamer sequence N1M7. According to the simulations, this interaction not only prevents the formation of the neomycin binding pocket, but also facilitates the emergence of the terminator hairpin.

5.5 Discussion

In this work, we combined a synthetic neomycin aptamer with computationally designed terminator hairpins that abrogate transcription in the absence of the aptamer-specific ligand. The design process followed our earlier successful construction of theophylline- and tetracycline-dependent, transcription-regulating riboswitches (Domin et al., 2017; Wachsmuth, Findeiß, et al., 2013). Here, we employed the neomycin aptamer N1M7, exhibiting an extraordinarily high binding affinity of $K_d = (9.2 \pm 1.3)\,\mathrm{nM}$ (Weigand, Schmidtke, et al., 2011), as ligand sensor. Of the four computationally designed constructs that were tested *in vivo*, LH1-N1M7-C and LH1-N1M7-D were functional neomycin riboswitches. Northern blot analysis proved that LH1-N1M7-D acts as a transcriptional regulator, as intended. This demonstrates that our *in silico* approach can be successfully applied to novel aptamers.

The function of our artificial neomycin riboswitches depends on a stable 5′ hairpin structure. A plausible explanation for this requirement is an increased resistance to mRNA degradation: the half-life of a mutant *ompA* transcript increases 3–4-fold when such a hairpin is added upstream of its single-stranded 5′ UTR sequence (Emory, Bouvet, and Belasco, 1992). ROSE elements – RNA thermometers in the 5′ UTR controlling many small heat shock genes in *E. coli* – consist of at least two consecutive hairpins, and only the last one is thermosensitive and regulating the mRNA's translation rate (Krajewski and Narberhaus, 2014). Interestingly, similar hairpins are occasionally also located upstream of naturally occurring riboswitches. For example, Roth, Winkler, et al. (2007) analyzed a preQ$_1$-dependent riboswitch exhibiting a small 5′-located hairpin that was shown not to be required for ligand binding. Its function might be to fine-tune the half-life of the transcript. In gram-negative bacteria including *E. coli*, mRNA degradation is carried out by the degradosome, a complex of several proteins including ribonuclease E (RNase E), supported by RNA 5′ pyrophosphohydrolase (RppH) (Jiang, Diwa, and Belasco, 2000; Mackie, 1998). Stable 5′ structures are known to effectively obstruct the enzymatic activity of RppH and, thus, of the degradosome, explaining mRNA stabilization (Deana, Celesnik, and Belasco, 2008). Our data show that the effect of the 5′ hairpin depends on its thermodynamic stability rather than its sequence, lending support to the hypothesis that stable 5′ hairpins play a general protective role in the process of mRNA degradation. An increased life-time of the mRNA would also explain the observed correlation between thermodynamic stability and fluorescence intensity. In the positive control, we observed no dependence of fluorescence intensity on the leader hairpin. However, this construct forms a very stable hairpin (instead of the flexible aptamer structure), which likely explains its stability.

While the leader hairpin is crucial for our neomycin riboswitch, no corresponding structure was required for similar theophylline or tetracycline riboswitches (Domin et al., 2017; Wachsmuth, Findeiß, et al., 2013), cf. Figure 37 for an overview of their thermodynamic ground states. We suspect that this is the consequence of differences in aptamer stability. While the neomycin aptamer has MFE of $\Delta G = -6.2\,\mathrm{kcal\,mol^{-1}}$ (Weigand, Schmidtke, et al., 2011), both the theophylline aptamer used by Wachsmuth, Findeiß, et al. (2013) ($\Delta G = -12.1\,\mathrm{kcal\,mol^{-1}}$) and the tetracycline aptamer used by Domin et al. (2017) ($\Delta G = -19.8\,\mathrm{kcal\,mol^{-1}}$) are considerably more stable. We therefore hypothesize that, in the presence of the ligand, a highly stable aptamer–ligand complex sufficiently protects the transcript from degradation. In absence of the ligand, the terminator hairpin forms rapidly and leaves the transcript with an unstructured $5'$ end, facilitating its quick degradation and thus contributing to a high activation ratio. The observed change in fluorescence activity upon ligand binding is therefore a result of two independent effects: an increased transcription rate due to the suppression of the terminator hairpin formation, and a reduced transcript degradation due to the stabilization of the aptamer structure located at its $5'$ end.

Synthetic biology is concerned with solving sophisticated design problems with minimal expenditure of time, effort and money and thus strives to construct complex genetic circuits as a combination of fully modular, independently optimized components. Natural biological systems, however, have not evolved to adhere to the paradigm of engineering. It is well-documented, therefore, that the embedding of engineered circuits into living organisms faces a multitude of problems, mostly from unintended and often unexpected interaction with the natural context. Orthogonal systems attempt to avoid or at least minimize this problem (Villa et al., 2019). Some quite spectacular success stories, e. g., the construction of re-engineered versions of secondary metabolite biosynthetic pathways (Medema et al., 2011) or the assembly of artificial operons (Basitta et al., 2017) have raised the hope for a “plug-and-play” synthetic biology that allows the usage of pre-fabricated components in a combinatorial fashion. Even within an orthogonal system, however, it appears that this goal will remain elusive in many cases, at least in the strict sense of devising context-independent components (Karamasioti, Lormeau, and Stelling, 2017).

Our data show that already conceptually very simple devices such as transcriptional riboswitches are more than a simple concatenation of building blocks. This is not surprising, given that the function of a transcriptional riboswitch is not just determined by the properties of aptamer and terminator, but depends on a delicate balance of mutually exclusive structural features and a carefully orchestrated time course of folding and refolding during its transcription (Quarta, Sin, and Schlick, 2012). As a consequence, prefabricated modules require adaptation and partial re-design to properly function in combination. In the case of RNA devices, the intrinsically global nature of RNA structure formation, in which base pairs are not restricted to local interaction, explains the imperfect modularity. At the same time, computational models of RNA folding are capable of capturing the undesired side effects and make it possible to algorithmically optimize novel designs based on existing components.

In previous work, for example, we observed that only one of two terminator hairpins functioned properly in the synthetic theophylline riboswitch RS10 (Wachsmuth, Domin, et al., 2015). Using a cotranscriptional folding algorithm, we found that the inactive construct harbored an intermediate structure that likely acted as a kinetic trap, delaying the formation of the terminator hairpin sufficiently to render it inactive.

The adaptation of the interaction between the modular components is, by itself, not necessarily sufficient, as the presented work shows. The neomycin switches depend on an obligatory $5'$ leader hairpin, presumably because the riboswitch itself is not sufficiently stable in itself to protect the mRNA from degradation. In addition, it needs to be adapted to the sequence context of the transcript to avoid interference of the leader sequence with the riboswitch domain – again an effect that can be captured by the RNA secondary structure model and remedied by adapting the constructs *in silico*.

Taken together, we conclude that complex RNA devices cannot be engineered by simply combining pre-optimized components in a plug-and-play fashion. Modular components can be combined and put into a functional context, however, with the help of *in silico* simulation techniques. We can therefore do better than “plug and pray”: computationally, we can adapt the modules to function in context and optimize the constructs as a whole. We are confident that the reliability of the computational predictions will continue to improve as our understanding of the individual components and their underlying mechanisms of action evolves.

CHAPTER 6

Conclusion

Contents

6.1 Summary **100**
6.2 Outlook **106**
6.3 Concluding remarks **106**

RNAs are ubiquitous biomolecules found in every living cell. They serve many different purposes, including the transmission of genetic information in protein biosynthesis as messenger RNAs, catalytic and other metabolic functions as ribozymes, the regulation of gene expression as both small and long non-coding RNAs, and even as genomic defense mechanism via the RNA interference pathway. In many of these cases, the structural conformation of the RNA molecule plays a major role by mediating or modifying its biological function. For example, small structural elements like riboswitches or RNA thermometers are located in the $5'$ UTR of an mRNA transcript and regulate its expression by interacting with either the RNA polymerase during transcription or the ribosome during translation.

Remarkably, these examples do not only demonstrate the relevance of a specific, stable conformation, but show how dynamic structural rearrangements follow a precise schedule to achieve the desired regulatory effect. This emphasizes the necessity to understand the RNA folding process as a whole in order to make reliable predictions about the behaviour of a given RNA. While experimental approaches to gathering equilibrium structure data exist, e. g., in the form of SHAPE analyses or X-ray crystallography, a high-resolution analysis of the refolding process in real time is hardly possible. Therefore, the development of computational methods for kinetic folding analyses of RNAs is a worthwhile endeavour. This includes the assessment and refinement of RNA folding simulations as well as the underlying models.

This thesis tackles this challenge at various levels. Multiple criteria to assess the quality of folding simulations have been invented. A comprehensive software package called *BarMap-QA* to conduct cotranscriptional folding analyses has been developed. The statistical properties of free energies and probabilities of RNA structures have been studied in detail. Finally, the described methods were applied in practice to design a synthetic, transcriptional riboswitch responding to the antibiotic neomycin. The obtained construct was transfected into bacterial host cells by a collaborative partner and proved to be functional *in vivo*. In follow-up experiments, the interaction of the designed RNA device with its sequence context has been analyzed and was in accordance with the predictions of the computational models.

In this final chapter, the major findings of this work will highlighted again. Afterwards, an outlook of related topics and areas for potential future work is presented. Some final remarks close this thesis.

6.1 Summary

As described in Chapter 2, this work uses the abstraction of secondary structures as tractable model of RNA structures, i. e., structural conformations of a given RNA molecule are described as a set of non-crossing base pairs. The stability of a secondary structure x can be quantified in terms of its Gibbs free energy $\Delta G(x)$, and this energy can be predicted using the nearest-neighbor model and a set of energy parameters. Here, the parameters of Turner and Mathews (2010) were used. The secondary structure model allows to efficiently compute the MFE structure (Zuker and Stiegler, 1981) as well as the partition function of the structure ensemble (McCaskill, 1990) – and thus the equilibrium probabilities of arbitrary structures – with a polynomial time complexity of $\mathcal{O}(n^3)$ using dynamic programming. Specifically, this work relies on the *ViennaRNA* software package as implementation of the aforementioned algorithms. In equilibrium, the probability of an RNA structure x follows a Boltzmann distribution, i. e., it is proportional to its Boltzmann weight $\exp(-\beta \Delta G(x))$, where β is the inverse temperature, a (temperature-dependent) constant. With the availability of the partition function, equilibrium probabilities of arbitrary structures can be computed easily.

While this thermodynamic approach to RNA folding is efficient and sufficiently precise in many cases, the above-mentioned examples show that the assumption of the RNA being in a state of equilibrium is not always justified. In these cases, it is thus necessary to explicitly model the kinetics of the folding process. In this work, this is achieved by employing the concept of the RNA energy landscape (Flamm, Hofacker, P. F. Stadler, et al., 2002), which defines the neighborhood of a structure based on elementary transitions like the opening or closing of a single base pair. Adjacent structures are then assigned transition rate coefficients based on the difference of their free energies via the Metropolis rule (Metropolis et al., 1953). The folding reaction can then be considered as a continuous-time Markov process, for which the possible structures serve as the set of states. The population of the states at a given time τ can then be computed by determining the matrix exponential $\exp(\tau \mathbf{R})$, which is achieved by a diagonalization of the rate matrix $\mathbf{R}$.

As the number of structures grows exponentially with the sequence length of an RNA (Stein and Waterman, 1979), a kinetic simulation including *all* possible structures is usually infeasible even for short molecules. To reduce the number states, several heuristics are used. The algorithm of Wuchty et al. (1999) is used to restrict the enumeration to structures with low energy and thus high stability. It is also possible to exclude structures containing – putatively unstable – isolated base pairs to obtain an ensemble of canonical structures. While these methods significantly cut down the number of structural conformations, their abundance still renders kinetic simulations intractable for all but the shortest molecules. To achieve another significant reduction in the number of states, a coarse graining based on the notion of gradient basins is applied to the structure ensemble (Flamm, Hofacker, P. F. Stadler, et al., 2002). To this end, a gradient descent is applied to the individual structures, following the steepest path down to a local minimum. All structures with an identical associated local minimum are then binned together in a single gradient basin, serving as macrostates for the kinetic simulation. Transition rate coefficients between the macrostates states are then calculated as weighted sums over the microscopic rate coefficients of the individual structures they contain. Using this coarse graining approach together with the proposed heuristics for structure reduction, kinetic folding simulations become feasible for RNA of lengths up to about 100 nt.

A disadvantage of the described model for RNA folding kinetics is the fact that the transition rate matrix – and thus the underlying energy landscape – is fixed. To be able to incorporate dynamic effects like temperature changes or sequence elongation, Hofacker, Flamm, et al. (2010) developed the *BarMap* framework, which allows to run folding simulation across a sequence of energy landscapes. This is achieved by constructing maps that allow to transfer the population of a macrostate in one landscape to a corresponding macrostate in the next one. The corresponding state is chosen such that its representing local minimum has a minimal base pair distance to the minimum of the mapped state.

Building on the described concepts, this thesis continues with the quality analysis of RNA folding models in Chapter 3. To do so, the concept of *ensemble coverage* is proposed. Let X be the structure ensemble of a given RNA sequence, i. e., the set of all possible secondary structures. As the partition function $\mathrm{Z} = \mathrm{Z}[X]$ of X can be computed efficiently, the probability $\Pr[Y]$ of any set of structures $Y \subseteq X$ can be calculated easily as $\Pr[Y] = \mathrm{Z}[Y] / \mathrm{Z}$ if only the partition function $\mathrm{Z}[Y]$ is known. If, for example, the individual structures in Y are explicitly constructed, $\mathrm{Z}[Y]$ can simply be computed by summing over their individual Boltzmann weights. In this work, the probability $\Pr[Y]$ is also referred to as the *coverage* of Y with respect to X because it measures to which extent a random sample from the equilibrium distribution of X is represented by the structures in Y. As the probability of a structure decreases exponentially with increasing free energy, high-energy structures can often be disregarded during folding analyses without impairing the quality of the results. Thus, it is interesting to construct a set of structures Y with a coverage close to 1, but $|Y| \ll |X|$.

One possibility to do so is the algorithm of (Wuchty et al., 1999), which constructs a set $X_{\leq \Delta G_{\mathrm{enum}}} \subseteq X$ containing all structures with an energy of at most ΔG_{enum}. Now, an obvious question is how to choose ΔG_{enum} to obtain a high coverage with as few structures as possible. Therefore, Section 3.2 analyzes the coverage obtained by enumerating various energy ranges for random RNAs of differing sizes. A value of $10\,\mathrm{kcal\,mol^{-1}}$ above the sequence MFE was found to produce high coverages of 96–100% for sequences up to a length of 160 nt. With increasing sequence length, the coverage in a given energy range decreases. Additionally, the computation time required to enumerate all structures becomes unbearable at some point, as the number of structures increases exponentially. Increasing ΔG_{enum} is thus often not an option for very long sequences, which limits the application of Wuchty's algorithm to sequences of the aforementioned sizes. Also, the coverage within a fixed energy range may vary significantly even for sequences of the same size. It should thus be checked for any given sequence if Wuchty's algorithm or a similar method is used to construct a representative set of structures for folding analyses.

An approach to further reduce the number of conformations in the ensemble is the restriction to canonical structures, i. e., structures with isolated base pairs are excluded, as these structures are putatively unstable. To assess whether this simplification excludes a significant fraction of likely structures as well, the distribution of coverage for canonical structures of random sequences was analyzed for multiple sequence lengths in Section 3.3. It has been shown that, with increasing sequence length, the median coverage of canonical structures decreases, while the variance of coverages significantly. For very short sequences of 30 nt, canonical structures have a coverage of more than 85% for half of the random sequences. For sequences of length 90 nt, this number reduces to only 47%. At this sequence length, the observed coverage values range from 0.02% to 95%. But even for sequences of length 30 nt, the canonical structures may exhibit a coverage of only a few percent in some cases. The consequence is that, whenever the ensemble is to be restricted to canonical structures, their coverage should be analyzed first to prevent unexpected, spurious results.

Section 3.4, finally, presents the author's software package *BarMap-QA*. It builds on the *BarMap* framework of Hofacker, Flamm, et al. (2010) to enable the user to conduct high-quality cotranscriptional folding analyzes of RNA molecules. The original, prototypical implementation of *BarMap* is very general and thus requires many manual steps to prepare and conduct this type of analysis. Its direct and indirect dependencies require a manual compilation. The generated output is verbose and hardly readable to humans, such that an evaluation of the results is cumbersome. Most significantly, it is not possible evaluate the reliability of the generated results. *BarMap-QA* alleviates these issues by providing a semi-automatic pipeline that guides the user through the process of model generation, runs the simulation, and provides tools to evaluate the results, including the automatic generation of plots of the entire simulation run. Importantly, three novel quality criteria measuring the simulation quality at different levels were proposed. *BarMap-QA* computes each of these scores for each component of the model (i. e., for each energy landscape, each representing the input sequence at a given length, and for each mapping step between any two consecutive landscapes), thus allowing the user to selectively adjust the simulation parameters to obtain optimal results with minimal computational effort. The developed package is distributed as free and open source software. To allow for an easy deployment and reproducible results in a predefined environment, it was packaged into a Docker image released at Docker Hub. The use of the Docker container technology allows to install *BarMap-QA* on any major platform with the use of a single command.

Chapter 4 deals with the statistics of free energies of random RNA sequences and gradient basins as well as with the distribution of Boltzmann weight within these. A thorough understanding of these is useful for modelling RNA folding as well as for the design of RNA sequences exhibiting a given set of features. Section 4.1 analyses the distribution of MFEs for random RNA sequences of different lengths. It reproduces the linear dependence of the expected MFE and its variance from the sequence length as well as the slight negative skew in the distribution of MFEs, which has already been shown by other studies (Wolfsheimer and Hartmann, 2010). Additionally, it is shown that a part of the skewness of the distributions for short sequence lengths can be explained by the truncation of the distribution at $0\,\mathrm{kcal\,mol^{-1}}$, which occurs because the open RNA chain is defined to have a free energy of zero. Furthermore, several families of distributions have been fitted to the MFE distributions of the various sequence lengths and their goodness of fit has been evaluated. For long sequences of 140 nt and more, a normal distribution was found to be a reasonable approximation. Very short sequences below 40 nt, a truncated normal distribution is the best choice as it can explicitly model the cutoff at $0\,\mathrm{kcal\,mol^{-1}}$. For sequence lengths in between, the skew normal distribution clearly outperforms the (regular) normal distribution. It has one more free parameter to adjust the skew of the distribution. The best fit for all lengths except the shortest sequences was obtained by fitting a generalized hyperbolic distribution, which has four free parameters and thus one more than the skew normal distribution. The goodness of fit was evaluated using both the Akaike information criterion and the Cramér–von Mises criterion.

In Section 4.2, canonical structures were revisited to analyze their abundance in the low-energy part of the structure ensemble. Canonical structures are defined as structures not containing any isolated base pairs. Excluding non-canonical structures from a kinetic folding simulation is an effective way to significantly reduce the number of structures under consideration. To which extend this approach will succeed, however, depends on the fraction of non-canonical structures in the ensemble of random sequences. In the analysis, it was found that the fraction of non-canonical structures in the low-energy part of the ensemble is decreasing with the sequence length for sequences of more than 40 nt. The average fraction reduces from over 90% to only 65% of the non-canonical structures for long sequences of length 200 nt. Notably, the spread of the distribution of fractions also increases: for the long sequences, fractions of non-canonical structures ranging from 44% to 98% have been observed.

The justification for restriction to canonical structures is the claim that isolated base pairs are mostly unstable and thus refold into another conformation immediately. To assess whether this is indeed a well-grounded assumption, the two subclasses of non-canonical structures that only have stable or unstable isolated pairs, respectively, have been computed as well. If all isolated base pairs of a non-canonical structure are unstable, then that structure can safely be excluded. A base pair is considered stable if its presence reduces the free energy of the structure. If all isolated base pairs are stable, then an exclusion can hardly be justified. It was shown that in the low-energy part of the ensemble random sequences, the fraction of non-canonical structures that only contain unstable isolated base pairs reduces with increasing sequence length. For sequence length 200 nt, only half of the canonical structures fall into this category. The second category, non-canonical structures with only stable isolated pairs, shows a reverse trend: it increases with the sequence length. While the fraction of these structures is not too big, it is as high as 13% percent of all low-energy structures for sequences of length 200 nt, which corresponds to a fraction of 20% of the non-canonical structures. This shows that every fifth structure excluded by the restriction to canonical pairs is actually stable and should be included in the analysis. In this light, the removal of non-canonical structures should be considered carefully and only be applied if really necessary.

Section 4.3 continues with an analysis of the lowest 10 000 minima of the energy landscapes of sequences of different lengths. As described above, the coarse graining algorithm implemented in *barriers* (Flamm, Hofacker, P. F. Stadler, et al., 2002) assigns each structure to the local minimum reached by applying a gradient descent to it, and the resulting gradient basins can then serve as macrostate in a kinetic simulation. The selected threshold of 10 000 minima was chosen as this size can still be processed using the simulation tool *Treekin* (Wolfinger et al., 2004) within a reasonable time. The energy range covered by the selected minima was then analyzed and found to follow a generalized extreme value distribution for a fixed sequence length. For increasing sequence lengths, the median covered energy range as well as the standard deviation of the range decrease exponentially. This means that, for long sequences, the explorable energy range is almost always very small. It is thus hard to analyze them. This length-dependent effect is not closely tied to the actual MFE of a given random sequence. This was demonstrated by analyzing the (negative) correlation between sequence MFE and explorable energy range, which rapidly approaches zero with increasing sequence length and is as low as -0.33 for sequences of length 200 nt.

In Section 4.4, it is shown that the distribution of energies within a single gradient basin approximately follows a normal distribution, with a slight deviation in the lower tail. Additionally, it is discussed how estimators of the first order statistic, i. e., the minimum of a sample of a given size, can be used to predict the MFE of a gradient basin if the parameters of the energy distribution and the number of structures are known. If the set of all structures in a gradient basin is considered a normal-distributed sample, then the MFE can be considered to be the first order statistic of the basin, which matches the data.

In addition to the distribution of energies, the distribution of probability mass in gradient basins is discussed too. Section 4.5 first shows empirical data demonstrating that the vast majority of the structures does not have any relevant probability mass. Only very few low-energy states dominate the basin's partition function. Secondly, a theoretical result concerning the distribution of Boltzmann weight in the basin is presented. Specifically it is shown that, if the energies of a basin follow a normal distribution with mean μ and variance σ^2, then the Boltzmann weight in the gradient basin follows a normal distribution with the same variance, but with mean $\mu - \beta\sigma^2$, where β is the inverse temperature. In other words, the distribution is shifted to the left by an amount of $\beta\sigma^2$. Using the truncated and rescaled cumulative distribution function of this distribution, the partition function of an example basin is then successfully predicted.

In Chapter 5, finally, the described models and approaches are put to use to design synthetic transcriptional riboswitches. To this end, a previously described RNA aptamer called N1M7 with a high binding affinity for the antibiotic neomycin (Weigand, Sanchez, et al., 2008) was used as input for an *in silico* design pipeline similar to that of Wachsmuth, Findeiß, et al. (2013). In previous studies, it was shown that riboswitches are not perfectly modular, and their regulatory function may be impaired depending on the genomic context they are used in (Domin et al., 2017). Using the RNA folding simulation techniques described in this work, it was shown that these detrimental effects on the functionality of the switch can often be explained by structural interactions with the surround sequences. Using an objective function based on thermodynamic criteria measuring the interaction of the riboswitch candidates with their surroundings, an optimization procedure was used to design a decoupling sequence insert to be placed upstream of the riboswitch candidates. The constructs where then combined with the fluorescent reporter gene eGFP and transfected into a neomycin-resistant strain of *E. coli*. Thus it was shown that the designed putative riboswitches are indeed functional, but need to be combined with the designed decoupling insert to prevent disrupting interactions with the sequence context. Furthermore, it was found that a stable hairpin structure is required at the $5'$ end of the transcript to ensure proper switching. The reason was assumed to be the increased resistance of the $5'$ end against dephosphorylation mediated by RppH (Deana, Celesnik, and Belasco, 2008). This enzymatic reaction is a necessary requirement for the degradation of the transcript by the endonuclease RNase E (Jiang, Diwa, and Belasco, 2000) in *E. coli*. Thus, a stable $5'$ hairpin may be necessary to prolong the lifetime of the transcript such that a sufficiently high readout is attained during the fluorescence measurements.

6.2 Outlook

While a comprehensive set of results has been presented in this work, there is always more to do than can possibly be done, and so some tasks and challenges had to be postponed for future projects. Some especially interesting open questions that arose during the author's studies shall be elaborated here.

The methodology used to perform kinetic folding simulations in this work is based on the coarse graining of the underlying energy landscape into gradient basins. It was also mentioned that the number of macrostates that can be considered in such a simulation is bounded due to the required computation time. It is thus an interesting question whether an even coarser state representation can be found, which still preserves the results of the simulation at least on a qualitative level. One possibility to do so could be the application of flow-based clustering or community detection methods, such that gradient basins connected by high transition rates a grouped into new macrostates. Another option is to repeat the application of the gradient-based coarse graining on the level of macrostates. This requires a formally clean definition of gradient descents on macrostates.

For the author's pipeline *BarMap-QA*, there are some advanced features that need yet to be. For example, the combination of cotranscriptional folding with ligand interactions is sensible next development step. Additionally, the ability to vary the transcription rate during cotranscriptional folding simulations is a new feature about to be released.

Concerning the distributions of free energies and Boltzmann weight within gradient basins, an important next step is the development of an effective, low-bias sampling procedure for basins. The method has to be significantly faster than a full enumeration, e. g., by means of local flooding (Entzian and Raden, 2020), and produce samples suitable for estimation of the mean and variance of the underlying normal distribution. If this can be achieved, partition function of the basin could be estimated, too, and it would be possible to effectively limit local flooding to the energy level required for the requested coverage. This would make the exploration of energy landscapes much more effective.

In the field of RNA design, there are numerous possible applications for the methods presented in this work. One specifically interesting task would be to create a dual-ligand riboswitch implementing an exclusive or (XOR) logic for controlling gene expression.

6.3 Concluding remarks

The simulation of RNA folding is a highly complex and versatile bioinformatical problem. In this thesis, it has been approached from many different perspectives with the goal to gain a deeper understanding of the behaviour of RNA molecules. While this versatility was a challenge on the one hand, it was a great opportunity to learn on the other. With the presented results, the author hopes to contribute a little piece to the big puzzle of life; a puzzle that is still far from being complete, and that will continue to challenge generations of researchers to come.

Appendices

APPENDIX A

Supplemental Information: Design of Artificial Cotranscriptional Riboswitches

The content presented in this appendix has been published as supplemental information for the following article:

> Günzel, C. et al. (2020). “Beyond Plug and Pray: Context Sensitivity and in silico Design of Artificial Neomycin Riboswitches”. In: *RNA Biology*, pp. 1–11. DOI: `10.1080/15476286.2020.1816336`

A.1 Supplemental figures

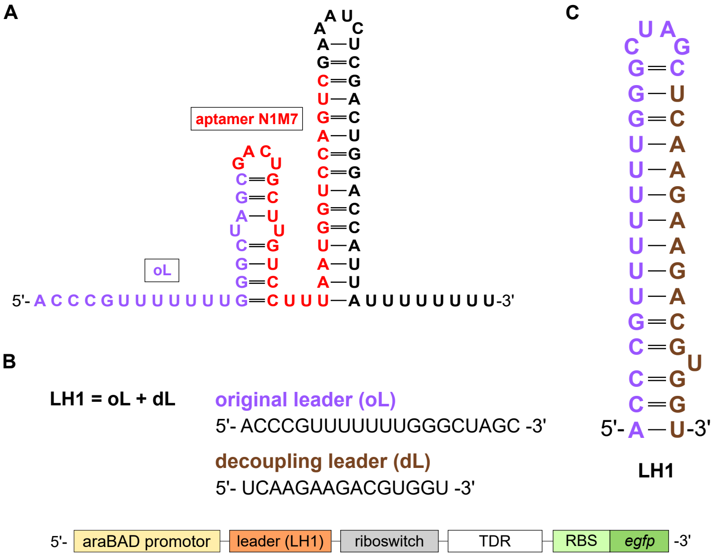

A
aptamer N1M7
oL
5'- A C C C G U U U U U U U G=C U U U—A U U U U U U U U-3'
B
LH1 = oL + dL
original leader (oL)
5'- ACCCGUUUUUUUGGGCUAGC -3'
decoupling leader (dL)
5'- UCAAGAAGACGUGGU -3'
5'- araBAD promotor leader (LH1) riboswitch TDR RBS egfp -3'
C
LH1

Figure 33: A decoupling leader for riboswitches. (*A*) Secondary structure of riboswitch N1M7-D with the oL as computed by free energy minimization (Lorenz, Bernhart, et al., 2011). (*B*) Schematic of riboswitch-surrounding elements in the plasmid pRSF1030Tp-SD-eGFP. A dL was inserted downstream of oL to protect the riboswitch from leader-dependent misfolding by forming the stable hairpin lH1 depicted in (*C*).

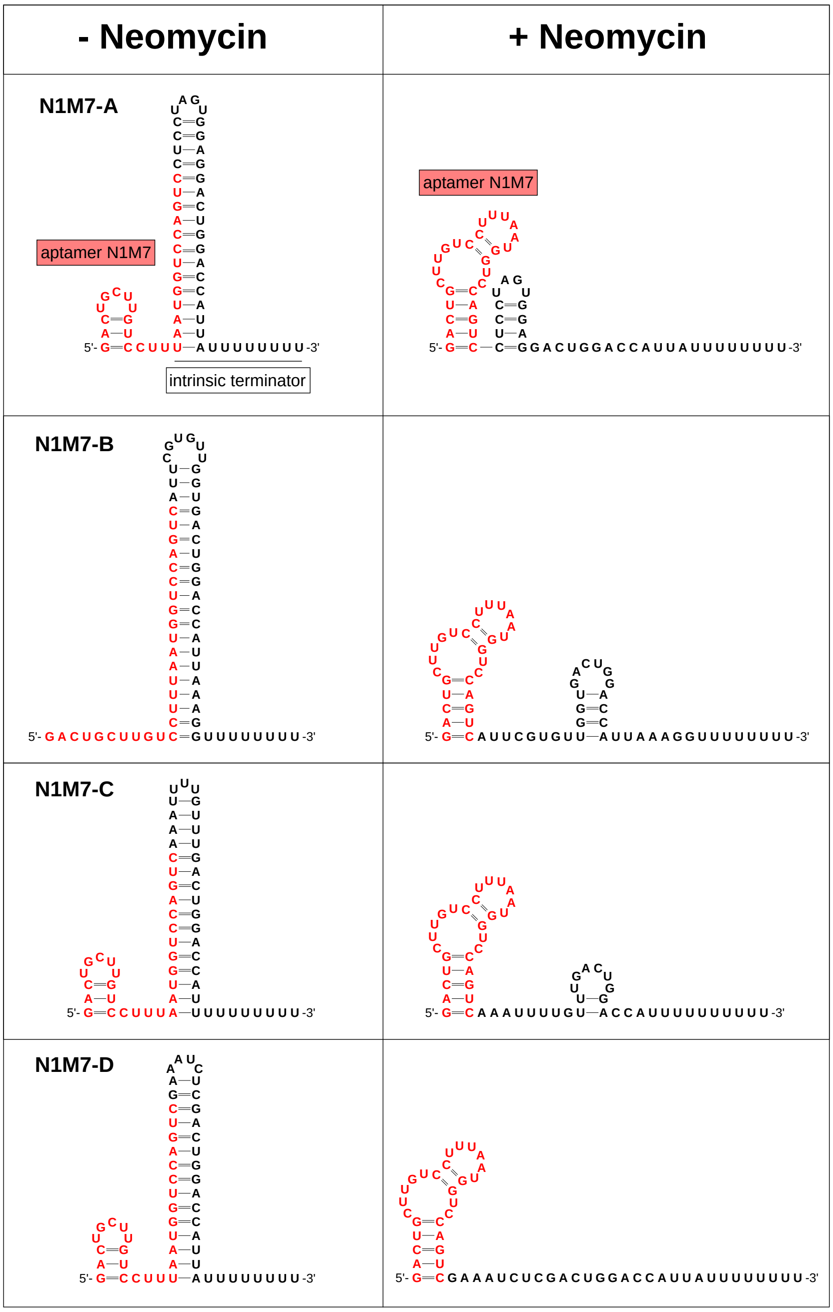


Figure 34: Secondary structure of the riboswitch constructs with the aptamer N1M7 used for the eGFP assay in Figure 29.

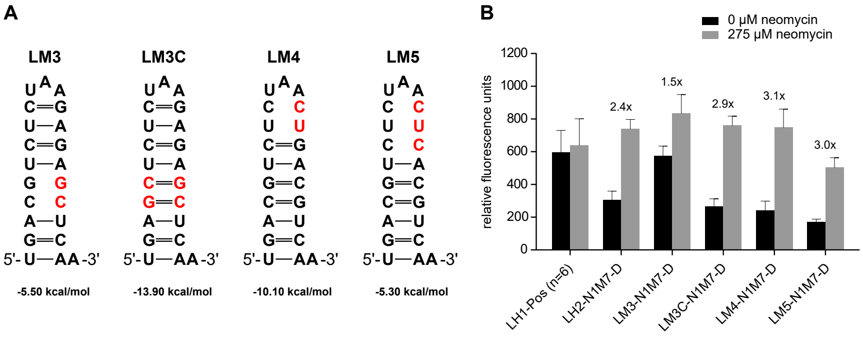


Figure 35: Impact of point mutations in the leader hairpin LH2. Fluorescence intensity of the modified leaders LM3, LM3C, LM4 and LM5, folded into their predicted secondary structure (*A*), in conjunction with N1M7-D (*B*). LH2-N1M7-D and LH1-Pos were used as control. Legend as described in Figure 29.

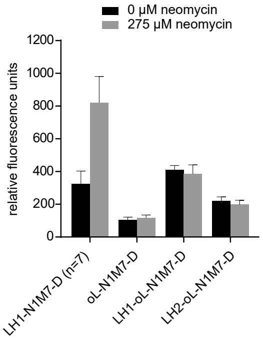


Figure 36: Influence of a sequence predicted to interfere with N1M7-D by *in silico* methods. The fluorescence intensity of N1M7-D with oL in conjunction with LH1 or LH2 is shown. Legend as described in Figure 29.

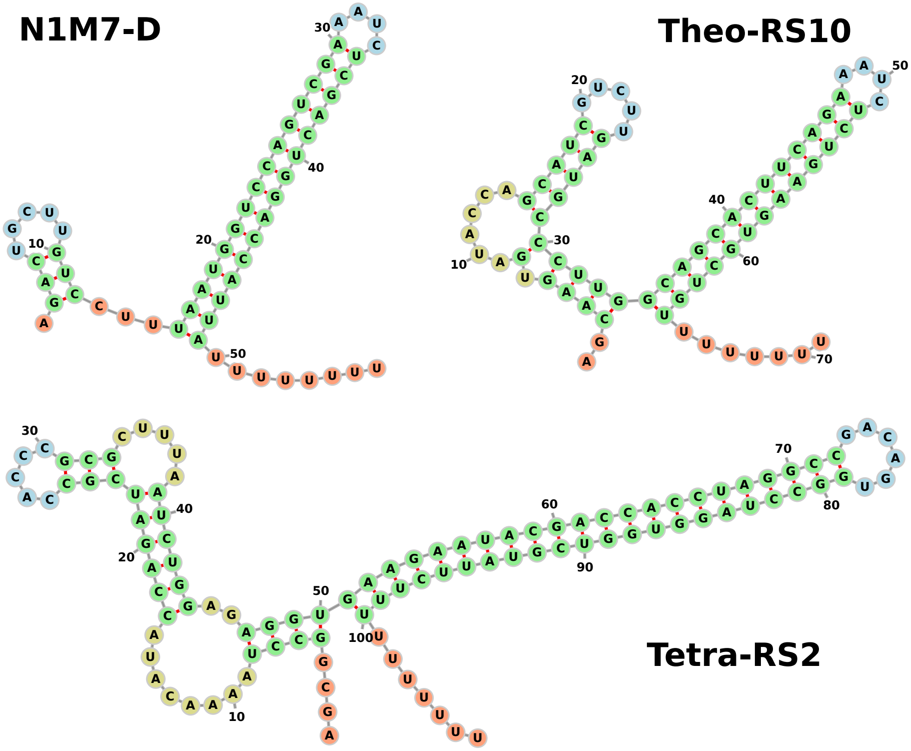


Figure 37: Minimum free energy secondary structures of the neomycin riboswitch N1M7-D, the theophylline riboswitch RS10 (Wachsmuth, Findeiß, et al., 2013), and the tetracycline riboswitch RS2 (Domin et al., 2017) drawn with *forna* (Kerpedjiev, Hammer, and Hofacker, 2015). Colors mark different structural features (hairpin loops in blue, interior loops and bulges in light brown, stems in green, and exterior loops in red). All riboswitches display a long terminator hairpin followed by an 8 nt poly-U stretch at their 3′ end. In contrast to the other constructs shown, N1M7-D exhibits a remarkably small structure at its 5′ end.

A.2 Supplemental data

Table 1: Overview of all constructs analysed in this work and which figures they appear in, along with their corresponding fluorescence intensity data (mean, standard deviation (SD), and the number of repetitions (N)). Sequence data can be found in a supplementary spreadsheet file.

		- neomycin			+ neomycin		
		mean	SD	N	mean	SD	N
Figure 4B	Pos	493.32	133.42	6			
	LH1-N1M7-A	54.77	2.96	3	63.86	4.61	3
	LH1-N1M7-B	712.27	100.31	3	812.05	109.73	3
	LH1-N1M7-C	528.68	108.21	3	963.45	83.19	3
	LH1-N1M7-D	342.28	21.67	3	899.39	121.33	3
	LH1-Term-N1M7-D	207.54	29.29	3	207.92	19.49	3
Figure 5C	Pos	703.90	33.10	3	676.98	33.92	3
	LH1-Pos	724.84	65.92	6	779.09	56.16	6
	N1M7-D	467.18	132.07	3	504.81	101.70	3
	LH1-N1M7-D	214.95	18.29	3	704.86	24.39	3
	LH2-N1M7-D	304.37	54.78	3	739.21	57.94	3
Figure 6C	Pos	662.93	79.58	3			
	U1-N1M7-D	121.50	4.84	3	189.22	24.45	3
	U2-N1M7-D	213.61	25.92	3	296.02	21.88	3
	LH1-U1-N1M7-D	525.68	153.12	3	880.53	195.61	3
	LH1-U2-N1M7-D	609.89	176.20	3	903.36	185.90	3
Figure 7B	LH1-Pos	691.41	19.95	3	762.86	55.21	3
	LH1-N1M7-D	214.95	18.29	3	704.86	24.39	3
	LM1-N1M7-D	193.40	4.10	3	665.74	144.60	3
	LM1C-N1M7-D	275.94	51.94	3	876.00	171.67	3
	LM2-N1M7-D	82.61	10.98	3	229.10	34.66	3
	LM2C-N1M7-D	244.70	3.76	3	803.76	248.15	3
Figure 7C	LH1-Pos	758.27	84.34	3	795.33	63.61	3
	LM1-Pos	690.96	46.96	3	707.80	50.35	3
	LM1C-Pos	855.49	90.97	3	835.45	76.11	3
	LM2-Pos	392.54	19.53	3	403.31	18.43	3
	LM2C-Pos	800.75	192.71	3	801.90	206.77	3
Figure A.3	LH1-Pos	595.96	134.81	6	638.34	163.03	6
	LH2-N1M7-D	304.37	54.78	3	739.21	57.94	3
	LM3-N1M7-D	574.30	60.17	3	833.45	116.35	3
	LM3C-N1M7-D	266.00	46.67	3	759.79	57.27	3
	LM4-N1M7-D	240.69	57.47	3	747.31	112.81	3
	LM5-N1M7-D	170.10	18.16	3	502.81	60.70	3
Figure A.4	LH1-N1M7-D	321.31	78.80	7	818.07	161.59	7
	oL-N1M7-D	102.61	16.15	3	114.33	17.41	3
	LH1-oL-N1M7-D	408.52	25.88	3	382.05	56.92	3
	LH2-oL-N1M7-D	218.60	23.81	3	194.26	27.54	3

Table 2: Effects of mutations on the stability of leader hairpins LH1 and LH2. The stability is given as MFE of the entire (mutated) hairpin. Activities of the resulting constructs are shown in Figure 32 and Figure 35.

Hairpin	Mutations	MFE (kcal mol^{-1})
LH1	none	-15.9
LM1	G28C, A29U	-10.9
LM1C	G28C, A29U, U6A, U7G	-18.5
LM2	U21C, C22G,	-9.7
LM2C	U21G, C22G, G13C, G14C	-18.3
LH2	none	-13.1
LM3	C17G, G18C	-5.5
LM3C	C17G, G18C, C3G, G4C	-13.9
LM4	G13C, A14U	-10.1
LM5	G13C, A14U, G15C	-5.3

Table 3: p-values of fluorescence activity in the presence and absence of neomycin for the riboswitch candidates with unstructured leaders depicted in Figure 31. They were calculated using a paired t-test from three biological replicates. The false discovery rate (FDR) for the four constructs was determined using the Benjamini–Hochberg procedure (Benjamini and Hochberg, 1995).
With p-values well below 3%, the difference between *on* and *off* state is statistically significant for the first three riboswitches. Only LH1-U2-N1M7-D has a p-value of 5.6% that is slightly above the commonly accepted significance threshold of 5%. Since the very similar LH1-U1-N1M7-D shows significant activity, however, we do believe that the construct acts as a neomycin-sensitive switch and that the high p-value is only a consequence of the low number of replicates. Since our conclusions follow already from the results for LH1-U1-N1M7-D, we refrained from further experiments.

Riboswitch	p-value	FDR
U1-N1M7-D	0.0272	0.0363
U2-N1M7-D	0.0061	0.0193
LH1-U1-N1M7-D	0.0097	0.0193
LH1-U2-N1M7-D	0.0561	0.0561

APPENDIX B

Supplemental Information: Assessing the quality of cotranscriptional folding models

B.1 Table of output files

Table 4 provides a list of all files generated by running the all-in-one processing script `barmap_gen_barmapfile`, which automatically performs a full *BarMap-QA* analysis without user interaction.

Table 4: Files that are generated when running `barmap_gen_barmapfile`. Curved arrows (↷) indicate line wraps introduced to make the table fit to the page. The notation .[a|b] means the file exists with both extensions a and b.

File	Description
*.bar	Standard output of Barriers containing information about the respective coarse-grained landscape, e. g., representative structure and the barrier height.
*.bar.log	Log file that summarizes the Barriers run.
*.rates.bin	Binary rate matrix of the respective landscape generated by Barriers.
*.evals.bin *.evecs.bin	Eigenvalues and -vectors of the respective rate matrix stored in a binary format. They are used in the final simulation step to speed up the calculation.
*.rates.bin.log	Log file of the diagonalization process performed by Treekin.
barmap.out	*BarMap*'s state mapping indicating exact (`->`) and approximate (`~>`) mappings between consecutive landscapes.
barmap.out.kin_t8_1e3	Full kinetics simulation output generated by multiple consecutive Treekin simulations over all landscapes.
barmap.out.kin_t8_1e3↷ .filt	Kinetics simulation output filtered only for highly populated states.
barmap.out.kin_t8_1e3↷ .filt.[pdf|svg]	Plot of the filtered Treekin simulation in SVG and PDF format.
barmap.out.kin_t8_1e3↷ .merge	Filtered and merged (cf. above) kinetics simulation output.
barmap.out.kin_t8_1e3↷ .merge.[pdf|svg]	Plot of the merged Treekin simulation in SVG and PDF format.

List of Abbreviations

B. clausii *Bacillus clausii.*

E. coli *Escherichia coli.*

bgaB the β-galactosidase gene.

egfp the enhanced green fluorescent protein gene.

AIC Akaike information criterion.

AUR aptamer upstream region.

BgaB β-galactosidase.

CvM Cramér–von Mises.

dL decoupling leader.

eGFP the enhanced green fluorescent protein.

GEV generalized extreme value.

lH leader hairpin.

MFE minimum free energy.

mRNA messenger RNA.

nt nucleotide.

oL original leader.

PCR polymerase chain reaction.

Pos positive control.

RBS ribosomal binding site.

RFU relative fluorescence unit.

RNAP RNA polymerase.

RNase E ribonuclease E.

RppH RNA $5'$ pyrophosphohydrolase.

rRNA ribosomal RNA.

SAM S-adenosyl-L-methionine.

SELEX systematic evolution of ligands by exponential enrichment.

TDR terminator downstream region.

TSS transcription start site.

U unstructured region.

UTR untranslated region.

YFP the yellow fluorescence protein.

Bibliography

Akaike, H. (Dec. 1974). “A new look at the statistical model identification”. In: *IEEE Transactions on Automatic Control* 19.6, pp. 716–723. DOI: `10.1109/TAC.1974.1100705`.

Akutsu, T. (Aug. 2000). “Dynamic programming algorithms for RNA secondary structure prediction with pseudoknots”. In: *Discrete Applied Mathematics* 104.1, pp. 45–62. DOI: `10.1016/S0166-218X(00)00186-4`.

Alberts, B. (2022). *Molecular biology of the cell*. Seventh edition. New York: W. W. Norton & Company. ISBN: 978-0-393-88482-1.

Ausländer, S., P. Stücheli, C. Rehm, D. Ausländer, J. S. Hartig, and M. Fussenegger (Nov. 2014). “A general design strategy for protein-responsive riboswitches in mammalian cells”. In: *Nature Methods* 11.11, pp. 1154–1160. DOI: `10.1038/nmeth.3136`.

Bannister, A. J. and T. Kouzarides (Mar. 2011). “Regulation of chromatin by histone modifications”. In: *Cell Research* 21.3, pp. 381–395. DOI: `10.1038/cr.2011.22`.

Basitta, P., L. Westrich, M. Rösch, A. Kulik, B. Gust, and A. K. Apel (2017). “AGOS: A Plug-and-Play Method for the Assembly of Artificial Gene Operons into Functional Biosynthetic Gene Clusters”. In: *ACS Synth. Biol.* 6 (5), pp. 817–825. DOI: `10.1021/acssynbio.6b00319`.

Becker, T., R. Herrmann, V. Sandor, D. Schäfer, and U. Wellisch (2016). *Stochastische Risikomodellierung und statistische Methoden: Ein anwendungsorientiertes Lehrbuch für Aktuare*. Berlin, Heidelberg: Springer. ISBN: 978-3-662-49406-6 978-3-662-49407-3. DOI: `10.1007/978-3-662-49407-3`.

Benjamini, Y. and Y. Hochberg (1995). “Controlling the false discovery rate: a practical and powerful approach to multiple testing”. In: *Journal of the Royal statistical society: series B (Methodological)* 57.1, pp. 289–300. DOI: `10.1111/j.2517-6161.1995.tb02031.x`.

Berman, H. M., J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne (2000). “The Protein Data Bank”. In: *Nucleic Acids Research* 28.1, pp. 235–242. DOI: `10.1093/nar/28.1.235`. URL: `www.rcsb.org`.

Bevilacqua, P. C. and J. M. Blose (2008). “Structures, kinetics, thermodynamics, and biological functions of RNA hairpins”. In: *Annu Rev Phys Chem* 59, pp. 79–103. DOI: `10.1146/annurev.physchem.59.032607.093743`.

Blom, G. (1958). “Statistical estimates and transformed beta-variables”. PhD thesis. Stockholm: Almqvist & Wiksell.

Bompfünewerer, A. F., R. Backofen, S. H. Bernhart, J. Hertel, I. L. Hofacker, P. F. Stadler, and S. Will (Nov. 21, 2007). “Variations on RNA folding and alignment: lessons from Benasque”. In: *Journal of Mathematical Biology* 56.1, pp. 129–144. DOI: `10.1007/s00285-007-0107-5`.

Breaker, R. R. (2012). “Riboswitches and the RNA world”. In: *Cold Spring Harbor perspectives in biology* 4.2. DOI: `10.1101/cshperspect.a003566`.

Chang, K.-Y. and I. Tinoco (May 1997). “The structure of an RNA “kissing” hairpin complex of the HIV TAR hairpin loop and its complement”. In: *Journal of Molecular Biology* 269.1, pp. 52–66. DOI: `10.1006/jmbi.1997.1021`.

Chen, Y.-J., P. Liu, A. A. K. Nielsen, J. A. N. Brophy, K. Clancy, T. Peterson, and C. A. Voigt (July 2013). “Characterization of 582 natural and synthetic terminators and quantification of their design constraints”. In: *Nature Methods* 10.7, pp. 659–664. DOI: `10.1038/nmeth.2515`.

Clote, P. (2006). “Combinatorics of saturated secondary structures of RNA”. In: *Journal of Computational Biology* 13.9, pp. 1640–1657. DOI: `10.1089/cmb.2006.13.1640`.

Clote, P., E. Kranakis, D. Krizanc, and B. Salvy (Oct. 1, 2009). “Asymptotics of canonical and saturated rna secondary structures”. In: *Journal of Bioinformatics and Computational Biology* 07.5, pp. 869–893. DOI: `10.1142/S0219720009004333`.

Clote, P., Y. Ponty, and J.-M. Steyaert (Sept. 2012). “Expected distance between terminal nucleotides of RNA secondary structures”. In: *Journal of Mathematical Biology* 65.3, pp. 581–599. DOI: `10.1007/s00285-011-0467-8`.

Cooksy, A. (2014). *Physical Chemistry: Thermodynamics, Statistical Mechanics & Kinetics*. International ed. Always learning. Boston: Pearson. ISBN: 978-0-321-81415-9.

Csörgő, S. and J. J. Faraway (1996). “The Exact and Asymptotic Distributions of Cramér-von Mises Statistics”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1, pp. 221–234. DOI: `10.1111/j.2517-6161.1996.tb02077.x`.

Day, L., O. A. E. Souki, A. A. Albrecht, and K. Steinhöfel (2016). “Random versus Deterministic Descent in RNA Energy Landscape Analysis”. In: *Advances in Bioinformatics* 2016, p. 9654921. DOI: `10.1155/2016/9654921`.

Deana, A., H. Celesnik, and J. G. Belasco (2008). “The bacterial enzyme RppH triggers messenger RNA degradation by 5’ pyrophosphate removal”. In: *Nature* 451.7176, p. 355. DOI: `10.1038/nature06475`.

Devi, G., Y. Zhou, Z. Zhong, D.-F. K. Toh, and G. Chen (2015). “RNA triplexes: from structural principles to biological and biotech applications”. In: *Wiley Interdisciplinary Reviews: RNA* 6.1, pp. 111–128. DOI: `10.1002/wrna.1261`.

Domin, G., S. Findeiß, M. Wachsmuth, S. Will, P. F. Stadler, and M. Mörl (Apr. 2017). “Applicability of a computational design approach for synthetic riboswitches”. In: *Nucleic Acids Res* 45.7, pp. 4108–4119. DOI: `10.1093/nar/gkw1267`.

Drew, H. R., R. M. Wing, T. Takano, C. Broka, S. Tanaka, K. Itakura, and R. E. Dickerson (1981). “Structure of a B-DNA dodecamer: conformation and dynamics”. In: *Proceedings of the National Academy of Sciences* 78.4, pp. 2179–2183. DOI: `10.1073/pnas.78.4.2179`.

Ellington, A. D. and J. W. Szostak (1990). “*In vitro* selection of RNA molecules that bind specific ligands”. In: *nature* 346.6287, p. 818.

Emory, S. A., P. Bouvet, and J. G. Belasco (Jan. 1992). “A 5’-terminal stem-loop structure can stabilize mRNA in *Escherichia coli.*” en. In: *Genes Dev.* 6.1, pp. 135–148. DOI: `10.1101/gad.6.1.135`.

Entzian, G. and M. Raden (Jan. 15, 2020). “pourRNA—a time- and memory-efficient approach for the guided exploration of RNA energy landscapes”. In: *Bioinformatics* 36.2, pp. 462–469. DOI: `10.1093/bioinformatics/btz583`.

Etzel, M. and M. Mörl (Mar. 2017). “Synthetic Riboswitches: From Plug and Pray toward Plug and Play”. In: *Biochemistry* 56.9, pp. 1181–1198. DOI: `10.1021/acs.biochem.6b01218`.

Evers, D. and R. Giegerich (2001). “Reducing the conformation space in RNA structure prediction”. In: *Proceedings of the German Conference on Bioinformatics*.

Findeiß, S., M. Wachsmuth, M. Mörl, and P. F. Stadler (2015). “Design of transcription regulating riboswitches”. In: *Methods Enzymol* 550, pp. 1–22. DOI: `10.1016/bs.mie.2014.10.029`.

Flamm, C., W. Fontana, I. L. Hofacker, and P. Schuster (2000). “RNA folding at elementary step resolution”. In: *RNA* 6.3, pp. 325–338. DOI: `10.1017/S1355838200992161`.

Flamm, C. and I. L. Hofacker (2008). “Beyond energy minimization: approaches to the kinetic folding of RNA”. In: *Chemical Monthly* 139, pp. 447–457.

Flamm, C., I. L. Hofacker, S. Maurer-Stroh, P. F. Stadler, and M. Zehl (2001). “Design of multistable RNA molecules”. In: *RNA* 7, pp. 254–265.

Flamm, C., I. L. Hofacker, P. F. Stadler, and M. T. Wolfinger (Jan. 2002). “Barrier Trees of Degenerate Landscapes”. In: *Zeitschrift für Physikalische Chemie* 216.2/2002. DOI: `10.1524/zpch.2002.216.2.155`.

Fletcher, R. (2008). *Practical methods of optimization.* 2. ed., reprinted in paperback, June 2008. A Wiley-Interscience publication. Chichester: Wiley. 436 pp. ISBN: 978-0-471-91547-8 978-0-471-49463-8.

Fontana, W., D. A. M. Konings, P. F. Stadler, and P. Schuster (Sept. 1993). “Statistics of RNA secondary structures”. In: *Biopolymers* 33.9, pp. 1389–1404. DOI: `10.1002/bip.360330909`.

Fontana, W., P. F. Stadler, E. G. Bornberg-Bauer, T. Griesmacher, I. L. Hofacker, M. Tacker, P. Tarazona, E. D. Weinberger, and P. Schuster (Mar. 1, 1993). “RNA folding and combinatory landscapes”. In: *Physical Review E* 47.3, pp. 2083–2099. DOI: `10.1103/PhysRevE.47.2083`.

Fowler, C. C., E. D. Brown, and Y. Li (2008). “A FACS-based approach to engineering artificial riboswitches.” In: *ChemBioChem* 9.12, pp. 1906–1911. DOI: `10.1002/cbic.200700713`.

Geis, M., C. Flamm, M. T. Wolfinger, A. Tanzer, I. L. Hofacker, M. Middendorf, C. Mandl, P. F. Stadler, and C. Thurner (2008). “Folding Kinetics of Large RNAs”. In: *Journal of Molecular Biology* 379.1, pp. 160–173. DOI: `10.1016/j.jmb.2008.02.064`.

Gerdes, K. and E. G. Wagner (2007). “RNA antitoxins”. In: *Current Opinion in Microbiology* 10, pp. 117–124. DOI: `10.1016/j.mib.2007.03.003`.

Giegerich, R., B. Voß, and M. Rehmsmeier (2004). “Abstract shapes of RNA”. In: *Nucleic acids research* 32.16, pp. 4843–4851. DOI: `10.1093/nar/gkh779`.

Gillespie, D. T. (Dec. 1, 1977). “Exact stochastic simulation of coupled chemical reactions”. In: *The Journal of Physical Chemistry* 81.25. Publisher: American Chemical Society, pp. 2340–2361. DOI: `10.1021/j100540a008`.

Green, A. A., P. A. Silver, J. J. Collins, and P. Yin (Nov. 2014). “Toehold Switches: De-Novo-Designed Regulators of Gene Expression”. In: *Cell* 159.4, pp. 925–939. DOI: `10.1016/j.cell.2014.10.002`.

Gruber, A. R., S. Findeiß, S. Washietl, I. L. Hofacker, and P. F. Stadler (2010). “RNAz 2.0: improved noncoding RNA detection”. In: *Biocomputing 2010*, pp. 69–79.

Grüner, W., R. Giegerich, D. Strothmann, C. Reidys, J. Weber, I. L. Hofacker, P. F. Stadler, and P. Schuster (Apr. 1996). “Analysis of RNA sequence structure maps by exhaustive enumeration I. Neutral networks”. In: *Chemical Monthly* 127.4, pp. 355–374. DOI: `10.1007/BF00810881`.

Günzel, C., F. Kühnl, K. Arnold, S. Findeiß, C. E. Weinberg, P. F. Stadler, and M. Mörl (2020). “Beyond Plug and Pray: Context Sensitivity and in silico Design of Artificial Neomycin Riboswitches”. In: *RNA Biology*, pp. 1–11. DOI: `10.1080/15476286.2020.1816336`.

Gusarov, I. and E. Nudler (1999). “The Mechanism of Intrinsic Transcription Termination”. In: *Molecular Cell* 3.4, pp. 495–504. DOI: `10.1016/S1097-2765(00)80477-3`.

Halder, S. and D. Bhattacharyya (Nov. 2013). “RNA structure and dynamics: A base pairing perspective”. In: *Progress in Biophysics and Molecular Biology* 113.2, pp. 264–283. DOI: `10.1016/j.pbiomolbio.2013.07.003`.

Hammer, S., Y. Ponty, W. Wang, and S. Will (Apr. 21, 2018). “Fixed-Parameter Tractable Sampling for RNA Design with Multiple Target Structures”. In: RECOMB 2018 – 22nd Annual International Conference on Research in Computational Molecular Biology.

Hammer, S., B. Tschiatschek, C. Flamm, I. L. Hofacker, and S. Findeiß (2017). “RNAblueprint: flexible multiple target nucleic acid sequence design”. In: *Bioinformatics* 33.18, pp. 2850–2858. DOI: `10.1093/bioinformatics/btx263`.

Hancock, J. T. (2017). *Cell signalling.* 4th ed. Oxford: Oxford University Press. ISBN: 978-0-19-965848-0.

Hofacker, I. L., C. Flamm, C. Heine, M. T. Wolfinger, G. Scheuermann, and P. F. Stadler (July 1, 2010). “BarMap: RNA folding on dynamic energy landscapes”. In: *RNA* 16.7, pp. 1308–1316. DOI: `10.1261/rna.2093310`.

Hofacker, I. L., W. Fontana, P. F. Stadler, L. S. Bonhoeffer, M. Tacker, and P. Schuster (Feb. 1994). “Fast Folding and Comparison of RNA Secondary Structures”. en. In: *Monatshefte für Chemie / Chemical Monthly* 125.2, pp. 167–188. DOI: `10.1007/BF00818163`.

Höner zu Siederdissen, C., S. Hammer, I. Abfalter, I. L. Hofacker, C. Flamm, and P. F. Stadler (2013). “Computational Design of RNAs with Complex Energy Landscapes”. en. In: *Biopolymers* 99.12, pp. 1124–1136. DOI: `10.1002/bip.22337`.

“IEEE Standard for Floating-Point Arithmetic” (2019). In: *IEEE Std 754-2019 (Revision of IEEE 754-2008)*, pp. 1–84. DOI: `10.1109/IEEESTD.2019.8766229`.

Jenison, R. D., S. C. Gill, A. Pardi, and B. Polisky (Mar. 1994). “High-resolution molecular discrimination by RNA”. In: *Science* 263.5152, pp. 1425–1429. DOI: `10.1126/science.7510417`.

Jiang, X., A. Diwa, and J. G. Belasco (2000). “Regions of RNase E Important for 5’-End-Dependent RNA Cleavage and Autoregulated Synthesis”. In: *Journal of Bacteriology* 182.9, pp. 2468–2475. DOI: `10.1128/JB.182.9.2468-2475.2000`.

Jones, C. P., G. Piszczek, and A. R. Ferré-D’Amaré (2019). “Isothermal Titration Calorimetry Measurements of Riboswitch–Ligand Interactions”. In:

Microcalorimetry of Biological Molecules. Ed. by E. Ennifar. Vol. 1964. New York, NY: Springer New York, pp. 75–87. DOI: `10.1007/978-1-4939-9179-2_6`.

Jucker, F. M., R. M. Phillips, S. A. McCallum, and A. Pardi (Mar. 2003). “Role of a heterogeneous free state in the formation of a specific RNA-theophylline complex.” In: *Biochemistry* 42.9, pp. 2560–2567. DOI: `10.1021/bi027103+`.

Kampen, N. G. v. (2007). *Stochastic Processes in Physics and Chemistry.* 3rd ed. North-Holland personal library. Amsterdam, The Netherlands Oxford, UK: Elsevier. ISBN: 978-0-444-52965-7.

Karamasioti, E., C. Lormeau, and J. Stelling (2017). “Computational design of biological circuits: putting parts into context”. In: *Mol. Systems Design Eng.* 2 (4), pp. 410–421. DOI: `10.1039/c7me00032d`.

Kelemen, O., P. Convertini, Z. Zhang, Y. Wen, M. Shen, M. Falaleeva, and S. Stamm (Feb. 1, 2013). “Function of alternative splicing”. In: *Gene* 514.1, pp. 1–30. DOI: `10.1016/j.gene.2012.07.083`.

Kerpedjiev, P., S. Hammer, and I. L. Hofacker (Oct. 15, 2015). “Forna (force-directed RNA): Simple and effective online RNA secondary structure diagrams”. In: *Bioinformatics* 31.20, pp. 3377–3379. DOI: `10.1093/bioinformatics/btv372`.

Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi (1983). “Optimization by simulated annealing”. In: *science* 220.4598, pp. 671–680.

Krajewski, S. S. and F. Narberhaus (Oct. 2014). “Temperature-driven differential gene expression by RNA thermosensors”. In: *Biochimica et Biophysica Acta – Gene Regulatory Mechanisms.* Riboswitches 1839.10, pp. 978–988. DOI: `10.1016/j.bbagrm.2014.03.006`.

Kucharík, M., I. L. Hofacker, P. F. Stadler, and J. Qin (July 15, 2014). “Basin Hopping Graph: a computational framework to characterize RNA folding landscapes”. In: *Bioinformatics* 30.14, pp. 2009–2017. ISSN: 1460-2059, 1367-4803. DOI: `10.1093/bioinformatics/btu156`.

Kühnl, F., P. F. Stadler, and S. Findeiß (2019). “Assessing the Quality of Cotranscriptional Folding Simulations”. In: *RNA Design.* Ed. by R. Lorenz. Methods in Molecular Biology. Manuscript accepted for publication. Berlin: Springer Nature.

Kurland, C. G. (June 1, 1960). “Molecular characterization of ribonucleic acid from Escherichia coli ribosomes: I. Isolation and molecular weights”. In: *Journal of Molecular Biology* 2.2, pp. 83–91. DOI: `10.1016/S0022-2836(60)80029-0`.

Lai, D., J. R. Proctor, and I. M. Meyer (2013). “On the importance of cotranscriptional RNA structure formation”. In: *RNA* 19, pp. 1461–1473. DOI: `10.1261/rna.037390.112`.

Li, Y. and S. Zhang (Mar. 21, 2012). “Predicting folding pathways between RNA conformational structures guided by RNA stacks”. In: *BMC Bioinformatics* 13.3, S5. ISSN: 1471-2105. DOI: `10.1186/1471-2105-13-S3-S5`.

Linnstaedt, S. D., W. K. Kasprzak, B. A. Shapiro, and J. L. Casey (Aug. 2006). “The role of a metastable RNA secondary structure in hepatitis delta virus genotype III RNA editing”. In: *RNA* 12.8, pp. 1521–1533. DOI: `10.1261/rna.89306`.

Lorenz, R., S. H. Bernhart, C. Höner zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, and I. L. Hofacker (Nov. 2011). “ViennaRNA Package 2.0”. In: *Algorithms for Molecular Biology* 6.1, p. 26. DOI: `10.1186/1748-7188-6-26`.

Lorenz, R., I. L. Hofacker, and P. F. Stadler (2016). “RNA folding with hard and soft constraints”. In: *Algorithms for Molecular Biology* 11.1. DOI: `10.1186/s13015-016-0070-z`.

Lutz, B., M. Faber, A. Verma, S. Klumpp, and A. Schug (2014). “Differences between cotranscriptional and free riboswitch folding”. In: *Nucleic Acids Res* 42, pp. 2687–2696. DOI: `10.1093/nar/gkt1213`.

Lyngsø, R. B. and C. N. Pedersen (2000). “RNA pseudoknot prediction in energy-based models”. In: *Journal of Computational Biology* 7.3, pp. 409–427. DOI: `10.1089/106652700750050862`.

Mackie, G. A. (1998). “Ribonuclease E is a 5’-end-dependent endonuclease”. In: *Nature* 395.6703, pp. 720–723. DOI: `10.1038/27246`.

Mandal, M., M. Lee, J. E. Barrick, Z. Weinberg, G. M. Emilsson, W. L. Ruzzo, and R. R. Breaker (2004). “A Glycine-Dependent Riboswitch That Uses Cooperative Binding to Control Gene Expression”. In: *Science* 306.5694, pp. 275–279. DOI: `10.1126/science.1100829`.

Mathews, D. H., D. H. Turner, and M. Zuker (2007). “RNA Secondary Structure Prediction”. In: *Current Protocols in Nucleic Acid Chemistry* 28.1, pp. 11.2.1–11.2.17. ISSN: 1934-9289. DOI: `10.1002/0471142700.nc1102s28`.

Matsumoto, M. and T. Nishimura (Jan. 1, 1998). “Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator”. In: *ACM Transactions on Modeling and Computer Simulation* 8.1, pp. 3–30. ISSN: 1049-3301. DOI: `10.1145/272991.272995`.

McCaskill, J. S. (1990). “The equilibrium partition function and base pair binding probabilities for RNA secondary structure”. In: *Biopolymers* 29.6, pp. 1105–1119. DOI: `10.1002/bip.360290621`.

Medema, M. H., R. Breitling, R. Bovenberg, and E. Takano (2011). “Exploiting plug-and-play synthetic biology for drug discovery and production in microorganisms”. In: *Nature Rev Microbiol* 9, pp. 131–137. DOI: `10.1038/nrmicro2478`.

Mehta, R. and W. S. Champney (Sept. 1, 2003). “Neomycin and Paromomycin Inhibit 30S Ribosomal Subunit Assembly in Staphylococcus aureus”. In: *Current Microbiology* 47.3, pp. 0237–0243. DOI: `10.1007/s00284-002-3945-9`.

Melnykov, A. V., R. K. Nayak, K. B. Hall, and A. Van Orden (2015). “Effect of loop composition on the stability and folding kinetics of RNA hairpins with large loops”. In: *Biochemistry* 54, pp. 1886–1896. DOI: `10.1021/bi5014276`.

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller (June 1953). “Equation of State Calculations by Fast Computing Machines”. In: *The Journal of Chemical Physics* 21.6, pp. 1087–1092. DOI: `10.1063/1.1699114`.

Mohan, S., C. Hsiao, H. VanDeusen, R. Gallagher, E. Krohn, B. Kalahar, R. M. Wartell, and L. D. Williams (Mar. 5, 2009). “Mechanism of RNA Double Helix-Propagation at Atomic Resolution”. In: *The Journal of Physical Chemistry B* 113.9, pp. 2614–2623. DOI: `10.1021/jp8039884`.

Møller-Jensen, J., T. Franch, and K. Gerdes (Sept. 2001). “Temporal Translational Control by a Metastable RNA Structure”. In: *Journal of Biological Chemistry* 276.38, pp. 35707–35713. DOI: `10.1074/jbc.M105347200`.

Montange, R. K. and R. T. Batey (2008). “Riboswitches: Emerging Themes in RNA Structure and Function”. In: *Annual Review of Biophysics* 37.1, pp. 117–133. DOI: `10.1146/annurev.biophys.37.032807.130000`.

Nahvi, A., N. Sudarsan, M. S. Ebert, X. Zou, K. L. Brown, and R. R. Breaker (2002). “Genetic control by a metabolite binding mRNA”. In: *Chemistry & Biology* 9.9, p. 1043.

Napierala, M. and W. J. Krzyzosiak (Dec. 1997). “CUG Repeats Present in Myotonin Kinase RNA Form Metastable "Slippery" Hairpins”. In: *Journal of Biological Chemistry* 272.49, pp. 31079–31085. DOI: `10.1074/jbc.272.49.31079`.

Nelder, J. A. and R. Mead (Jan. 1, 1965). “A Simplex Method for Function Minimization”. In: *The Computer Journal* 7.4, pp. 308–313. DOI: `10.1093/comjnl/7.4.308`.

Nielsen, S., Y. Yuzenkova, and N. Zenkin (June 28, 2013). “Mechanism of Eukaryotic RNA Polymerase III Transcription Termination”. In: *Science* 340.6140, pp. 1577–1580. DOI: `10.1126/science.1237934`.

Nussinov, R., G. Pieczenik, J. R. Griggs, and D. J. Kleitman (July 1, 1978). “Algorithms for Loop Matchings”. In: *SIAM Journal on Applied Mathematics* 35.1, pp. 68–82. DOI: `10.1137/0135006`.

Onoa, B. and I. Tinoco Jr (2004). “RNA folding and unfolding”. In: *Current Opinion in Structural Biology* 14.3, pp. 374–379. DOI: `10.1016/j.sbi.2004.04.001`.

Ouellet, J. (June 28, 2016). “RNA Fluorescence with Light-Up Aptamers”. In: *Frontiers in Chemistry* 4. DOI: `10.3389/fchem.2016.00029`.

Peters, J. M., A. D. Vangeloff, and R. Landick (2011). “Bacterial transcription terminators: the RNA 3’-end chronicles”. In: *Journal of Molecular Biology* 412.5, pp. 793–813. DOI: `10.1016/j.jmb.2011.03.036`.

Picard, F., C. Dressaire, L. Girbal, and M. Cocaign-Bousquet (Nov. 2009). “Examination of post-transcriptional regulations in prokaryotes by integrative biology”. In: *Comptes Rendus Biologies* 332.11, pp. 958–973. DOI: `10.1016/j.crvi.2009.09.005`.

Pörschke, D., O. C. Uhlenbeck, and F. H. Martin (1973). “Thermodynamics and kinetics of the helix–coil transition of oligomers containing GC base pairs”. In: *Biopolymers: Original Research on Biomolecules* 12.6, pp. 1313–1335.

Price, I. R., J. C. Grigg, and A. Ke (Oct. 1, 2014). “Common themes and differences in SAM recognition among SAM riboswitches”. In: *Biochimica et Biophysica Acta (BBA) - Gene Regulatory Mechanisms*. Riboswitches 1839.10, pp. 931–938. DOI: `10.1016/j.bbagrm.2014.05.013`.

Puton, T., L. P. Kozlowski, K. M. Rother, and J. M. Bujnicki (Apr. 1, 2013). “CompaRNA: a server for continuous benchmarking of automated methods for RNA secondary structure prediction”. In: *Nucleic Acids Research* 41.7, pp. 4307–4323. DOI: `10.1093/nar/gkt101`.

Quan, S., O. Skovgaard, R. E. McLaughlin, E. T. Buurman, and C. L. Squires (2015). “Markerless *Escherichia coli* rrn Deletion Strains for Genetic Determination of Ribosomal Binding Sites”. In: *G3 (Bethesda, Md.)* 5.12, pp. 2555–2557. DOI: `10.1534/g3.115.022301`.

Quarta, G., K. Sin, and T. Schlick (Feb. 16, 2012). “Dynamic Energy Landscapes of Riboswitches Help Interpret Conformational Rearrangements and Function”. In: *PLOS Computational Biology* 8.2, e1002368. DOI: `10.1371/journal.pcbi.1002368`.

Ramesh, A., C. A. Wakeman, and W. C. Winkler (2011). “Insights into metalloregulation by M-box riboswitch RNAs via structural analysis of manganese-bound complexes". In: *Journal of molecular biology* 407.4, pp. 556–570. DOI: `10.1016/j.jmb.2011.01.049`.

Ray-Soni, A., M. J. Bellecourt, and R. Landick (June 2016). "Mechanisms of Bacterial Transcription Termination: All Good Things Must End". In: *Annual Review of Biochemistry* 85.1, pp. 319–347. DOI: `10.1146/annurev-biochem-060815-014844`.

Riley, K. E. and P. Hobza (2013). "On the importance and origin of aromatic interactions in chemistry and biodisciplines". In: *Accounts of Chemical Research* 46.4, pp. 927–936. DOI: `10.1021/ar300083h`.

Roth, A. and R. R. Breaker (June 1, 2009). "The Structural and Functional Diversity of Metabolite-Binding Riboswitches". In: *Annual Review of Biochemistry* 78.1, pp. 305–334. DOI: `10.1146/annurev.biochem.78.070507.135656`.

Roth, A., W. C. Winkler, et al. (2007). "A riboswitch selective for the queuosine precursor preQ$_1$ contains an unusually small aptamer domain". In: *Nature structural & molecular biology* 14.4, pp. 308–317. DOI: `10.1038/nsmb1224`.

Royston, J. P. (1982). "Algorithm AS 177: Expected Normal Order Statistics (Exact and Approximate)". In: *Applied Statistics* 31.2, p. 161. ISSN: 00359254. DOI: `10.2307/2347982`.

Ryals, J., R. Little, and H. Bremer (Aug. 1, 1982). "Temperature dependence of RNA synthesis parameters in *Escherichia coli*." In: *Journal of Bacteriology* 151.2, pp. 879–887.

Sahu, S., R. Roy, and R. Anand (Mar. 25, 2022). "Harnessing the Potential of Biological Recognition Elements for Water Pollution Monitoring". In: *ACS Sensors* 7.3. Publisher: American Chemical Society, pp. 704–715. DOI: `10.1021/acssensors.1c02579`.

Schnall-Levin, M., L. Chindelevitch, and B. Berger (2008). "Inverting the Viterbi algorithm: an abstract framework for structure design". In: *Proceedings of the 25th international conference on Machine learning - ICML '08*. the 25th international conference. Helsinki, Finland: ACM Press, pp. 904–911. DOI: `10.1145/1390156.1390270`.

Schuster, P., W. Fontana, P. F. Stadler, and I. L. Hofacker (Mar. 22, 1994). "From Sequences to Shapes and Back: A Case Study in RNA Secondary Structures". In: *Proceedings of the Royal Society of London B: Biological Sciences* 255.1344, pp. 279–284. DOI: `10.1098/rspb.1994.0040`.

Serganov, A. and E. Nudler (2013). "A decade of riboswitches". In: *Cell* 152.1-2, pp. 17–24. DOI: `10.1016/j.cell.2012.12.024`.

Sharma, V., Y. Nomura, and Y. Yokobayashi (Dec. 3, 2008). "Engineering Complex Riboswitch Regulation by Dual Genetic Selection". In: *Journal of the American Chemical Society* 130.48, pp. 16310–16315. ISSN: 0002-7863. DOI: `10.1021/ja805203w`.

Singal, R. and G. D. Ginder (June 15, 1999). "DNA Methylation". In: *Blood* 93.12, pp. 4059–4070. DOI: `10.1182/blood.V93.12.4059`.

Smale, S. T. (May 1, 2010). "Galactosidase Assay". In: *Cold Spring Harbor Protocols* 2010.5, pdb.prot5423–pdb.prot5423. DOI: `10.1101/pdb.prot5423`.

Stadler, B. M. R. and P. F. Stadler (Dec. 2010). "Combinatorial vector fields and the valley structure of fitness landscapes". In: *Journal of Mathematical Biology* 61.6, pp. 877–898. DOI: `10.1007/s00285-010-0326-z`.

Stein, P. R. and M. S. Waterman (Jan. 1, 1979). "On some new sequences generalizing the Catalan and Motzkin numbers". In: *Discrete Mathematics* 26.3, pp. 261–272. DOI: `10.1016/0012-365X(79)90033-5`.

Sudarsan, N., M. C. Hammond, K. F. Block, R. Welz, J. E. Barrick, A. Roth, and R. R. Breaker (Oct. 13, 2006). “Tandem Riboswitch Architectures Exhibit Complex Gene Control Functions”. In: *Science* 314.5797, pp. 300–304. DOI: `10.1126/science.1130716`.

Suzuki, T. (2021). “The expanding world of tRNA modifications and their disease relevance”. In: *Nature reviews Molecular cell biology* 22.6, pp. 375–392. DOI: `10.1038/s41580-021-00342-0`.

Tacker, M., P. F. Stadler, E. G. Bornberg-Bauer, I. L. Hofacker, and P. Schuster (Dec. 1, 1996). “Algorithm independent properties of RNA secondary structure predictions”. In: *European Biophysics Journal* 25.2, pp. 115–130. DOI: `10.1007/s002490050023`.

Tang, X., S. Thomas, L. Tapia, D. P. Giedroc, and N. M. Amato (Sept. 12, 2008). “Simulating RNA Folding Kinetics on Approximated Energy Landscapes”. In: *Journal of Molecular Biology* 381.4, pp. 1055–1067. DOI: `10.1016/j.jmb.2008.02.007`.

The Grace contributors (2015). *Grace – a WYSIWYG 2D plotting tool for the X Window System and M*tif.* [Online; accessed 10-September-2019]. URL: `http://plasma-gate.weizmann.ac.il/Grace/`.

Tijsterman, M. and R. H. A. Plasterk (Apr. 2, 2004). “Dicers at RISC: The Mechanism of RNAi”. In: *Cell* 117.1, pp. 1–3. DOI: `10.1016/S0092-8674(04)00293-4`.

Timmers, H. T. M. and L. Tora (Oct. 4, 2018). “Transcript Buffering: A Balancing Act between mRNA Synthesis and mRNA Degradation”. In: *Molecular Cell* 72.1, pp. 10–17. DOI: `10.1016/j.molcel.2018.08.023`.

Trayhurn, P. (Mar. 1996). “Northern blotting”. In: *Proceedings of the Nutrition Society* 55.1, pp. 583–589. DOI: `10.1079/PNS19960051`.

Tuerk, C. and L. Gold (1990). “Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase”. In: *Science* 249.4968, pp. 505–510.

Turner, D. H. and D. H. Mathews (2010). “NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure.” In: *Nucleic Acids Res* 38.Database issue, pp. D280–D282. DOI: `10.1093/nar/gkp892`.

Villa, J. K., Y. Su, L. M. Contreras, and M. C. Hammond (2019). “Synthetic Biology of Small RNAs and Riboswitches”. In: *Regulating with RNA in Bacteria and Archaea.* Vol. 6. 3. American Society of Microbiology, pp. 527–545. DOI: `10.1128/microbiolspec.RWR-0007-2017`.

Vogel, U. and K. F. Jensen (May 1, 1994). “The RNA chain elongation rate in *Escherichia coli* depends on the growth rate.” In: *Journal of Bacteriology* 176.10, pp. 2807–2813. DOI: `10.1128/jb.176.10.2807-2813.1994`.

Wachsmuth, M., G. Domin, R. Lorenz, R. Serfling, S. Findeiß, P. F. Stadler, and M. Mörl (Feb. 2015). “Design criteria for synthetic riboswitches acting on transcription”. In: *RNA Biol* 12.2, pp. 221–231. DOI: `10.1080/15476286.2015.1017235`.

Wachsmuth, M., S. Findeiß, N. Weissheimer, P. F. Stadler, and M. Mörl (Feb. 2013). “*De novo* design of a synthetic riboswitch that regulates transcription termination”. In: *Nucleic Acids Research* 41.4, pp. 2541–2551. DOI: `10.1093/nar/gks1330`.

Wachter, A. (Jan. 1, 2010). “Riboswitch-mediated control of gene expression in eukaryotes”. In: *RNA Biology* 7.1, pp. 67–76. DOI: `10.4161/rna.7.1.10489`.

Waksman, S. A. and H. A. Lechevalier (1949). “Neomycin, a New Antibiotic Active against Streptomycin-Resistant Bacteria, Including Tuberculosis Organisms”. In: *Science* 109.2830, pp. 305–307. URL: `https://www.jstor.org/stable/1677311`.

Wallis, M. G., U. von Ahsen, R. Schroeder, and M. Famulok (1995). “A novel RNA motif for neomycin recognition”. In: *Chemistry & Biology* 2.8, pp. 543–552. DOI: `10.1016/1074-5521(95)90188-4`.

Washietl, S., I. L. Hofacker, and P. F. Stadler (Feb. 15, 2005). “Fast and reliable prediction of noncoding RNAs”. In: *Proceedings of the National Academy of Sciences of the United States of America* 102.7, pp. 2454–2459. DOI: `10.1073/pnas.0409169102`.

Washietl, S., I. L. Hofacker, P. F. Stadler, and M. Kellis (May 2012). “RNA folding with soft constraints: reconciliation of probing data and thermodynamic secondary structure prediction”. In: *Nucleic Acids Research* 40.10, pp. 4261–4272. DOI: `10.1093/nar/gks009`.

Weigand, J. E., M. Sanchez, E.-B. Gunnesch, S. Zeiher, R. Schroeder, and B. Suess (2008). “Screening for engineered neomycin riboswitches that control translation initiation”. In: *RNA* 14.1, pp. 89–97. DOI: `10.1261/rna.772408`.

Weigand, J. E., S. R. Schmidtke, T. J. Will, E. Duchardt-Ferner, C. Hammann, J. Wohnert, and B. Suess (2011). “Mechanistic insights into an engineered riboswitch: a switching element which confers riboswitch activity”. In: *Nucleic Acids Research* 39.8, pp. 3363–3372. DOI: `10.1093/nar/gkq946`.

Wieland, M. and J. S. Hartig (Sept. 4, 2006). “Turning Inhibitors into Activators: A Hammerhead Ribozyme Controlled by a Guanine Quadruplex”. In: *Angewandte Chemie International Edition* 45.35, pp. 5875–5878. DOI: `10.1002/anie.200600909`.

Wilson, K. S. and P. H. v. Hippel (Sept. 12, 1995). “Transcription termination at intrinsic terminators: the role of the RNA hairpin”. In: *Proceedings of the National Academy of Sciences* 92.19, pp. 8793–8797.

Wolfinger, M. T., W. A. Svrcek-Seiler, C. Flamm, I. L. Hofacker, and P. F. Stadler (Apr. 2004). “Efficient computation of RNA folding dynamics”. In: *Journal of Physics A: Mathematical and General* 37.17, pp. 4731–4741. DOI: `10.1088/0305-4470/37/17/005`.

Wolfsheimer, S. and A. K. Hartmann (Aug. 3, 2010). “Minimum-free-energy distribution of RNA secondary structures: Entropic and thermodynamic properties of rare events”. In: *Physical Review E* 82.2, p. 021902. DOI: `10.1103/PhysRevE.82.021902`.

Wuchty, S., W. Fontana, I. L. Hofacker, and P. Schuster (Feb. 1999). “Complete suboptimal folding of RNA and the stability of secondary structures.” In: *Biopolymers* 49.2, pp. 145–65. DOI: `10.1002/(SICI)1097-0282(199902)49:2$<$145::AID-BIP4$>$3.0.CO;2-G`.

Xayaphoummine, A., T. Bucher, and H. Isambert (July 2005). “Kinefold web server for RNA/DNA folding path and structure prediction including pseudoknots and knots”. In: *Nucleic Acids Research* 33.suppl-2, W605–W610. DOI: `10.1093/nar/gki447`.

Zhu, C., R. H. Byrd, P. Lu, and J. Nocedal (Dec. 1997). “Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization”. In: *ACM Transactions on Mathematical Software* 23.4, pp. 550–560. DOI: `10.1145/279232.279236`.

Zuker, M. (1989). “On finding all suboptimal foldings of an RNA molecule”. In: *Science* 244.4900, pp. 48–52. DOI: `10.1126/science.2468181`.

Zuker, M. and P. Stiegler (Jan. 10, 1981). “Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information”. In: *Nucleic Acids Research* 9.1, pp. 133–148. DOI: `10.1093/nar/9.1.133`.

Selbstständigkeitserklärung

Hiermit erkläre ich, die vorliegende Dissertation selbstständig und ohne unzulässige fremde Hilfe angefertigt zu haben. Ich habe keine anderen als die angeführten Quellen und Hilfsmittel benutzt und sämtliche Textstellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen wurden, und alle Angaben, die auf mündlichen Auskünften beruhen, als solche kenntlich gemacht. Ebenfalls sind alle von anderen Personen bereitgestellten Materialien oder erbrachten Dienstleistungen als solche gekennzeichnet.

Ort, Datum

Unterschrift