FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

> APPLIED LIFE SCIENCES

FACHBEREICH:
> BIOENGINEERING

RNA Energy Landscapes And Their Impact on Folding Kinetics

Diplomarbeit

zur Erlangung des akademischen Grades
Master of Science in Engineering
der
Fachhochschule Campus Wien
Master-Studiengang Bioinformatik

Vorgelegt von:
Werner Gradwohl
Personenkennzeichen: c1210542013

FH-Hauptbetreuer:
Mag. Dr. Michael Wolfinger

Zweitprüfer:
Dr. Anton Beyer

Abgabetermin: 15.09.2014

Erklärung:

Ich erkläre, dass die vorliegende Diplomarbeit von mir selbst verfasst wurde und ich keine anderen als die angeführten Behelfe verwendet bzw. mich auch sonst keiner unerlaubter Hilfe bedient habe.

Ich versichere, dass ich diese Diplomarbeit bisher weder im In- noch im Ausland (einer Beurteilerin/einem Beurteiler zur Begutachtung) in irgendeiner Form als Prüfungsarbeit vorgelegt habe.

Weiters versichere ich, dass die von mir eingereichten Exemplare (ausgedruckt und elektronisch) identisch sind.

Datum: ..... Unterschrift: .....

Abstract

The folding kinetics of RNA and other biomolecules can be described as an stochastic process in the discrete state space of an energy landscape, which incorporates both the thermodynamic and the kinetic view of a molecule. The thermodynamic view is a result of statistical mechanics, whereby the different states of the molecule are seen as gaseous particles. The kinetic view introduces a kinetic reaction coordinate in the system, because the molecule can under normal conditions not switch from one state to any other without passing through intermediate states. The folding landscape reflects these restrictions by defining a topological neighborhood, resulting in a hyper surface. Adding the energy parameter results in an energy landscape, which allows for investigating the folding of the molecule as trajectory on its hyper surface.
In the case of biomolecules structural changes are called folding kinetics. The temporal development of a molecule is simulated as a stochastic process, whereby the probabilities to occupy particular states at a particular time are calculated. Several algorithms have been developed to perform this task. The problem is that the execution of the computational implementation of these algorithms requires much time and memory, so that it is almost impossible to predict the folding kinetics of larger biomolecules. A first step was to partition the conformational space (as it deals with probabilities) into macro states, which represent not only one but many micro states, with the assumption that it is sufficient to describe the folding kinetics as transitions between such macro states (coarse graining). Although the partitioning of the state space into for example gradient basins as macro-states results in a remarkable reduction of computational effort to predict the folding process, it is still impossible to calculate the dynamics of larger RNA molecules. Therefore it is necessary to investigate particular properties of this model and evaluate their accordance with the dynamics of the folding process. The main task of this work will be to define exclusion (or inclusion) criteria based on topological properties of the energy landscape, resulting in further reduction of the conformation space (enhanced coarse graining) and evaluate correlations of these properties with the folding process to improve and/or enable prediction of larger RNA molecules. In particular we will compare the investigated features with the *states of interest*[19].

Kurzfassung

Die Faltungskinetik von RNA und anderen Biomolekülen kann als stochastischer Prozeß im diskreten Phasenraum einer Energielandschaft beschrieben werden, was sowohl die thermodynamische als auch die kinetische Sichtweise beinhaltet. Die thermodynamische Sichtweise ist ein Ergebnis der statistischen Mechanik, wobei die verschieden Zustände des Moleküls als gasförmige Teilchen betrachtet werden. Die kinetische Sichtweise führt eine kinetische Reaktionskoordinate ein, da das Molekül unter normalen Bedingungen nicht von einem in einen anderen Zustand wechseln kann, ohne dabei Zwischenzustände zu durchlaufen. Die Faltungslandschaft spiegelt diese Einschränkungen wider indem sie eine topologische Nachbarschaft definiert, woraus eine Hyperfläche resultiert. Das Hinzufügen des Energieparameters führt zu einer Energielandschaft, was uns die Untersuchung der Molekülfaltung als Trajektorie auf der Hyperfläche ermöglicht.
Im Fall von Biomolekülen werden strukturelle Veränderungen als Faltungskinetik bezeichnet. Die zeitliche Entwicklung eines Moleküls wird als stochastischer Prozeß simuliert, wobei die Wahrscheinlichkeiten einen bestimmten Zustand zu einer bestimmten Zeit einzunehmen berechnet werden. Um diese Aufgabe zu bewältigen wurden mehrere Algorithmen entwickelt. Das Problem ist, dass die Ausführung der Implementation dieser Algorithmen auf Computern viel Zeit und Speicherbedarf beansprucht, weshalb es beinahe unmöglich ist die Faltungskinetik größerer Biomoleküle vorherzusagen. Ein erster Schritt bestand in der Partitionierung des Konformationsraums (da er mit Wahrscheinlichkeiten operiert) in Makrozustände, welche nicht einen sondern viele Mikrozustände repräsentieren, unter der Annahme dass es genügt den Faltungsprozess als Übergänge zwischen solchen Makrozuständen zu beschreiben ("gröbere Auflösung"). Obwohl die Partitionierung des Phasenraumes in beispielsweise Gradientbasins als Makrozustände zu einer beachtlichen Reduktion des rechnerischen Aufwands der Vorhersage des Faltungsprozesses geführt hat, ist es immer noch nicht möglich die Dynamik größerer RNA-Moleküle zu berechnen. Deshalb ist es notwendig bestimmte Eigenschaften dieses Models zu untersuchen und ihrer Übereinstimmung mit der Dynamik des Faltungsprozesses zu evaluieren. Die Hauptaufgabe dieser Arbeit wird darin bestehen Ausschlußkriterien (oder Einschlußkriterien) basierend auf den topologischen Eigenschaften der Energielandschaft zu definieren, was zu einer weiteren Reduktion des Konformationsraumes führen wird ("verbesserte grobe Auflösung"), und die Korrelation dieser Eigenschaften mit dem Faltungsprozeß zu evaluieren um die Vorhersage größerer RNA-Moleküle zu verbesseren und/oder zu ermöglichen. Insbesondere werden die untersuchten Eigenschaften mit den *states of interest*[19] verglichen.

Table of contents

1 Introduction

1.1 The importance of structure

The function of all kind of machinery (be it physical, chemical or biological) operating at a level above its functional representation is determined by its structure. The only system not depending on its structure, is structure itself, namely the subatomic particles, waves and fields. There is no structure of structure, because this would not only be an infinite regress, but also require a new quality of meaning. Function does not require the change of structure, it is the change of structure. The structural foundation in which these changes occur need for their part be able to map the functionality of the systems in an executable way. The structural level of such base systems need not only be suitable to represent the complexity required to perform system maintaining actions, but also not too error prone on the mapping level. Such levels seem to be far above the "structure" of the universe, although it is impossible to measure this position from a non anthropocentric view.
If an executable system changes its state, it changes its structure. A different structure results in different functionality, which is again able to change the structure and so on. A completely unchangeable structure cannot interact with the environment (at least not beyond its quantum uncertainty). Since structure is so important, it seems natural, that more stable systems such as living organisms, have found a way to preserve their structure. The basic functionality of all stable systems is the ability to transport the system in space and time. This complex task seems to be impossible, since any action of the system requires its manipulation. The contradiction is avoided by performing only changes which further stabilize the system. However, this does not work always perfectly, because no system has direct access to its mapping system (and many have not even indirect access). If systems tend to stay functional for longer, they must have developed particular functionalities to do this. A wide spread method is to use "backup systems", which means that a system can rebuild its functionality of its own as long as the functionality to do this is working. The only way to create functional structures is to use other structures as model. These model structures are often non functional (on the operating level of the system), resulting in the remarkable ability to preserve the "structure" of structure by use of structure not as functionality representation, but as a code to control the formation of other structures - structure as information.
To clarify things it must be said, that we are always talking about coded information, not about information as such. This definition problem is analog to the "structure of structure"-problem. There is no encoded information in structure, structure is the encoded information. The nature of non encoded information (if it exists) is currently subject of research.
For our considerations we always mean encoded information, when using the term information.
Although the changes and development of a systems could be seen as information process, they are all structural processes. In this sense the use of structure as information storage by decoding the information is nothing more than a structural transformation occurring during the permanent transport of the system. Since we should not argument teleological, such informational structures could be the result of accidental transformations. And since the structural information (depending on the interacting system) remains the same, the transformation can happen in both directions. Systems will only survive if the control these transformations to stay as long as possible in a functional state (if they do not, it is not probable to encounter them).
System with the ability to partly rebuild themselves are already quite sophisticated. But with the ability to transform in principle all structures into structural information, the step from rebuilding to reproduction is not so unlikely. From an evolutionary point of view such a system has become almost immortal, although its stability is only phylogenetical, not ontogenetical. The problem might be, that the complexity of a system correlates with its error possibilities.

Since systems are not isolated, they might mutually influence their development, which results finally in sexual reproduction. The information stored in a system might not only encode its own functionality. Depending on the definition of the system-borders, different systems will have different experiences, which can then be interchanged and incorporated in further adaption processes of all individual systems. Those type of interchanges could have also played an important role in the origin of self organizing systems, whereby the individual parts were just chemical reactions, until the system developed new structural mappings and functionalities (theory of the hyper cycle [7]).

1.2 Functionality of RNA in living organisms

Structures play two important roles in the world - functionality and information storage.
Since this proposition does not only apply to all kind of computing machinery (computers in the strictest sense), but also to all living organisms, the structure of biomolecules is essential for their functionality. Analyses have shown, that (functional) structure is much more conserved than sequence (informational structure). We observe a relation between structure and functionality, which seems to have also been a crucial step in biochemical evolution in the form of back coupling of the phenotype on the genotype [7].

The three big groups of biomolecules operating in living organisms (proteins, DNA, RNA) all fulfill this relation.

	RNA	**DNA**	**Protein**
Genetic Information	yes	yes	no
Catalysis	yes	no	yes
Regulation/Interactions	yes	yes	yes
Structure	Usually singe stranded	Usually duplex	Different

Table 1: Functions of biomolecules

As seen in table 1 only RNA has the ability to work in all fields. It has been thought for decades, that RNA cannot act catalytically or regulative. Its only purpose seemed to be an information transporter. Although this is a very useful feature, because it uncouples the functional from the informal structure, it is more complicated. Effective systems do not need to be big and complicated, so we can neither say if the current distribution of information is the result of a long or short development, nor if future systems will have less or even more biomolecules for information processing.
RNA has a special position among the biomolecules. Despite its comparatively simple structure it overtakes many tasks, what makes it very interesting and suitable for research.

The only coding RNA is the messenger RNA. Its task is to transport genetic information from the DNA to the ribosome, where proteins are built. All other RNAs are non-coding in the sense that the do not carry genetic information. However, transfer RNAs provide the ribosome with the amino acids they encode, according to the mRNA sequence read by the ribosome during the translation. The ribosome itself is built of ribosomal RNAs and proteins, whereby the active sites consist of rRNA (an example of RNA acting catalytic). Ribosomal RNA has been found in all cells and is therefor of special importance for phylogenetical studies.
Many other types of RNA are participating in such important processes as splicing (small nuclear RNA) or gene regulation (small interfering RNA, micro RNA).

2 Modeling RNA

2.1 Structural levels of RNA

The RNA molecule itself is often described on different structural levels. In the case of RNA the *primary structure* represents the sequence of ribonucleotides as a one-dimensional string. It includes no real information about the spatial formation itself. Their chemical properties determine the pairing possibilities and hence the formation possibilities of the RNA molecule. The chemical building blocks of RNA are the ribonucleotides adenosine monophosphate, cytidine monophosphate, guanosine monophosphate and uridine monophosphate.
The *secondary structure* contains information about base pairing (defining the topological shape of the structure). It also includes no real information about the actual spatial formation, but many restrictions due to the pairing abilities of the nucleobases. Those hydrogen bonds are normally established between cytosine-guanine, adenine-uracil and guanine-uracil.
Secondary structure can be represented in many ways, for instance as dot-bracket-string or circular graph (see figure 1).

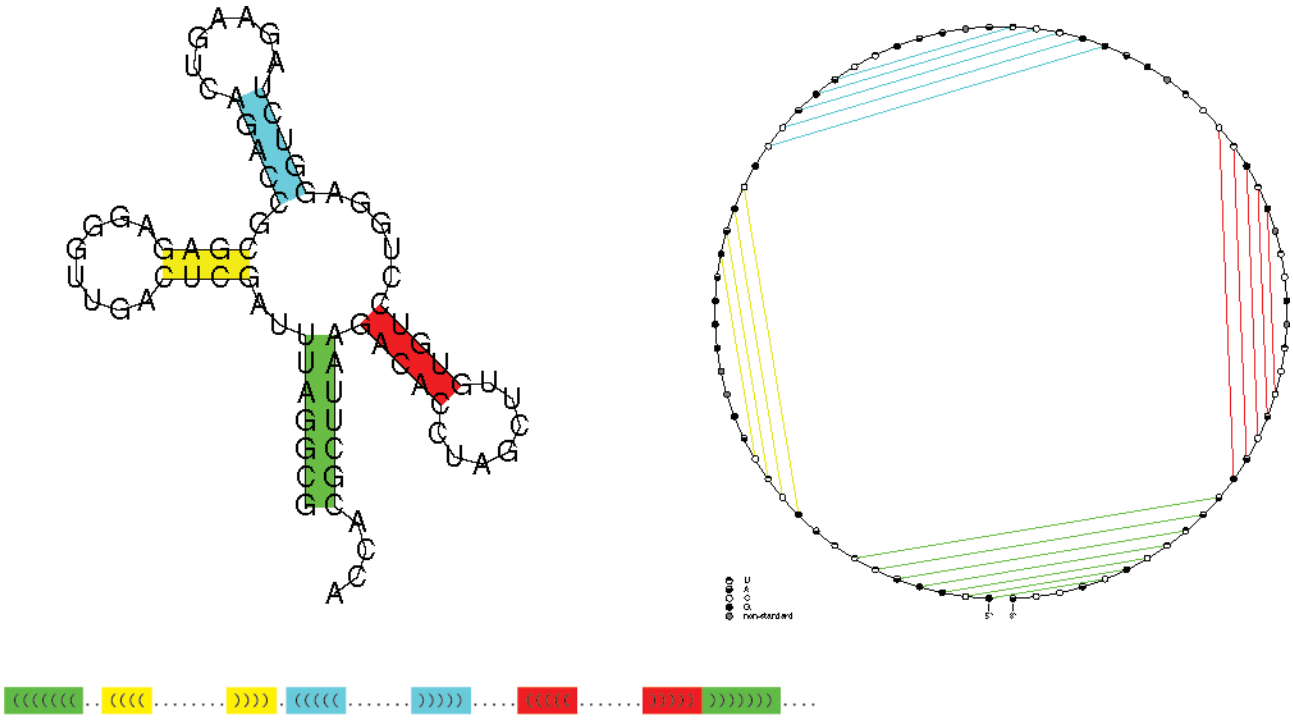

Figure 1: Secondary structure of phenylalanine tRNA from yeast as conventional drawing (left), in circular representation (right) and as dot-bracket string (bottom). Same colors represent the same base-pairing regions. The chords in the circular representation must not cross in secondary structure graphs. (Adapted from [34]).

The *tertiary structure* of a nucleic acid is its precise three-dimensional structure. It can be represented by the positions of the involved atoms in a coordinate system and visualized for example as ball-and-stick model (see figure 2) giving a impression of how the molecule might "look".

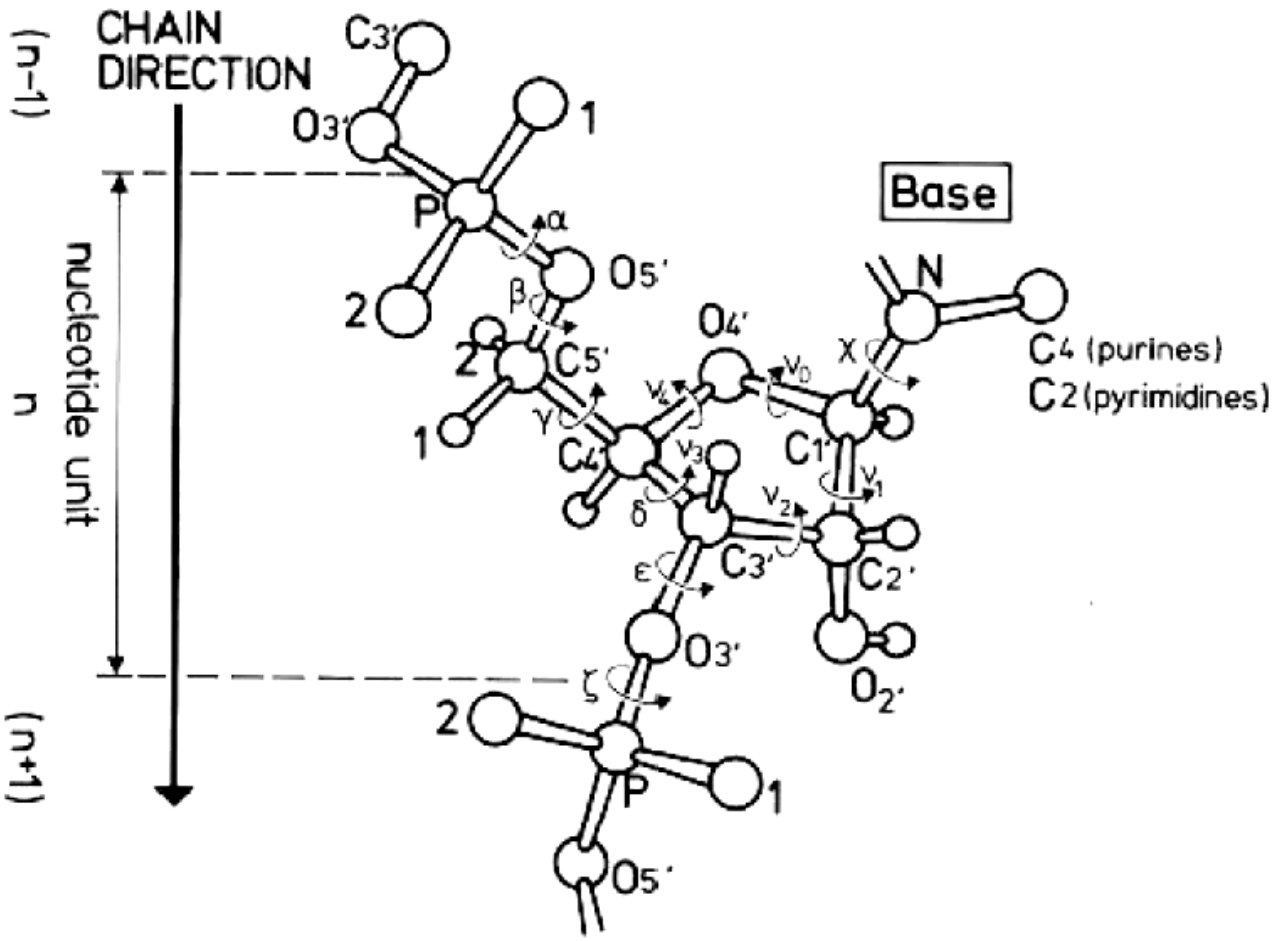

Figure 2: Torsion angles of nucleic acid backbone. [30]

2.2 Formal representation of RNA

The definitions have been adapted from [22].

To represent RNA in our model we need to capture the primary and secondary structure in a symbolic formalism.

Primary sequence
The set of all primary sequences of length n is given by

$$\mathfrak{S}_p(n) = \{A, C, G, U\}^n \tag{1}$$

The primary sequence p of an RNA of length n is a sequence of $\mathfrak{S}_p(n)$.

Secondary structure
Formally a secondary structure is a set of nucleobase pairings.
The set of possible pairings is defined as

$$\Omega = \{(A,U),(U,A),(G,C),(C,G),(G,U),(U,G)\}$$

Let $p \in \mathfrak{S}_p(n)$ be a primary sequence p of length n, and i, j, k, j are positional indices of p, then the set of all possible base pairings of p is given by

$$\mathfrak{S}_s(p) = \{(i,j) \mid 1 \leq i < j \leq |p|, (a_i, a_j) \in \Omega\} \tag{2}$$

Considering steric hindrance $\forall(i,j) : |j-i| > 3$ results in

$$\mathfrak{S}_s(p) = \{(i,j) \mid 1 \leq i < j \leq |p| \wedge |i-j| > 3 \wedge (a_i, a_j) \in \Omega\} \tag{3}$$

Notice that one base can take part in many pairings in this definition.

To obtain a valid RNA secondary structure two restrictions have to be added.
The first restriction is to allow only one pairing per base.

$$\forall(i,j),(k,l) : i = k \leftrightarrow j = l$$

The second restriction is to prohibit pseudoknots.

$$\forall(i,j),(k,l) : i < k < l < j \vee k < i < j < l$$

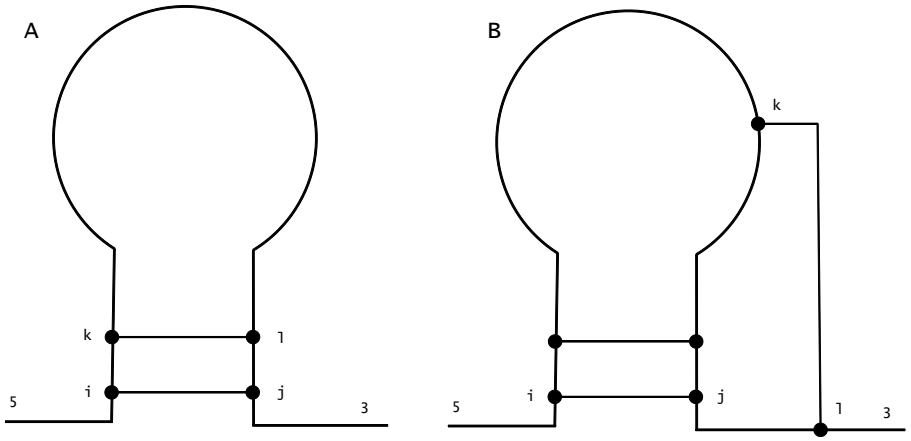

Figure 3: Loop without pseudoknot (A) and with pseudoknot (B).

A set s is a secondary structure of p if

$$s = \{(i,j) \in \mathfrak{S}_s(p) \mid \forall(i,j),(k,l) : ((i < k < l < j) \vee (k < i < j < l)) \wedge (i = k \leftrightarrow j = l)\}$$

If $s = \{\}$ then s is the open chain.

The set of all possible secondary structures compatible with a primary structure p is given by

$$\begin{aligned} \mathcal{S}(p) = \{\{(i,j)\} \in \mathcal{P}(\mathfrak{S}_s(p)) \mid \forall(i,j),(k,l) : \\ ((i < k < l < j) \vee (k < i < j < l)) \wedge \\ (i = k \leftrightarrow j = l)\} \end{aligned} \tag{4}$$

where $\mathcal{P}(\mathfrak{S}_s(p))$ is the power set of $\mathfrak{S}_s(p)$.

The set $\mathcal{S}(p)$ builds the conformation space $\mathcal{C}(p)$ of the primary sequence p.

2.3 Energetic description

The energy "representing" a molecule should not only contain its internal energy, but also kinetic and external induced potential energy. This energy is called the *free energy* of the system. Since the term "*system*" originates from thermodynamics, we should follow its nomenclature and use the term *free enthalpy* (also called *Free Gibbs Energy*) [1].
The free enthalpy includes the available energy of the molecule to perform non-mechanical (or better non expansion) work, resulting in the ability to change its conformation. Since a secondary structure is a macroscopic state, its energetic state is temperature dependent. The

[1] *IUPAC recommends to call "Free enthalpy" "Gibbs energy". [20, http://goldbook.iupac.org/G02629.html]*

term enthalpy also includes the (thermodynamic) entropy (the number of possible microstates) of the system resulting in $\Delta G = U + pV - T\Delta S$.

Conceptually, there are different components that contribute to ΔG(adapted from [32]). Hydrogen bonding, the strongest of the weak forces, is responsible for base pairing. The very weak van der Waal's interactions (mainly London dispersion force) are in sum not negligible. Actually they are responsible for the main part of stabilization via stacking of aromatic systems between neighbored nucleobases. Steric repulsion is a result of interaction of different charges. As the entropy of the system cannot decrease, the loss of spatial freedom results in increase of entropy of the environment, so entropy acts as an entropic "force". Hence the overall enthalpy of RNA can be calculated as sum of the enthalpies of its secondary structures, decomposed into "loops" L.

$$F(S) = \sum_{L \in S} F_L \tag{5}$$

The enthalpic contributions of different loops are taken from the Turner energy model [37].

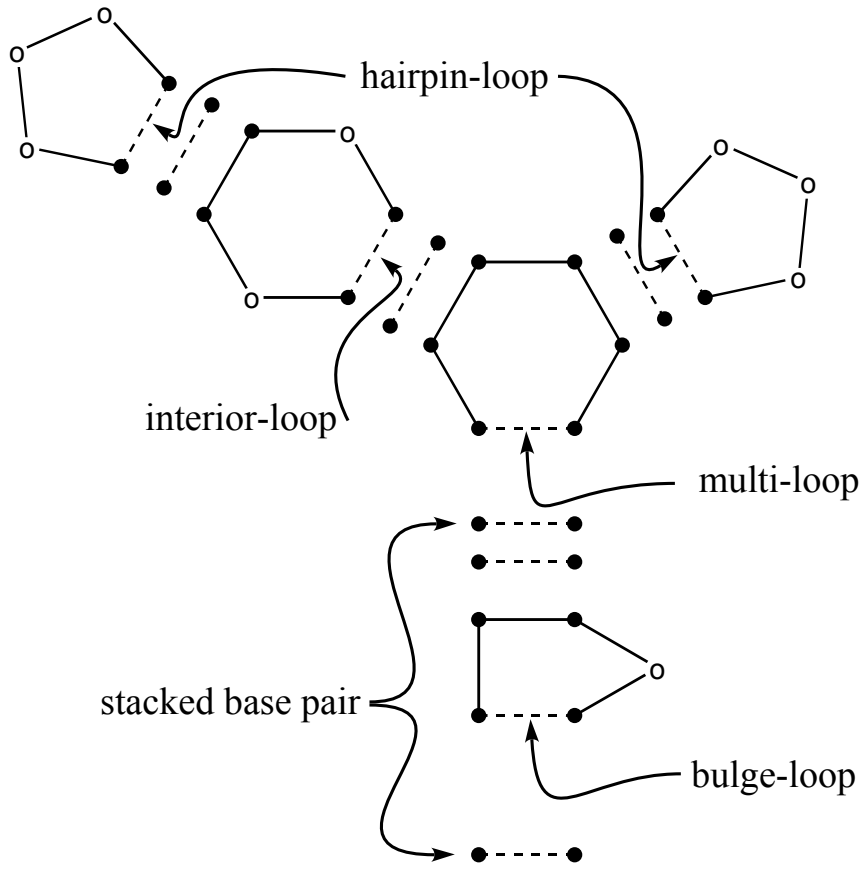

Figure 4: Typical secondary structure elements are outlined in this figure. Each structural motive can be decomposed into loops allowing calculation of their enthalpy.(Adapted from [8]).

3 RNA folding - thermodynamic versus kinetic view

In living organisms RNA is produced by RNA polymerase. The environment has great influence on the folding process, but beside intensive properties such as temperature or pressure the interaction with for example solution molecules will not be taken into account, so the folding process is considered to take place in some kind of "vacuum".
The energetic state of a molecule correlates with its stability. Generally the folding process should end in such a meta stable state. This means the molecule will only change its macroscopic conformation with the energy of external forces.
With these thermodynamic potentials the folding process can be seen as interaction of entropy *S* and enthalpy *H*. [23]
Tertiary structures will not be considered in our calculations, because it is assumed that the majority of free enthalpy of RNA is covered by the enthalpy of its secondary structures. Thus we suppose the folding of RNA to be determined mainly by secondary structure formation processes. The RNA formation process is hierarchical in the sense that intermediate secondary structures are formed as the RNA folds. The secondary structure motifs are better defined than in the protein case.
During the folding process the RNA molecule will change its conformation and hence its configuration and finally its secondary structure. As shown in Figure 2, RNA has many rotational bindings due to many only partial double bonds, causing a high degree of spatial freedom. This freedom has led to the assumption that RNA (rather than DNA) could have been the first enzymatic active bio-molecule (''RNA-World-Hypothesis"[15]).
Due to limitations of knowledge and computer power it is currently not possible to simulate the real process of RNA folding. Several approaches have been made to circumvent this problem.

- Model only properties which are thought of to be important for folding.
- Use stochastic methods to narrow the solution space.
- Calculate probabilities of possible folding pathways instead of the folding process.
- Use only small sequences as input.

The first step to simulate RNA folding is to reduce the nature of the RNA molecule to properties which are relevant for the formation process. RNA folding is a biochemical reaction based on interaction of electromagnetic fields, Van-der-Waals-forces, quantum effects and so on. Part of the success of living organisms is the evolutionary development of error tolerant systems, so nature does not rely on the necessity of exact solutions, but is content with approximative results. It is not clear whether any kind of evolution could have taken place at levels below the molecular level[12], but this possibility cannot rule out that the organic evolution on earth is driven by molecular forces.
Therefore it is sufficient to consider only those forces acting on the RNA molecule, and rule out quantum effects, what makes the problem a matter of classical kinetics.
To obtain a comprehensive picture of the (temporal) development of the folding process and enhance the prediction of stable structures it is necessary to model not only the formation of the structures itself, but also the dynamics of transitions between different structural realization possibilities. The standard model to capture all those possibilities is the energy landscape, which will be described in more detail in the following sections.

3.1 Continuous space models

Continuous space models based upon the calculation of infinitesimal small changes in the conformation of molecules caused by interaction of force fields. To make these models computable the changes are integrated over a finite time step by incorporation of Newtonian laws of motion (the usual solution in Molecular Dynamics).[34]

3.2 Discrete space models

It is not known how (and even if) nature calculates the folding process, but probably not by use of our sophisticated models (actually nature does not need a model at all). A model is always an idealization and simplification, whereby its quality depends on the chosen parameters. As it is impossible to simulate the folding process itself, we describe it as sequence of changes in free enthalpy at the level of secondary structure.
In this model a secondary structure represents an ensemble of RNA molecules, because the spatial information about unpaired regions is neglected.
The reduction to defined states of secondary structures is responsible for the discreetness of the model.

3.3 Conformation space as thermodynamic ensemble

Applying a thermodynamic potential function E over the conformation space $\mathcal{C}(p)$ results in a thermodynamic ensemble. Each formal structure has its free enthalpy, which can be calculated as shown above. The energy distribution of these tuples follows a Maxwell-Boltzmann statistic with the partition function

$$Z \equiv \sum_i e^{-\frac{E_i}{k_B T}} \tag{6}$$

where
$k_B = R/N_A\ [J/K] \ldots$ *Boltzmann constant*
$R\ [J/molK] \ldots$ *Gas constant*
$N_A[mol^{-1}] \ldots$ *Avogadro constant*

In the case of degenerated energy levels the partition function can be expressed in terms of the density of states $g(E_i)$ [5][33].

$$Z \equiv \sum_i g(E_i) \cdot e^{-\frac{E_i}{k_B T}} \tag{7}$$

The ensemble average of any thermodynamical property depending on the energy can be calculated as weighted sum of

$$\langle \mathcal{P} \rangle_{eq} = \frac{1}{Z} \cdot \sum_i \mathcal{P}(E_i) \cdot g(E_i) \cdot e^{-\frac{E_i}{k_B T}} \tag{8}$$

The partition function can be efficiently calculated by methods of *Dynamic Programming* (see 3.4.1).

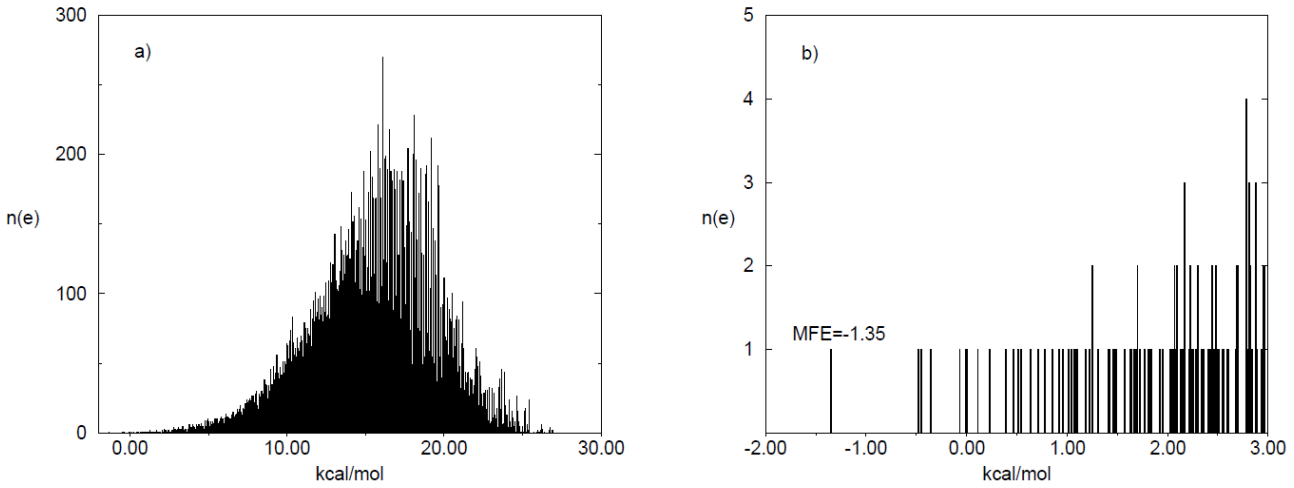

Figure 5: Density of States of the RNA sequence GUCGUAGUCGAUGCUAGCUGAUCAC. Left Whole energy range. Right Enlargement of lower energy range shows the discrete nature of the state space and also an energy gap between *MFE* and the most stable suboptimal structure. (Adapted from [5]).

The pure energetic view of the state space does not provide much information about the actual folding. To overcome this problem we need to introduce an additional reaction coordinate.

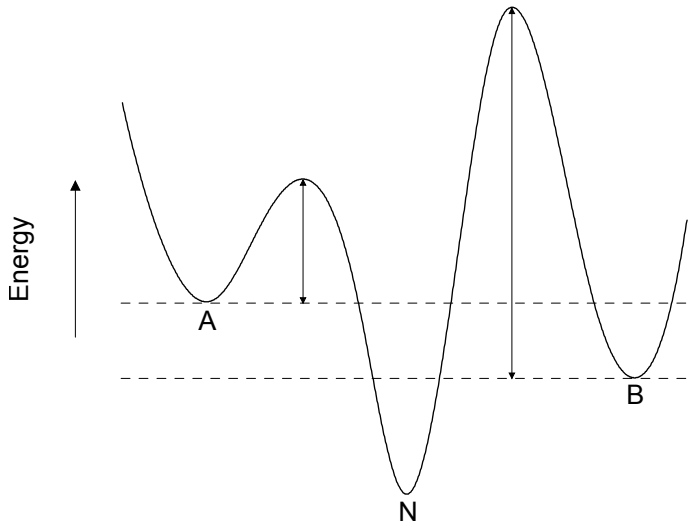

Figure 6: Thermodynamic versus kinetic reaction coordinate. State B is energetically closer to N (lower energy), but state A is kinetically closer to N (smaller barrier to cross).For didactic reasons a continuous reaction coordinate is used as abscissa. In the realm of RNA secondary structures energy and reaction coordinate are discrete. (Adapted from [8]).

3.4 Secondary structure prediction

3.4.1 Dynamic programming

Many problems in bioinformatics can be solved by dynamic programming. The term *dynamic programming* is quite confusing, since it is neither *dynamic* nor *programming* nor a *dynamic programming language* nor a *programming paradigm*. Actually it as a mathematical method, which has been used for centuries in physics to transform a (global) phase space into a (time-depended) state space, whereby the optimization is done by use of functionals. This method is in turn based more or less on the principles of scientific work as proposed by Rene Descartes [6] and goes even back to antic geometry.
To stop confusion it was Richard Bellman who formulated the *Principle of Optimality* as he worked on engineering control theory, which describes the necessary conditions to solve a

problem with dynamic programming [3]. Bellman chose the term "dynamic programming" to express the time-varying aspect of the problems (for further reasons see [4]). These conditions are *optimal substructure* and *overlapping subproblems* (if the latter condition is missed, the problem can be solved by *"divide and conquer"*[2]).
Optimal substructure means that a problem can be solved by solving its subproblems.
Overlapping subproblems requires that the subproblems can be solved by use of the same algorithm in a recursive manner, thus allowing dynamic programming to solve each subproblem only once. The following algorithms all use this optimization method.

3.4.2 Secondary structure prediction algorithms

The first effective algorithm for secondary structure prediction of RNA was the *Algorithm of Nussinov* also called the *Maximum-Matching-Algorithm*. It calculates the structure with the maximum number of nucleobases paired. Since no energetic or biological information is used, the resulting structure is not always biological relevant. [29]
The *Algorithm of Zuker* works similar to the algorithm of Nussinov, but improves the predictive result by incorporation of the free enthalpy of the structures. [40][39][38]
A somewhat different approach is used by the *Algorithm of McCaskill*. This algorithm calculates the partition function of all secondary structures and also the base pairing probability matrix in the resulting Boltzmann weighted ensemble. [27]
The *Algorithm of Wuchty* is an enhancement of the algorithm of Zuker. It calculates all structures within a given enthalpy range and finds more suboptimal structures due to better handling of multi loops. [36]

3.5 Conformation space as discrete folding landscape

The conformation space $\mathcal{C}$ is now extended to a metric conformation space by introduction of a kinetic reaction coordinate defined by a move function. A move function μ defines a mapping of one structure to another.

$$\mu : \mathcal{C} \to \mathcal{P}(\mathcal{C}), \boldsymbol{x} \mapsto \mu(\boldsymbol{x})$$

The set $\mu(\boldsymbol{x})$ is the neighborhood of *x* defined by μ.

The function performs the mapping by adding or deleting one or more base pairs from the secondary structure representation.

[2] *Correctly "Divide et impera".*

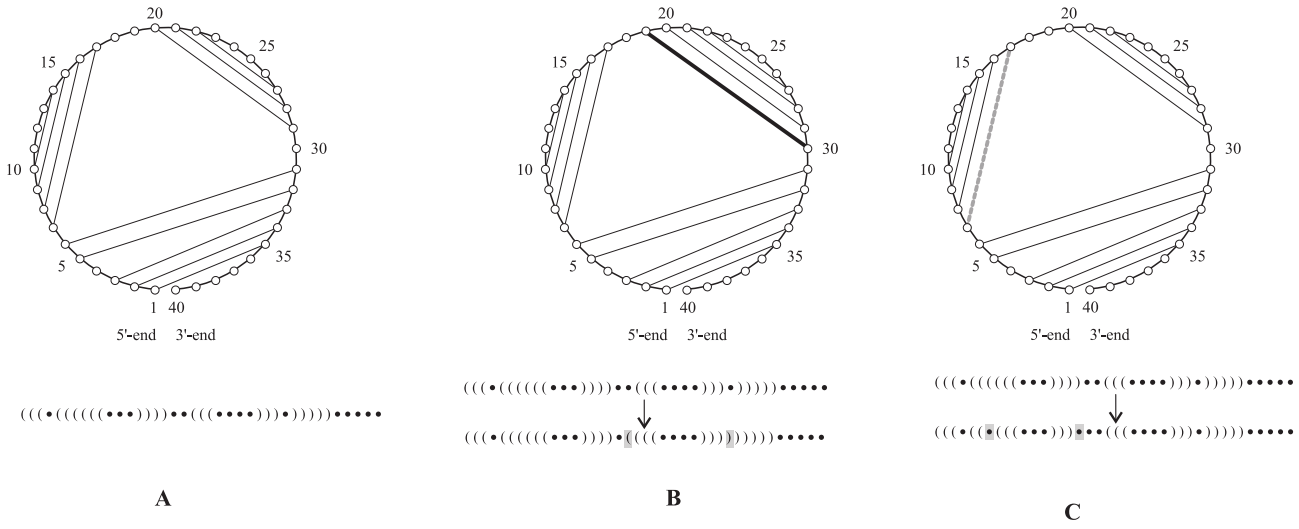

Figure 7: Elementary moves in RNA folding. Secondary structures are shown in circle and parenthesis representation. The structure A is changed by the formation B or the removal C of a base pair. The base pair after a move is shown in bold, the one being changed is shown by a gray dotted line. (Adapted from [9]).

We will restrict ourselves to insertion and deletion moves (figure 7). The set of allowed move functions defines the move set $\mathcal{M}$.

Adapted from [26] a move set of size *k* (whereby *k* is the number of move functions) is defined as follows.

$$\mathcal{M}_k : \mathcal{C} \to \mathcal{P}(\mathcal{C}), x \mapsto \bigcup_{1 \leq i \leq k} \mu_i(x)$$

with the following conditions

$$\forall x \in \mathcal{C} : x \notin \mathcal{M}(x)$$
$$\forall x, y \in \mathcal{C} : y \in \mathcal{M}(x) \leftrightarrow x \in \mathcal{M}(y) \ (\textit{symmetry})$$
$$\forall x \in \mathcal{C} : \mathcal{M}(x) \in C \ (\textit{closure})$$
$$\forall x, y \in \mathcal{C} : \exists (t_i, t_{i+1}, \ldots, t_{n-1}, t_n), t_i \in \mathcal{C} :$$
$$t_i = x \wedge t_n = y \wedge \forall 1 < i \leq n : t_i \in \mathcal{M}(t_{i-1}) \ (\textit{weak ergodicity})$$

The power function $\mathcal{P}(\mathcal{C})$ is needed, because each element is mapped to a set of elements.

The move set $\mathcal{M}$ defines a relation of topological neighborhood between the elements of the conformation space according to their structure.
Higher dimensional spaces such as the conformational space (depending on the length of the sequence) are better seen as a hyper surface, where each point represents a specific state of the system (hence a specific structure). The folding process can now be seen as movement on a hyper surface.

3.6 Discrete energy landscape

The metric conformation space can be described as a connected, undirected, simple graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$.

However, the hypersurface represented by this graph is still euclidean "flat".
Applying a thermodynamic potential function *E* on the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ results in an energy landscape $\mathfrak{E}$.

An energy landscape $\mathfrak{E}$ is a tuple $(\mathcal{G}(\mathcal{V}, \mathcal{E}), \mathcal{M}, E)$.

The model of energy landscape was first used to describe protein folding in 1975 by Frauenfelder [2][13]. In the case of the RNA model the conformational coordinates are the pairing states of the nucleobases.
With the incorporation of the potential function the folding process is much better described, because both energetic and conformational nearness are included in the model. The usual potential function is the free enthalpy $\Delta G = H - T\Delta S$, whereby the enthalpy H accounts for the non mechanical work (such as non covalent intra-molecular forces) and the (conformational) entropy S for all possible conformations and solvent configurations. As our model assumes the folding to take place in some kind of vacuum, the environment for entropy interchange should be seen as imaginary heat bath.

3.7 Topological properties of the discrete energy landscape

Several paragraphs in these section are adapted from [26], [11] and [35].

The ensembles of structures define the topology of the energy landscape, which in return influence the folding process.
As the folding is mapped on the hypersurface it is crucial to define and understand its properties.

Neighborhood
Since the move set $\mathcal{M}$ generates the graph $\mathcal{G}$ we can define neighbors as follows.
The set of adjacent vertices of $x \in \mathcal{V}$ is given by

$$\mathcal{N}(x) = \{y \in \mathcal{V} \mid \{x, y\} \in \mathcal{E}\} \tag{9}$$

In other words the neighborhood N of a structure x is the set of its neighbors reachable by the moves defined in a move set $\mathcal{M}$.

$$N(x) = \{s \in \mathcal{C} \mid s \in \mathcal{M}(x)\} \tag{10}$$

Energetic-lexicographical order
The energy landscape of real molecules is often energetically degenerate, which means that $x, y \in \mathcal{V} : E(x) = E(y) \nRightarrow x = y$. Of course $x, y \in \mathcal{V} : x = y \Rightarrow E(x) = E(y)$ holds still true. For exact definitions of properties on such landscapes we refer to [11][10][31]. Since those exact definitions make no difference on the folding process in our model, we avoid this problem by introduction of an energetic-lexicographically order as used in many articles [33][22][26].

All structures can be lexicographically ordered according to their dot-bracket-notation with the alphabet A={ "(" , ")" , "." }, whereby the order "$\prec$" is defined by the order of natural numbers encoding the characters according to the ASCII table [1].
Given two structures $x, y \in \mathcal{V}$ the order $\prec_{lex}$ is defined as follows.

$$x \prec_{lex} y \leftrightarrow x_1 \prec y_1 \vee (\exists(x_i, x_{i+1}, \ldots, x_{n-1}, x_n) : 1 \leq i \leq n < |x| : x_i = y_i \wedge x_{n+1} \prec y_{n+1}) \tag{11}$$

Figure 8: Explanation of lexicographic order. According to the ASCII table the order of alphabet A is $''leftbracket'' \prec ''rightbracket'' \prec ''dot''$. On the left the first character differs ($x_1 \prec y_1$). On the right exists a sequence $s = (x_1, x_2)$, where $\forall 1 \leq i \leq 2 : x_i = y_i$ and on the third position $x_3 \prec y_3$.
In both cases $x \prec_{lex} y$ holds true.

The lexicographical order takes effect if the energy of two structures is identical, resulting in the energetic-lexicographical order relation denoted by the symbol $\ll$.

$$x \ll y \leftrightarrow E(x) < E(y) \vee (E(x) = E(y) \wedge x \prec_{lex} y) \tag{12}$$

Local minimum[3]
A vertex $x \in \mathcal{V}$ is a local minimum if it is energetic-lexicographically lower than all its neighbors. We can therefore define the set of local minima as

$$\mathbb{M} = \{x \in \mathcal{V} \mid \forall y \in N(x) : x \ll y\} \tag{13}$$

Global minimum / Minimum free energy (*MFE*)[3]
A vertex $x \in \mathcal{V}$ is a global minimum if it is energetic-lexicographically lower than all other vertices. We can therefore define the global minimum as

$$MFE \Leftrightarrow x \in \mathcal{V} \mid \forall y \in \mathcal{V} \setminus \{x\} : x \ll y \tag{14}$$

Walk
A walk w is a sequence of vertices. The set of all walks is defined as

$$\mathbb{W} = \{(w_i, w_{i+1}, \ldots, w_{n-1}, w_n) | \ \forall\ 1 < i \leq n = |w| : w_i \in N(w_{i-1})\} \tag{15}$$

A structure can occur more than once in a walk.

Adaptive walk
An adaptive walk $\mathbb{A} \in \mathbb{W}$ is a walk in which each vertex is followed by an energetic-lexicographical lower neighbor.

$$\mathbb{A} = \{w \in \mathbb{W} \mid \forall 1 \leq i < |w| : w_{i+1} \in N(w_i) \wedge w_{i+1} \ll w_i\}$$

Gradient walk
A gradient walk $g \in \mathbb{W}$ is a walk in which each vertex is followed by its energetic-lexicographically lowest neighbor. The set of gradient walks is defined as

$$\mathbb{G} = \{w \in \mathbb{W} \mid \forall 1 \leq i < |w| : \nexists x \in N(w_i) : x \ll w_{i+1})\} \tag{16}$$

Gradient basin
The gradient basin $\mathcal{B}$ of a local minimum $m \in \mathbb{M}$ is the set of all vertices which are element of a gradient walk ending in m.

$$\mathcal{B}(m) = \{x \in \mathcal{V} \mid \exists\ g \in \mathbb{G} : g_1 = x \wedge g_{|g|} = m \wedge m \in \mathbb{M}\} \tag{17}$$

Each structure is uniquely assigned to a gradient basin (and its local minimum).[3]

Accessibility
Let $\mathbb{P}_{xy}$ be the set of all walks from x to y. We say that x and y are mutually accessible at level η, in symbols $x \underset{\rho}{\overset{\eta}{\leftrightarrow}} y$, if there is a walk $\rho \in \mathbb{P}_{xy}$ such that $E(z) \leq \eta$ *for all* $z \in \rho$, respectively.

The relation $\underset{\rho}{\overset{\eta}{\leftrightarrow}}$ is obviously symmetric ($x \underset{\rho}{\overset{\eta}{\leftrightarrow}} y$ implies $y \underset{\rho}{\overset{\eta}{\leftrightarrow}} x$) and transitive ($x \underset{\rho}{\overset{\eta}{\leftrightarrow}} y$ and $y \underset{\rho}{\overset{\eta}{\leftrightarrow}} z$ implies $x \underset{\rho}{\overset{\eta}{\leftrightarrow}} z$). It is reflexive for all $\eta \geq E(x)$.

[3] *It should be noticed, that this definition holds only true if there are not too many degenerate structures around the minimum (which is almost never the case when modeling an RNA energy landscape), because the definition of equation 12 could result in artificial local minima (and saddle points), where all structures have the same energy, but one or more are only lexicographically lower than all their neighbors (and by definition local minima).*

Saddle height
The saddle height $\hat{f}(x, y)$ between two vertices $x, y \in \mathcal{V}$ is the minimum height at which they are accessible from each other, i.e.,

$$\hat{f}(x, y) = \min_{\rho \in \mathbb{P}_{xy}} \max_{z \in \rho} f(z) = min\{\eta | x \underleftarrow{\rho\ \eta\ \rho} y\} \tag{18}$$

In particular, we have $\hat{f}(x, x) = f(x)$. The saddle heights $\hat{f}(x, y)$ forms an ultrametric distance measure between distinct local minima.
For completeness we note that the barrier of a local minimum $m \in \mathbb{M}$ is
$\beta(m) = \min\{+\infty; \hat{f}(m, n) - f(m) \mid n \in \mathbb{M} \wedge f(n) < f(m)\}$.

Lowest saddle point
Given two vertices $m, n \in \mathbb{M}$ the set of lowest saddle points $\bar{S}$ connecting these local minima is

$$\bar{S}(m, n) = \{x \in \{\rho \in \mathbb{P}_{mn} \mid m, n \in \mathbb{M} \wedge m \neq n\} \mid E(x) = \hat{f}(m, n)\} \tag{19}$$

For the sake of simplicity a saddle point $\bar{s}$ can be seen as the lowest vertex which has neighbors in different (gradient) basins.

Valley
The set of vertices x reachable from a lowest saddle point $\bar{s} \in \bar{S}$ at the level $\hat{f}(\bar{s})$ is the valley $\mathbb{V}(\bar{s})$.

$$\mathbb{V}(\bar{s}) = \{x \in \mathcal{V} \mid \bar{s} \underleftarrow{\rho\ \hat{f}(\bar{s})\ \rho} x\} \tag{20}$$

Suppose two saddle points $\bar{s}_1, \bar{s}_2$ such that $E(\bar{s}_1) < E(\bar{s}_2)$. If $\bar{s}_1 \in \mathbb{V}(\bar{s}_2)$ then $\mathbb{V}(\bar{s}_1) \subseteq \mathbb{V}(\bar{s}_2)$, i.e. the valley of $\bar{s}_1$ is a "sub-valley" of $\mathbb{V}(\bar{s}_2)$. If $\bar{s}_1 \notin \mathbb{V}(\bar{s}_2)$ in which case $\mathbb{V}(\bar{s}_1) \cap \mathbb{V}(\bar{s}_2) = \emptyset$, then the valleys are disjoint.
Figure 9 depicts such a situation. Each saddle point $\bar{s}_1$ *until* $\bar{s}_6$ defines the valley $\mathbb{V}(\bar{s}_1)$, because all of them are part of a walk ending in any of the four basins k, m, n, j, going thereby not higher than $\hat{f}(m, n) = \hat{f}(k, n)$. Since $\bar{s}_7 \in \mathbb{V}(\bar{s}_1)$ the valley defined by $\mathbb{V}(\bar{s}_7)$ is a sub valley of $\mathbb{V}(\bar{s}_1)$. Note that this figure is the projection of an one dimensional cross section and there might be more basin (but not valleys) accessible from the saddle points.

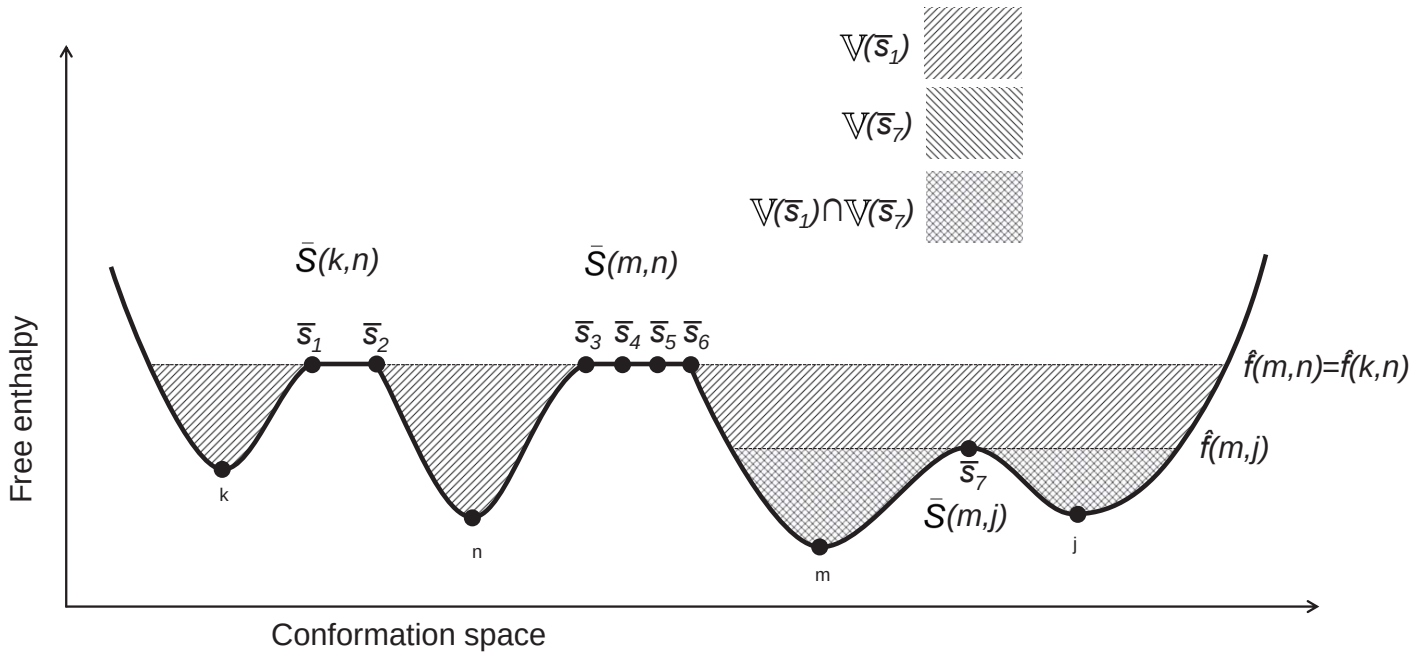

Figure 9: Cross section through a energy landscape with degenerate saddle points.
Saddle points $\bar{s}_1, \bar{s}_2 \in \bar{S}(k, n)$, $\bar{s}_3$ *until* $\bar{s}_6 \in \bar{S}(m, n)$ and $\bar{s}_7 \in \bar{S}(m, j)$.

Basin
Using the definition of the gradient basin, a basin B of a local minimum m can be defined as follows.

$$B(m) = \{x \in \mathcal{B}(m) \mid E(x) \leq \min\{\hat{f}(m,n) \mid m,n \in \mathbb{M} \wedge m \neq n\}\} \tag{21}$$

These are all vertices of the gradient basin which are not higher than the lowest saddle height of all walks connecting the local minimum m with any other local minimum n.
Note that the lowest saddle point connecting two local minima need not be a vertex of the corresponding gradient basins, because its gradient walk could end in another local minimum. Since we cannot go higher than $\hat{f}(m,n)$ the energy of that lowest saddle point might be lower than the energy of the lowest saddle point of the basins.

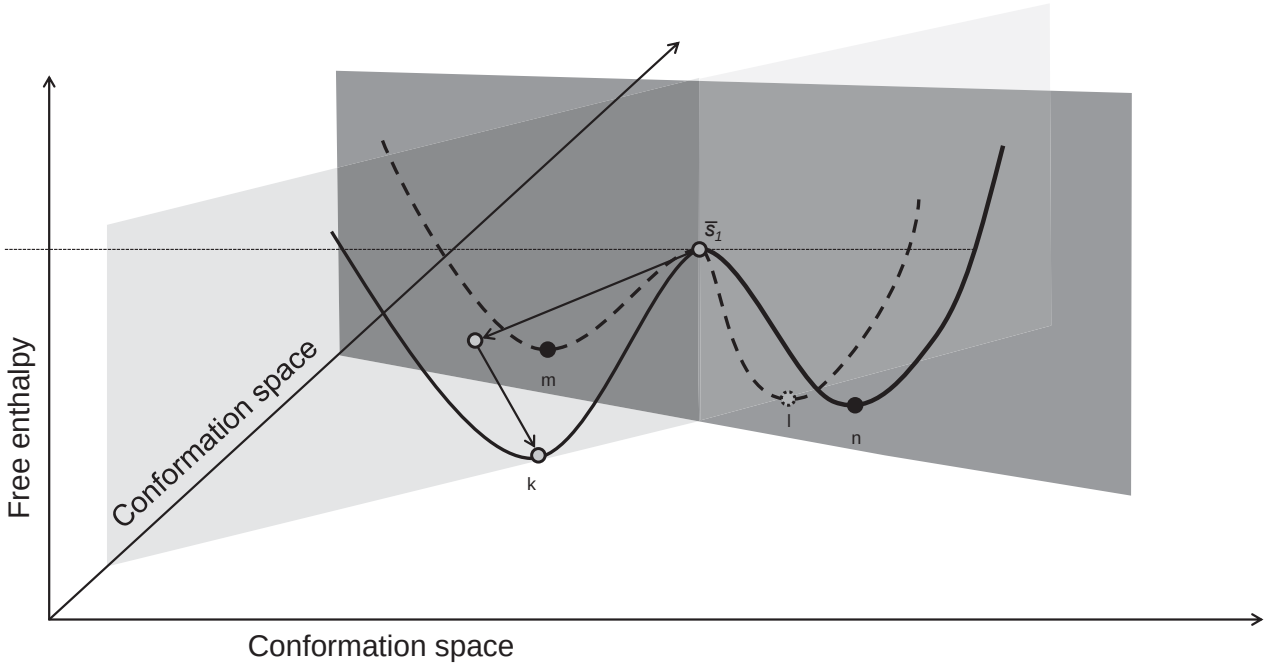

Figure 10: Energy landscape projection in two conformational coordinates. The saddle point $\bar{s}_1$ defines the basin height of local minima m and n, although it is not part of either, because its gradient walk ends in k.

3.8 Partitioning of the energy landscape into macro states

The property of valleys and sub valleys arranges the local minima and the saddle points in a unique hierarchical structure which is conveniently represented as a tree, termed *barrier tree*(first used in [18]). As a result we are able to describe the energy landscape on a more macroscopic level consisting of (gradient) basins around local minima separated by lowest saddle points. Such a graph can be obtained by different algorithms like `barriers`. Although the barrier tree is a smoothing approximation of the actually rough landscape, it has been very useful for simplifying the folding process and we will continue to build our considerations upon this model.

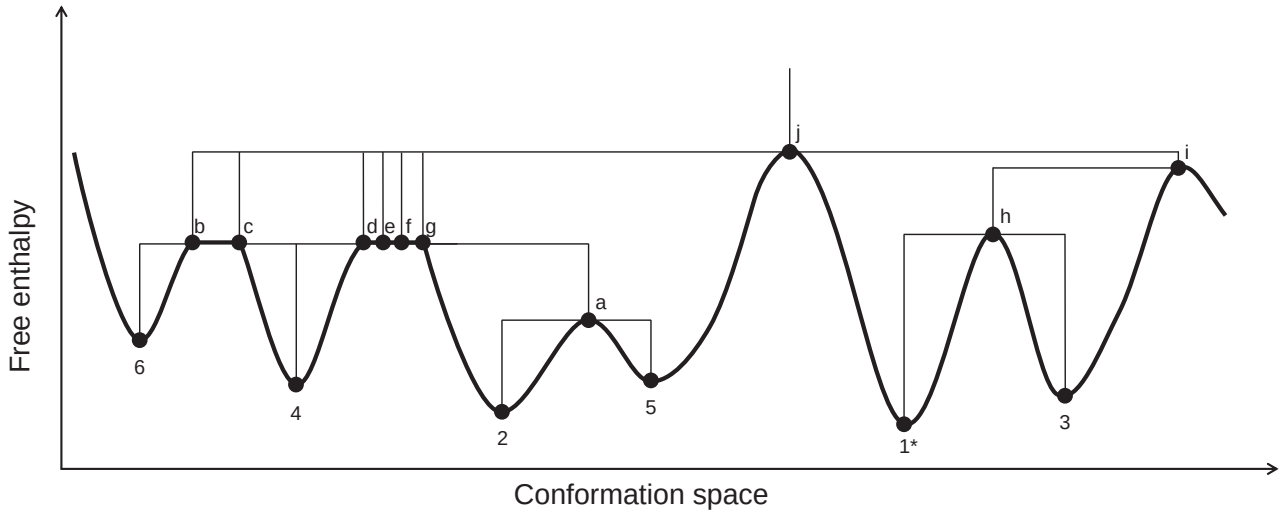

Figure 11: Barrier tree of an energy landscape. The global minimum is labeled with an asterisk. Local minima are labeled with numbers, saddle points with letters. The degenerate saddles $b - g$ are merging all together with each other and with saddle point *a*. Also the basins represented by local minima 6 and 4 are merged to the same set of saddle points $b - g$, because all for local minima *6*, *4*, *2*, *5* are in the same valley defined by them. Please note that the barrier tree shown here is also only a projection, although not so much simplifying the real situation as the depiction of the energy landscape itself.

4 Modeling the folding process of RNA

The energy landscape provides a top approximative for modeling of the folding of an RNA molecule with respect to energy and conformational changes, but we also need an instrument to characterize its time dependent behavior. If the temporal development of a large number of objects captured by the methods of statistical mechanics needs to be calculated, the method of choice is often a stochastic process.

A stochastic process $\{X_t \mid t \in T\}$ is a family of stochastic variables X_t over the same probability space and taking values of the measure space S, whereby the totally ordered set T is usually interpreted as time. We define $T \in \mathbb{R}$, resulting in a continuous time stochastic process. The process has to fulfill the Markov property (the behavior of each state only depends on its predecessor), resulting in a continuous time Markov process.
The process has to be time homogeneous (the states are not time dependent), resulting in a time homogeneous continuous time Markov process.
As the conformation space is discrete the measure space of our stochastic process is also discrete, resulting in a time homogeneous continuous time Markov process on discrete measure space (sometimes called time homogeneous continuous time Markov chain).
All our considerations are based upon time homogeneous continuous time Markov chains.

The probability to reach state j from state i within Δt is given by

$$Prob\{X_{\Delta t} = j \mid X_0 = i\} = p_{ji} \tag{22}$$

Although this transition probability is time independent, it scales with the length of the interval Δt.
We write for small Δt

$$Prob\{X_{\Delta t} = j | X_0 = i\} = k\Delta t + o(\Delta t) \tag{23}$$

whereby $o(\Delta t)$ is the sum over the probabilities to pass through intermediate states between i and j, and the factor k is a transition rate between i and j.
In contrast to transition probabilities p_{ij} the transition rates k_{ij} do not depend on the length of time intervals. This means that the probabilistic behavior of a continuous time Markov chain is completely described by the initial state (or distribution) and the transition rates between distinct states.

4.1 The master equation

Section adapted from [14][16][35].

The probability distribution over a discrete state space can be expressed by the differential form of the Chapman-Kolmogorov equation (which is called the master equation):

$$\frac{dP_i(t)}{dt} = \sum_{j \neq i} [P_j(t)k_{ij} - P_i(t)k_{ji}] \tag{24}$$

$$\text{where } p_{j \to i} = P_j(t)k_{ij} \text{ and}$$
$$p_{i \to j} = P_i(t)k_{ji}$$

according to laws of conditional probability.

In this form the master equation can be seen as a gain-loss equation for the probability of each state *n*, whereby the first term stands for all gains from other states *j* and the second term for all losses to other states *j*.
The master equation describes the relation between transition probabilities and transition rates (see equation 23).

$$\frac{dP_i(t)}{dt} = \sum_{j \neq i} [P_j(t)k_{ij} - P_i(t)k_{ji}] = \sum_j P_j(t)k_{ij} \tag{25}$$

Rewriting the master equation in matricial form

$$\frac{d}{dt}P(t) = \mathbf{U}P(t) \tag{26}$$

whereby $\mathbf{U} = (u_{ij})_{i \times j}$ is a square intensity matrix (transition matrix), which contains the transition rates between different states of the system derived from the transition probabilities *k*.

$$u_{ij} = \begin{cases} k_{ij} & \text{if } i \neq j, \\ -\sum_{l \neq i} k_{il} & \text{if } i = j \end{cases} \tag{27}$$

We are interested in calculating the temporal distribution vector $P(t)$, which can be calculated from the explicit solution of

$$P(t) = e^{t\mathbf{U}}P(0) \tag{28}$$

where $P(0)$ is the initial distribution vector.

It has been shown in [33] how $P(t)$ can be calculated in the eigenspace of the system, because it is quite difficult to calculate the exponentiation of a matrix numerical.

4.2 Markov chain Monte Carlo for RNA kinetics

Section adapted from [33].
Due to the high dimensionality of the conformation space, it is computational not feasible to calculate all transition rates between micro states with a direct approximation method. To overcome this problem one can use a Monte Carlo method to simulate the stochastic behavior of the probability distribution[21]. Since our model is already based upon Markov chains, we use a Markov chain Monte Carlo method for sampling of random values, whereby the underlying Boltzmann distribution is correlated to the Metropolis rule[28].

$$k_{ij} = \begin{cases} \begin{cases} e^{-\frac{\Delta G}{k_B T}} & \text{if } G_i > G_j, \\ 1 & \text{if } G_i \leq G_j \end{cases} & \text{if } i \in N(j), \\ 0 & \text{else} \end{cases} \tag{29}$$

$$\text{with } \Delta G = G_j - G_i.$$

The values generated by the Metropolis Rule are used in the transition matrix.
Written in the form $min\{e^{-\frac{\Delta G}{k_B T}}, 1\}$ it is more clearly, that downwards transitions are more probable than upwards ones, whereas all downwards steps have the same probability 1.

4.3 Micro state kinetics

The program `kinfold`[9] simulates the stochastic folding kinetics using a rejection-less Monte Carlo method to solve the master equation and calculate the trajectory of a single RNA secondary structure.
Let $\mathcal{S}(\mathcal{I})$ be the set of energetically ordered suboptimal secondary structures of the primary sequence $p \in \mathfrak{S}(n)$. A trajectory $\mathcal{T}(\mathcal{I})$ (as computed by `kinfold`) is a time-ordered series of secondary structures in $\mathcal{S}(\mathcal{I})$. Because the conformation space of secondary structures is always finite, every trajectory will reach S_0 after sufficiently long time. The *folding time* τ (associated with a trajectory) is defined as the first passage time, that is, the time elapsed until S_0 is encountered first.
Like almost all Monte Carlo methods the algorithm has to deal with high autocorrelation, which means that the system could stay for a long time in a meta stable state, before reaching the (thermodynamic) equilibrium, resulting in a quite long first passage time. Additionally we obtain only the trajectory from the defined start structure to the defined end structure. One solution is to reduce the number of folding paths in an appropriate way.

4.4 Reduced kinetics

To describe the folding dynamics not only of very small RNA molecules, we need to coarse-grain the representation of the energy landscape.

4.4.1 Arrhenius Kinetics

This model uses only the transition rates between the local minima and saddle points as obtained by the barrier tree approximation of `barriers`. It calculates the transitions from all to to all local minima resulting in a dense transition matrix. Given the macro states of the local minima $m, n \in \mathbb{M}$ the transition rate is given by

$$r_{nm} = r_0 \, e^{-\frac{E(n)-E(m)}{k_B T}} \tag{30}$$

Very rare transitions to further "away" local minima are almost zero, but in summation remarkable, leading to an overestimation of not really occurring folding processes and an acceleration of the whole process.
Furthermore entropic terms are completely neglected, because there are many possible paths connecting two local minima.

4.4.2 Barrier tree kinetics

Section adapted from [35].
A much better approximation can be derived from the microscopic dynamics. Let $\mathbf{\Pi} = \{\alpha, \beta, \dots\}$ be a partition of the phase space $\mathcal{C}$. The classes of such a partition are *macro states*, for example the gradient basins $\mathbb{B} = \{b \subseteq \mathcal{C} \mid b = \mathcal{B}(m) \wedge \in \mathbb{M}\}$ of the local energy minima. To each macro state α we can assign the partition function

$$Z_{\alpha \in \Pi} = \sum_{x \in \alpha} e^{-\frac{E(x)}{k_B T}} \tag{31}$$

and the corresponding free enthalpy

$$G(\alpha) = -k_B T \ln Z_\alpha. \tag{32}$$

Knowing the transition rate r_{yx} from x to y given by $\sum_y r_{yx} P_x(t)$,
the transition rate from macro state α to macro state β can be written as

$$r_{\beta\alpha} = \sum_{y\in\beta} \sum_{x\in\alpha} r_{yx}\, Prob(x|\alpha) \quad for\ \alpha \neq \beta \tag{33}$$

where $Prob(x|\alpha)$ is the probability of occupying state x given the process is in macro state α.
Describing the process in terms of its macro states using the Master equation

$$\frac{dP_\alpha(t)}{dt} = \sum_{\beta\in\Pi} r_{\alpha\beta} P_\beta(t) \tag{34}$$

where $P_\alpha(t) = \sum_{x\in\alpha} P_x(t)$.

Assuming (local) equilibrium the probability to find the process in macro state α and occupying micro state $x \in \alpha$ is

$$Prob(x|\alpha) = \frac{e^{-\frac{E(x)}{k_B T}}}{Z_\alpha} \tag{35}$$

Transition rate from macro state α to macro state β can be rewritten as

$$r_{\beta\alpha} = \frac{1}{Z_\alpha} \sum_{y\in\beta} \sum_{x\in\alpha} r_{yx}\, e^{-\frac{E(x)}{k_B T}} \tag{36}$$

Using the transition state model for micro states x and y

$$r_{yx} = r_0\, e^{-\frac{E_{yx}-E(x)}{k_B T}} \tag{37}$$

whereby E_{yx} is the energy of a saddle point

the transition rate between two macro states α and β is given by

$$r_{\beta\alpha} = r_0\, e^{-\frac{G_{\beta\alpha}-G(\alpha)}{k_B T}} \tag{38}$$

And the free enthalpy of the transition state (which can be seen as macro state of saddle points) can be calculated as follows.

$$G_{\beta\alpha} = -k_B T \ln \sum_{y\in\beta} \sum_{x\in\alpha} e^{-\frac{E_{yx}}{k_B T}} \tag{39}$$

This allows us to redraw the barrier tree (which was given in terms of the energies of metastable states and their connecting saddle points) in terms of free energies of the corresponding macro states and their transition states.
It should be mentioned, that the conformation space actually consists not of macro states, but of micro states (including their entropic freedom), which are of course all anticipating in the folding process.

5 Exploration of the energy landscape

As the probability distribution of the conformational ensembles moves on the energy landscape, some structures or set of structures act as (attractive or repulsive) attractors of population probabilities. These structures are the local minima and their (gradient) basins, or very unstable formations such the open chain. However, not all local minima are highly populated at equilibrium distribution. What properties of the energy landscape are actually responsible for the final state of the system?

First we will define some general properties of vertices, which will be used to evaluate specific properties of macro states.

To measure the shortest walk to reach structure y starting at structure x, we define a distance function d.

$$d : \mathcal{C} \times \mathcal{C} \to \mathbb{N}, d(x, y) \mapsto \arg\min_{l \geq 0} (\exists w \in \mathbb{W} : w_1 = x \wedge w_l = y) \tag{40}$$

Let F be the set of vertices fulfilling a particular feature.
The minimum radial size of a macro state with respect to a particular feature F can be obtained by checking for the maximal distance where all vertices fulfill this feature. As such the *minimum radius* r_{min} is given by

$$r_{min}(m) = \arg\max_{i \geq 0} (\{x |\ d(m, x) = i\} \subseteq F(m)) \tag{41}$$

The maximum radial size of a macro state with respect to a particular feature can be obtained by checking for the maximal distance where any vertex fulfills this feature. As such the *maximum radius* r_{max} is given by

$$r_{max}(m) = \arg\max_{i \geq 0} (\exists x \mid d(m, x) = i \wedge x \in F(m)) \tag{42}$$

The number of vertices of a macro state of a local minimum m up to $r_{min}(m)$ is given by

$$N_{min}(F(m)) = \left|\{x \in F(m) \mid d(m, x) \leq r_{min}(m)\}\right| \tag{43}$$

The number of vertices of a macro state of a local minimum m up to $r_{max}(m)$ fulfilling a particular feature F is given by

$$N_{max}(F(m)) = \left|\{x \in F(m) \mid d(m, x) \leq r_{max}(m)\}\right| \tag{44}$$

The number of vertices of a macro state of a local minimum m up to $r_{max}(m)$ not fulfilling a particular feature F is given by

$$N_{max}(\neg F(m)) = \left|\{x \in \mathcal{C} \mid d(m, x) \leq r_{max}(m) \wedge x \notin F(m)\}\right| \tag{45}$$

For didactic reasons we define the radii set of local minimum m with radius r as

$$\mathfrak{R}_r(m) = \{x \in \mathcal{C} \mid d(m, x) \leq r\} \tag{46}$$

The radii set contains all vertices with distance less or equal to r starting in m.

As the conformation space $\mathcal{C}$ fulfills the weak ergodicity, the radii set will contain all vertices of $\mathcal{C}$, if we set $r = +\infty$.

5.1 Definition gradient walk core element

Whether a macro state is more or less populated depends on the transition rates from (to) other macro states. As shown above these transition rates can be calculated by summation of the transition rates of the corresponding micro states. Highly populated macro states such as those representing the neighborhood of the open chain or the minimum free energy should consist of many micro-states which tend to stay in their macro-state. But besides the fact that we compare macro states, no other topological feature has been considered. We will now look at the "internals" of the macro states.

Each vertex of a macro state has neighbors in the macro state. The macro state can only be left, if the vertex has also neighbors in other macro states (which means it is a saddle point). That is why some vertices may lie deep in the macro state, other near the "border". Furthermore different vertices might have a different number of neighbors below and(or) above them. All this means, that it is more or less difficult for a structure represented by a vertex to fold along a walk leading outside its macro state, and the number of structures having different difficulties to leave their macro state will be individual for each macro state and might not correlate with the depth or entropic size of the macro state. So we will define a property which reflects the difficulty of a structure to leave its macro state.

The simplest assumption would be to count the minimum number of steps needed to reach a saddle point. But as we do not want to completely neglect the energetic component, we have to consider the transition rates of the micro states (the vertices). Since a macro state consists of all the connected vertices below the energy of the lowest saddle point (in the case of a normal basin), a walk to leave the macro state would be a not very probably only upwards walk without any entropic information considered (the same holds true for gradient basins, because they are super sets of their normal basins). To incorporate all possible only upwards walks one would need to check the combinations of all such walks for all vertices, which is not feasible. Instead of looking for all walks going only upwards, we will only look for the walks which go only downwards for each vertex. A special type of such only downwards walks is of course the gradient walk as defined in equation 16. The more vertices have their gradient walk ending in the local minimum of their basin, the more difficult it might be for any structure of the macro state to leave. We call these vertices the *gradient walk core elements* of the macro state. These vertices just build the set of vertices in the gradient basin. This first definition of core elements might look unnecessary, but it illustrates that established concepts are included in our view of stability of macro states.

The set of gradient walk core elements of a gradient basin $\mathcal{B}$ with local minimum m is given by

$$C_{gw}(m) = \mathcal{B}(m)$$

To obtain a value of size of a macro state, we need to use the measurement definitions provided above.

$r_{min}(m)$ yields the greatest distance where all vertices are still part of the gradient basin. This is value describes best the entropic "size" of the macro state.

$N_{min}(C_{gw}(m))$ yields the number of gradient walk core elements up to $r_{min}(m)$.
This set can also be greater than the basin of m, if the lowest saddle point is below the highest elements of this set. We call it the "*(gradient walk) core*" of the macro state. This value reflects more the enthalpic density of the macro state.

$r_{max}(m)$ yields the greatest distance where at least one vertex is still part of the gradient basin. Since gradient basins have no upper limit, this value reflects the maximum elongation of the macro state.

$N_{max}(C_{gw}(m))$ yields the number of vertices which are part of the gradient basin up to $r_{max}(m)$. This value is equal to the number of vertices in the gradient basin.

$N_{max}(\neg C_{gw}(m))$ yields the number of vertices which are not part of the gradient basin up to $r_{max}(m)$. This value reflects how much of the conformation space, which is in principle reachable by macro state, is actually occupied by it.

5.2 Definition adaptive walk core element

The set of gradient walk core elements provides an estimation of size of a macro state (additional to its energetic depth). But as we use the Metropolis rule for the calculation of the micro state transitions, all downwards steps have the same rate of 1. This means that the values obtained by gradient walk core complexes are overestimating the stability of a macro state, since it seems to be much more difficult to leave (or depopulate) the (gradient) basin.

We need a more stringent definition for core elements, which are not so dependent on the transition rates, but more on the connectivity of the vertices. A single vertex has usually many neighbors below and above it. The more neighbors above or below, the higher the probability to go up or down on the energy scale. The upwards steps are quite well approximated by the micro states transition rates, but the downwards steps are all the same. To overcome this problem we could check how many neighbors are there below and additionally what will happen if we occupy one of these lower neighbors and so on. Again it is better not to check how difficult it is to leave, but how easy it is to stay in the macro state, since it is not feasible to follow all combinations of walks. We start at the local minimum and check in ascending distance if there are vertices which could follow downwards walks leaving the macro state. The more lower neighbors, which have only downwards walks ending in the local minimum, a vertex has, the more difficult it is to leave the macro state (actually an upward step is required). And if more vertices have this feature, the whole macro state becomes more stable, what means that the macro state should be a great attractor on the energy landscape.

The set of adaptive walk core elements of a basin B with local minimum m is given by

$$C_{aw}(m) = \{x \in \mathcal{C} \mid \nexists w \in \mathbb{A} : w_{|w|} \neq m \wedge x \in w\} \tag{47}$$

$|w|$ is the cardinality of the set w, so $w_{|w|}$ means the last index of walk w.

Using the measurement definitions provided above results in the following definitions.

$r_{min}(m_1)$ yields the greatest distance where all vertices have only adaptive walks ending in the local minimum.

$N_{min}(C_{aw}(m))$ yields the number of adaptive walk core elements up to $r_{min}(m)$. This set can also be greater than the basin of m. We call it the *"(adaptive walk) core"* of the macro state. This value reflects the enthalpic density of the really stable part of the macro state.

$r_{max}(m)$ yields the greatest distance where at least one vertex has only adaptive walks ending in the local minimum.

$N_{max}(C_{aw}(m))$ yields the number of adaptive walk core elements up to $r_{max}(m)$. This set cannot be greater than the gradient basin of *m*, but much greater than its basin. This maximum elongation might be important to reflect the strength of the attractor of the macro state. We call this set of vertices the *(adaptive walk) core complex*.

$N_{max}(\neg C_{aw}m))$ yields the number of vertices which are not part of the Core-Complex up to $r_{max}(m)$. This value reflects how fast the density of the Core-Complex decreases (in relation to $N_{max}(C_{aw}(m))$).

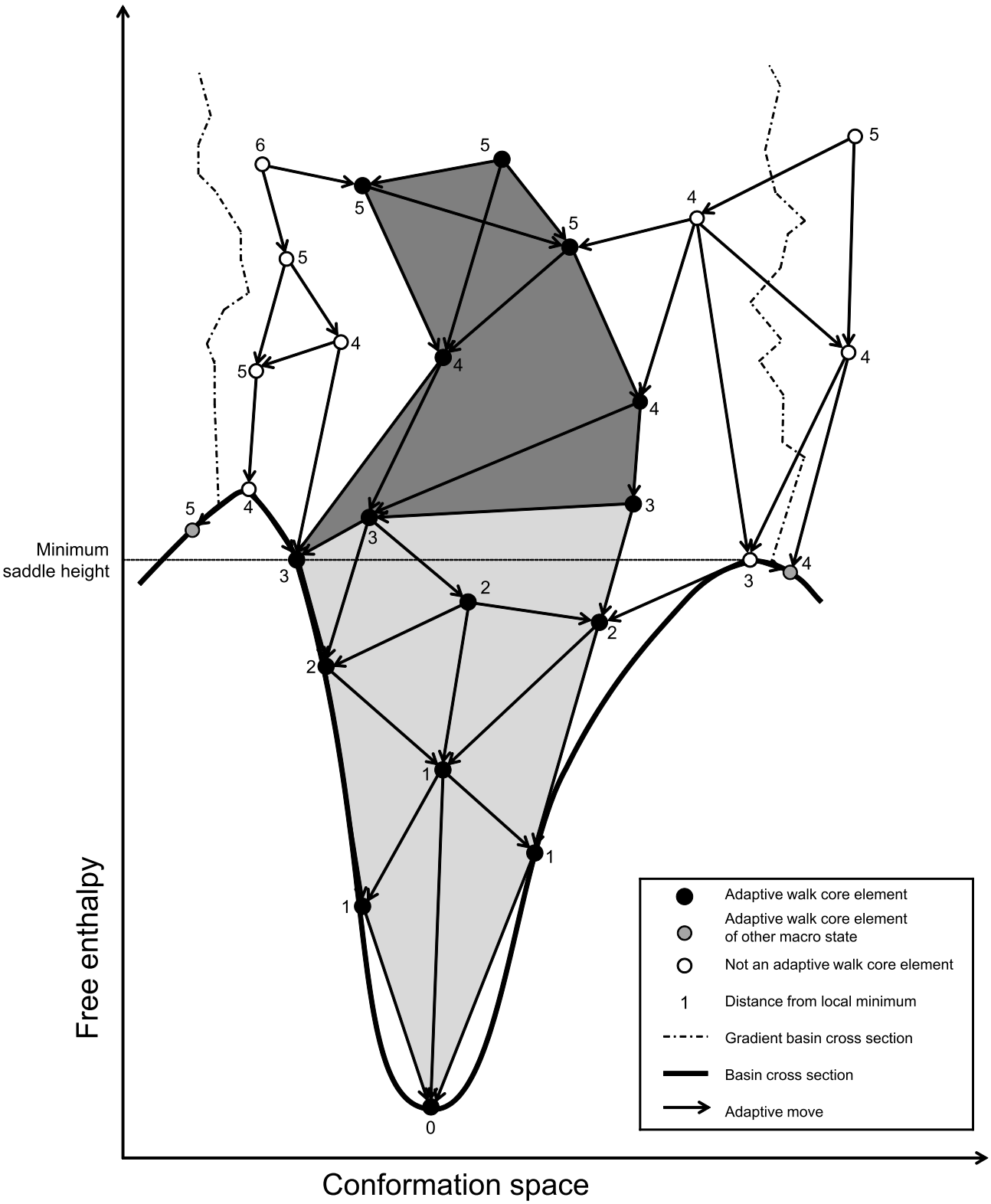

Figure 12: Cross section of energy landscape along the conformational coordinates of a macro state with adaptive walk core elements. Filled dots are adaptive walk core elements, gray dots are adaptive walk core elements of neighboring macro states, white dots are not part of the core complex. Each micro state is labeled with its distance from the local minimum. The light gray part shows the elements within the minimum core radius. The light and dark gray part together cover the elements within the maximum core radius. A little part of the minimum core lies over the minimum saddle height, protruding out of the basin. The chain dotted line marks the border of the gradient basin.

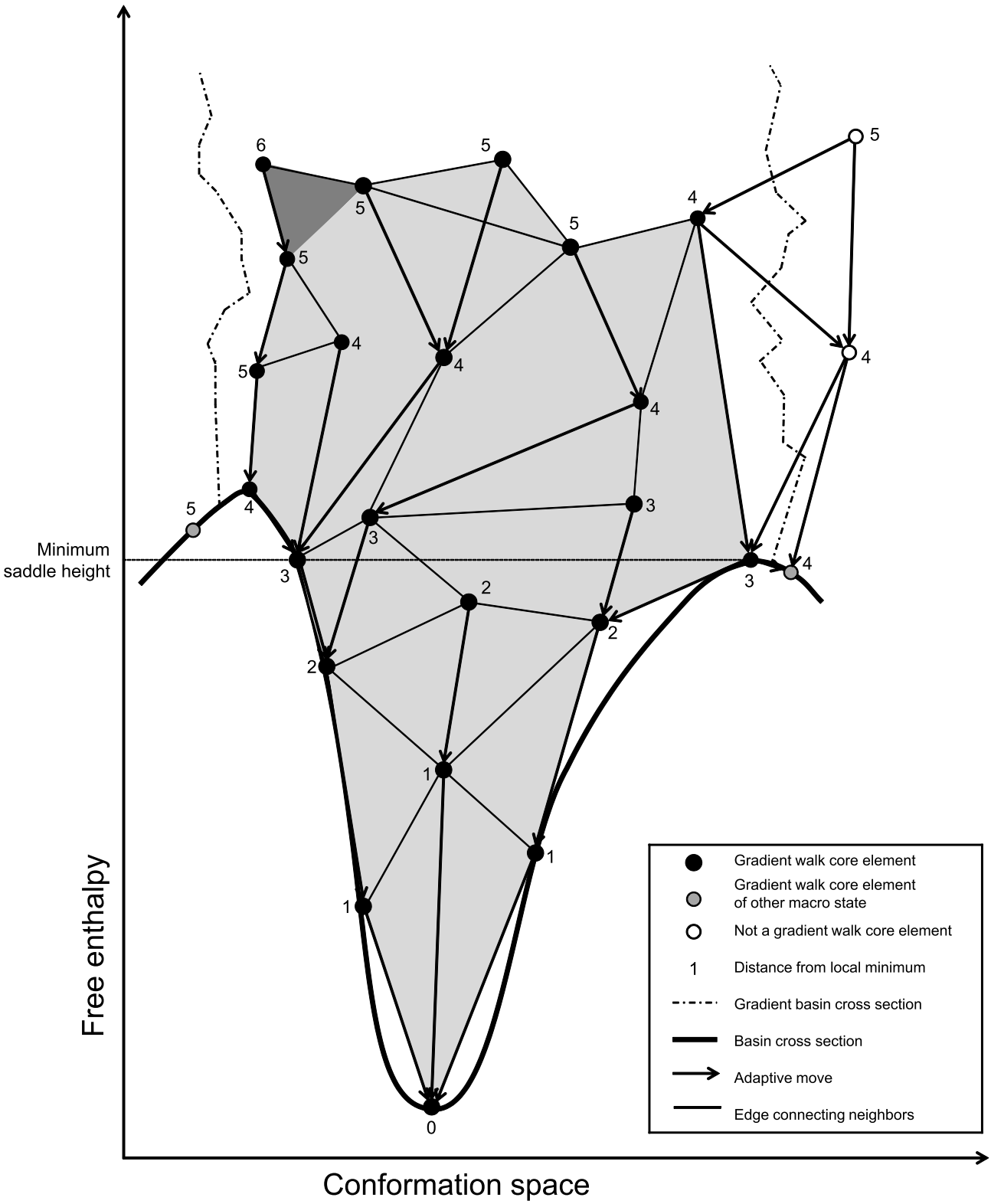

Figure 13: Cross section of energy landscape along the conformational coordinates of a macro state with gradient walk core elements. Filled dots are gradient walk core elements, gray dots are gradient walk core elements of neighboring macro states, white dots are not part of the core complex. Each micro state is labeled with its distance from the local minimum. The light gray part shows the elements within the minimum core radius. The light and dark gray part together cover the elements within the maximum core radius. Different from figure 12 a great part of the minimum core elements lies over the minimum saddle height. The dark gray addition of the maximum core elements is only so small in this simple example. Due to the less harder core definition more elements are part of the core, even the saddle points.

5.3 Adaptations to the definitions of the core elements

First results showed that the minimum radius r_{min} is almost always zero and never much higher than 1, even in the case of highly populated macro states. A short evaluation of the neighbors of local minima reveals that they are often energetic quite high, so the transition rate is quite small. Since it is not very likely to overcome such a barrier, we decided to skip checking of downward walks for those structures and assign them automatically to the core. This will in most cases not only enlarge r_{min}, but also r_{max}. However, the significance of r_{max} is still to be validated.

Set of adaptive walk core elements of a basin B with local minimum m, whereby upwards neighbors higher than a provided energy limit ΔL are declared as core element.

$$\begin{aligned} C_{aw(\Delta L)}(m) = \{x \in \mathcal{C} \mid \exists a \in \mathbb{A} : a_1 = x \wedge a_{|a|} = m \wedge \forall 1 \leq i < |a| : \\ \big((\nexists b \in \mathbb{A} : b_1 = a_i \wedge b_{|b|} \neq m) \vee \\ (\exists b \in \mathbb{A} : b_1 = a_i \wedge b_{|b|} \neq m \wedge (E(a_i) - E(a_{i+1})) \geq \Delta L)\big)\} \end{aligned} \tag{48}$$

Equation 48 should be read as follows. All elements x are part of the core, if there is an adaptive walk a starting in x and ending in local minimum m, and additionally for each element of this walk a if there is no adaptive walk b ending not in m, or if there is such a walk b, if the next element of a is below the energy limit ΔL.
We call this set of vertices the *adaptive walk core elements with upwards energy check*.

Figure 14 depicts the situation. Element B is only part of the core, because $E(B) - E(A) > \Delta L$. The same holds true for element D, because $E(D) - E(C) > \Delta L$. Without the upwards energy limit condition the minimum radius were $r_{min} = 0$, now it is $r_{min} = 2$. The maximum radius has also increased from $r_{max} = 3$ to $r_{max} = 6$.
Note that some parts of the core (complex) can even lie outside the gradient basin.

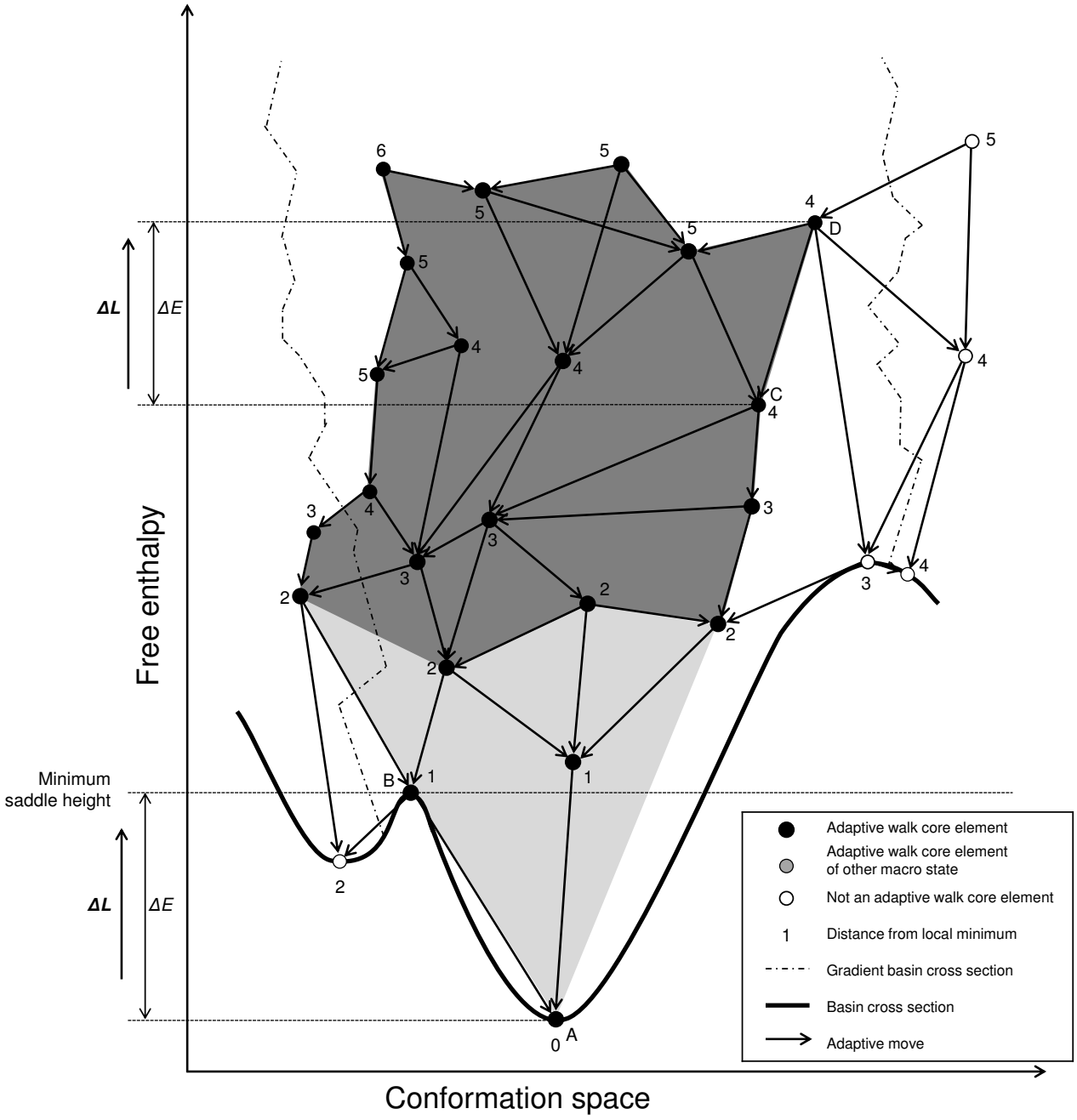

Figure 14: Cross section of energy landscape along the conformational coordinates of a macro state with adaptive walk core elements with upwards energy check. Filled dots are adaptive walk core elements with upwards energy check, white dots are not part of the Core-Complex. Each micro state is labeled with its distance from the local minimum. The light gray part shows the elements within the minimum core radius. The light and dark gray part together cover the elements within the maximum core radius. The chain dotted line marks the border of the gradient basin.

5.4 Algorithms

5.4.1 Gradient walk core element enumeration

Algorithm 1 Gradient walk core element enumeration

1: $T \leftarrow \{ m \}$ ▷ todo list of states to process
2: $C(m) \leftarrow \emptyset$ ▷ core set to fill
3: **while** $T \neq \emptyset$ **do**
4: $x \leftarrow \arg\min_{t \in T}(t)$ ▷ pick smallest element from todo list
5: $C(m) \leftarrow C(m) \cup \{ x \}$ ▷ x is core element
6: **for all** $n \in N(x)$ **do**
7: **if** $x \ll n$ **then** ▷ higher neighbor
8: $g \leftarrow \mathit{gradientWalk}(n)$ ▷ get end of gradient walk starting at n
9: **if** $g = m$ **then**
10: $T \leftarrow T \cup \{ n \}$ ▷ add to T if not contained
11: **end if**
12: **end if**
13: **end for**
14: **end while**
15: report size of $C(m)$

5.4.2 Adaptive walk core element enumeration

Algorithm 2 Adaptive walk core element enumeration

1: $T \leftarrow \{ m \}$ ▷ todo list of states to process
2: $C(m) \leftarrow \emptyset$ ▷ core set to fill
3: **while** $T \neq \emptyset$ **do**
4: $x \leftarrow \arg\min_{t \in T}(t)$ ▷ pick smallest element from todo list
5: *allDownInCore* $\leftarrow$ *1* ▷ assume x is part of core
6: **for all** $n \in N(x)$ **do**
7: **if** $n \ll x$ **then** ▷ lower neighbor
8: **if** $n \notin C(m)$ **then** ▷ check if NOT in core
9: *allDownInCore* $\leftarrow$ *0*
10: **break** ▷ stop neighbor enumeration
11: **end if**
12: **else** ▷ heigher energy neighbor
13: $T \leftarrow T \cup \{ n \}$ ▷ add to T if not contained
14: **end if**
15: **end for**
16: **if** *allDownInCore* $=$ *1* **then**
17: $C(m) \leftarrow C(m) \cup \{ x \}$ ▷ x is core element
18: **end if**
19: **end while**
20: report size of $C(m)$

5.4.3 Adaptive walk core element enumeration with upwards energy check

Algorithm 3 Adaptive walk core element enumeration with upwards energy check

1: $T \leftarrow \{ m \}$ ▷ todo list of states to process
2: $D \leftarrow \{ m \}$ ▷ list of states already added to T (to avoid duplicates and/or circles)
3: $C(m) \leftarrow \emptyset$ ▷ core set to fill
4: **while** $T \neq \emptyset$ **do**
5: $x \leftarrow \arg\min_{t \in T}(t)$ ▷ pick smallest element from todo list
6: *allDownInCore* $\leftarrow$ *1* ▷ assume x is part of core
7: **if** $x \in C(m)$ **then** ▷ check if element is part of core,
8: **goto** *doneCheckingElement* ▷ because it was too high in a previous loop
9: **end if**
10: **for all** $n \in N(x)$ **do** ▷ check all lower neighbors
11: **if** $n \ll x$ **then** ▷ lower neighbor
12: **if** $n \notin C(m)$ **then** ▷ check if NOT in core
13: *allDownInCore* $\leftarrow$ *0*
14: **goto** *doneCheckingElement* ▷ stop neighbor enumeration
15: **end if**
16: **end if**
17: **end for**
18: *doneCheckingElement*:
19: **if** *allDownInCore* $=$ *1* **then**
20: **for all** $n \in N(x)$ **do**
21: **if** $E(n) - E(x) \geq \Delta L$ **then** ▷ check if neighbor is too high
22: $C(m) \leftarrow C(m) \cup \{ x \}$ ▷ add to core without checking its adaptive walks
23: **end if**
24: **if** $n \notin D$ **then** ▷ check if n is or was in T
25: $T \leftarrow T \cup \{ n \}$ ▷ add to T
26: $D \leftarrow D \cup \{ n \}$ ▷ add to D
27: **end if**
28: **end for**
29: $C(m) \leftarrow C(m) \cup \{ x \}$ ▷ x is core element
30: **end if**
31: **end while**
32: report size of $C(m)$

5.4.4 Minimum radius computation

Based on $C(m)$ we can identify the radius r_{min} up to which all vertices are Core-Elements.

Algorithm 4 Minimum radius computation

```
1:  P ← ∅                                          ▷ set of already processed neighbors
2:  A ← N(m)                                       ▷ set of actually processing neighbors
3:  r(m) ← 1                                       ▷ core radius = minimum distance where all states in A
4:  while A ≠ ∅ do
5:      if A ⊆ C(m) then                           ▷ check if all current neighbors are core
6:          P ← P ∪ A                              ▷ mark actual neighbors as processed
7:          r(m) ← r(m) + 1                        ▷ we will extend radius
8:          T ← ∅                                  ▷ temporary set of 'new' neighbors in next radius
9:          for all x ∈ A do
10:             for all n ∈ N(x) do                ▷ increase radius
11:                 if n ∉ P then                  ▷ find all non-processed neighbors = next radius
12:                     T ← T ∪ { x }
13:                 end if
14:             end for
15:         end for
16:         A ← T                                  ▷ make next radius neighbors new actual set
17:     else
18:         r(m) ← r(m) − 1                        ▷ fall back to last core radius
19:         break                                  ▷ stop exploration
20:     end if
21: end while
22: report size of P
23: report core radius r
```

5.4.5 Maximum radius computation

Based on $C(m)$ we can identify the radius r_{max} up to which at least one vertex is a Core-Element.

Algorithm 5 Maximum radius computation

1: $P \leftarrow \emptyset$ ▷ set of already processed neighbors
2: $A \leftarrow N(m)$ ▷ set of actually processing neighbors
3: $r(m) \leftarrow 1$ ▷ core radius = greatest distance of any state in A
4: $N_{max}(m) \leftarrow 0$ ▷ number of Core-Elements
5: **while** $A \neq \emptyset$ **do**
6: $P \leftarrow P \cup A$ ▷ mark actual neighbors as processed
7: $r(m) \leftarrow r(m) + 1$ ▷ we will extend radius
8: $T \leftarrow \emptyset$ ▷ temporary set of 'new' neighbors in next radius
9: **for all** $x \in A$ **do**
10: **if** $x \in C(m)$ **then**
11: $N_{max}(m) \leftarrow N_{max}(m) + 1$ ▷ increment number of core elements
12: **for all** $n \in N(x)$ **do**
13: **if** $n \notin P$ **then** ▷ find all non-processed neighbors = next radius
14: $T \leftarrow T \cup \{ x \}$
15: **end if**
16: **end for**
17: **end if**
18: **end for**
19: $A \leftarrow T$ ▷ make next radius neighbors new actual set
20: **end while**
21: $r(m) \leftarrow r(m) - 1$ ▷ fall back to last core radius
22: report $N_{max}(m)$
23: report core radius r

5.4.6 Minimum and maximum radius computation united

Algorithm 6 Minimum and maximum radius computation united

1: $P \leftarrow \emptyset$ ▷ set of already processed neighbors
2: $A \leftarrow N(m)$ ▷ set of actually processing neighbors
3: $r_{min}(m) \leftarrow 1$ ▷ core radius = greatest distance where all states in A
4: $r_{max}(m) \leftarrow 1$ ▷ core radius = greatest distance of any state in A
5: $N_{min}(m) \leftarrow 0$ ▷ number of core elements within r_{min}
6: $N_{max}(m) \leftarrow 0$ ▷ number of core elements within r_{max}
7: $isMinimumRadiusDefined \leftarrow 0$
8: **while** $A \neq \emptyset$ **do**
9: $P \leftarrow P \cup A$ ▷ mark actual neighbors as processed
10: $r_{max}(m) \leftarrow r_{max}(m) + 1$ ▷ we will extend maximum radius
11: $T \leftarrow \emptyset$ ▷ temporary set of 'new' neighbors in next radius
12: **for all** $x \in A$ **do**
13: **if** $x \in C(m)$ **then**
14: $N_{max}(m) \leftarrow N_{max}(m) + 1$ ▷ increment number of maximum core elements
15: **for all** $n \in N(x)$ **do**
16: **if** $n \notin P$ **then** ▷ find all non-processed neighbors = next radius
17: $T \leftarrow T \cup \{ x \}$
18: **end if**
19: **end for**
20: **end if**
21: **end for**
22: **if** $isMinimumRadiusDefined \neq 0 \wedge A \subseteq C(m)$ **then**
23: $r_{min}(m) \leftarrow r_{min}(m) + 1$ ▷ we will extend radius
24: **else**
25: $isMinimumRadiusDefined \leftarrow 1$
26: $N_{min}(m) \leftarrow P$ ▷ all processed elements until now are minimum core elements
27: **end if**
28: $A \leftarrow T$ ▷ make next radius neighbors new actual set
29: **end while**
30: $r_{min}(m) \leftarrow r_{min}(m) - 1$ ▷ fall back to last minimum core radius
31: $r_{max}(m) \leftarrow r_{max}(m) - 1$ ▷ fall back to last maximum core radius
32: report core radius r_{min}
33: report core radius r_{max}
34: report $N_{min}(m)$
35: report $N_{max}(m)$

6 Computational implementation

6.1 Workflow of computer programs

The following programs were used for the calculations.

- RNAfold [17][24]
  Calculate minimum free energy secondary structures and partition function of RNAs
- RNAsubopt[17][24]
  Calculate suboptimal secondary structures of RNAs
- barriers [11]
  Compute local minima and energy barriers of a landscape
- treekin[35]
  Calculate a macro state dynamics of biopolymers
- ocfinder
  Searches for the open chain in the output of `barriers`.
- corefinder (see section 5.4)
  Enumerates all core elements of a particular type.
  Calculates radii and sizes of cores and core complexes.
- Rcorestat
  Statistical analysis and graphical presentation of the results.
- corestat
  Controls the processing of the workflow.

RNAfold, RNAsubopt, barriers and treekin are part of the *Vienna-RNA-Package*[24].

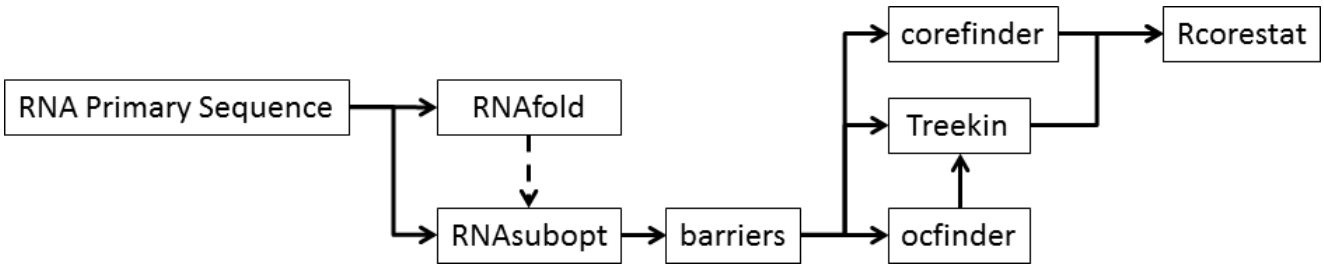

Figure 15: Workflow of corestat

Based upon the output of `barriers` and the chosen parameters the workflow will add four columns to each line of `barriers` (see figure 16).

Minimum radius r_{min}
Number of states within minimum radius $_{min}(F)$
Maximum radius r_{max}
Number of states within maximum radius $N_{max}(F)$

Additionally the number of structures which are (not) part of core complex will be printed out for each radius in the following scheme.

$$r = \begin{cases} \left|\{x \in F(m) \mid d(m,x) = r\}\right| & \textit{if structure is in corecomplex} \\ \left|\{x \in \mathcal{C} \mid d(m,x) = r\} \wedge x \notin F(m)\right| & \textit{if structure is not in corecomplex} \end{cases}$$

The radius *r* starts at zero and ends if no structure is part of the core complex.
The number of lines need not correspond with the maximum radius, because it is possible, that the set of structures with maximum radius have no more neighbors (even such which are not in the core complex), which have not been process in a previous step.

The number of structures in the core complex has to sum up to the number of structures within maximum radius.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	((((((((((((.....))))))))))))	-12.6	0	30	~~~~~~~~~~~~~~~~~~~~~~~~~~~~	558930	0	-12.83	18479	-12.82	0	1	11	1981
Radius	In Core	Not in Core												
0	1	0												
1	11	1												
2	49	17												
3	122	94												
4	205	265												
5	274	478												
6	317	676												
7	314	847												
8	272	925												
9	218	881												
10	146	747												
11	52	507												
12	0	184												

2	(((((....))))).(((((....)))))	-10.1	1	13.2	..........(.....)..........	1660	5626	-10.35	5565	-10.35	1	11	10	1775
Radius	In Core	Not in Core												
0	1	0												
1	10	0												
2	44	2												
3	115	24												
4	218	116												
5	334	340												
6	401	698												
7	347	1059												
8	236	1112												
9	66	832												
10	3	249												
11	0	12												

3	(((((((((.....))))))))))	-8.9	1	0.3	(((((((((..(.....)..)))))))))	1	11	-8.9	77132	-9.29	0	1	13	17305

Figure 16: Extended barriers output of d29-2 as generated by `corefinder` with adaptive walk. Columns 12-15 have been added by `corefinder`. The core complex is calculated without any energy limit, therefor the number of structures in the gradient basin (column 10) does not need to match the number of structures in the core complex (column 15).

7 Data analysis and results

7.1 Methods of data analysis

7.1.1 Methods of data analysis of single RNAs

We call highly populated macro states the *States of Interest (SOI) according to [19]*.
For the cutoff of SOIs we use 20%, 10%, 5% and 1%. Figure 17 displays the population probability of macro states versus time. Most macro states are below 5%, which means that the RNA molecule will not stay for long in any micro state of those macro states.

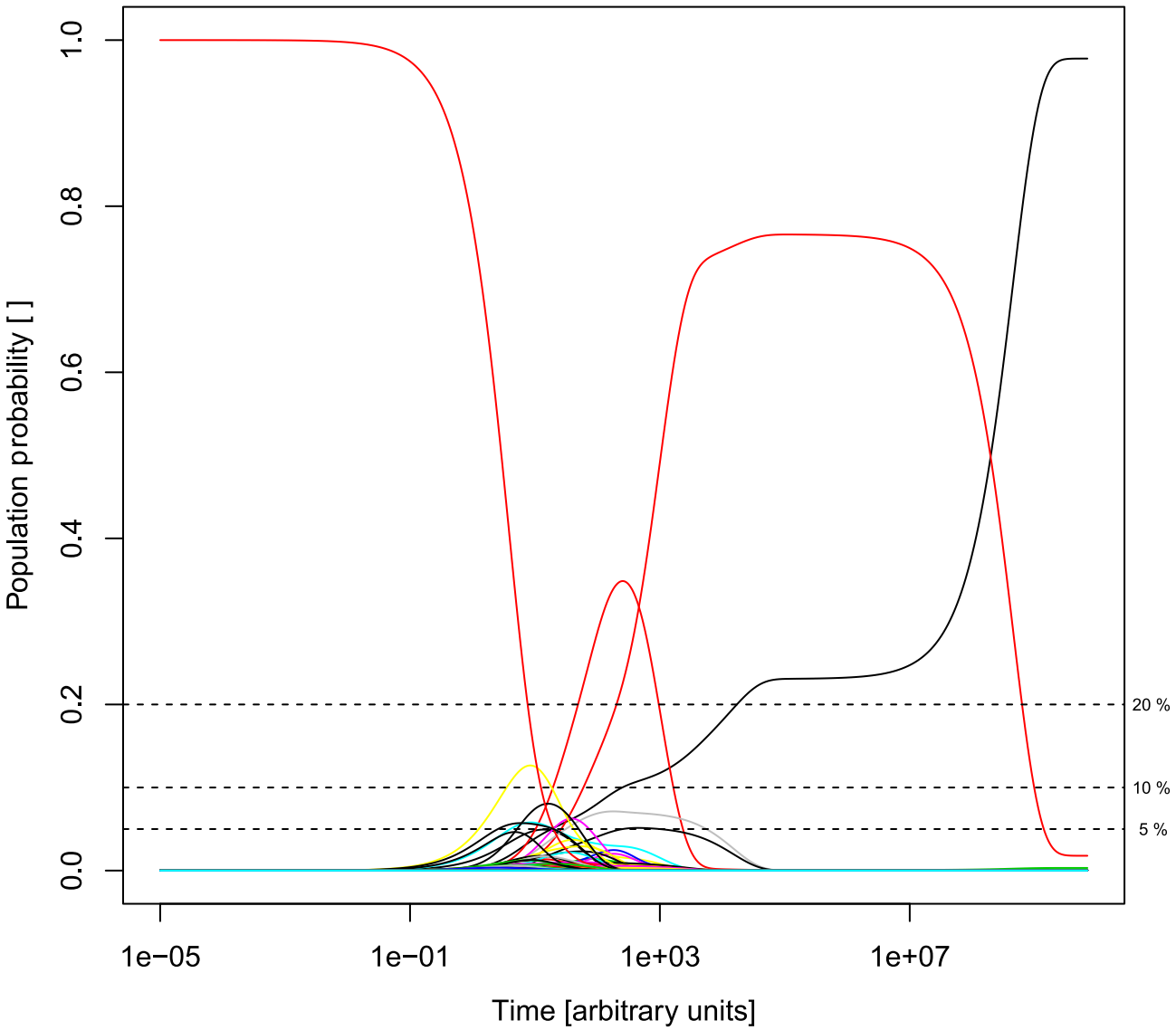

Figure 17: Adaptive walk macro states population versus time

From the output of `Rcorestat` we generate the following statistical test and plots.

- Evolution of probability distribution in time (treekin-like plots) for all methods.
  Solid lines for macro states with core, dashed lines otherwise.
  Labeling of curves with index of macro state and core radius in brackets.
- Distribution of macro states on minimum core radii for all methods.
- Distribution of macro states on maximum core radii for all methods.
- Log of maximum population probability versus energy.
  Color coding of core status. Symbol coding of SOI status.

- Boxplots of energy ranges per minimum core radius.
- Boxplots of energy ranges per maximum core radius.
- Boxplots of log probability ranges per minimum core radius.
- Boxplots of log probability ranges per maximum core radius.

- Table of proportion SOIs which $r_{min} > 1$ for defined SOIs and all methods.

- Table of proportion SOIs which $r_{max} > 1$ for defined SOIs and all methods.

- Table of correlations of N_{min}/N_{max} with energy and maximum population probability for all methods.
- Analysis of variance between groups of core radii by energy and maximum population probability.

7.1.2 Methods of data analysis of many RNAs

Many statistics/plot can be adapted from analysis of one RNA to many RNAs.

Additionally we test the correlation of the GC content and the length of the RNA sequence with

- Proportion of SOIs which have $r_{min} > 0$ and those which have $r_{max} > 0$
- RNA-Sequence-Length
- Summation of N_{min} of all macro states
- Summation of N_{max} of all macro states
- Median and minimum of energy of all macro states with maximum r_{min} and r_{max}
- Median and maximum of maximum probability of all macro states with maximum r_{min} and r_{max}

7.2 Data Analysis of specific RNAs

7.2.1 RNA molecule d29-2

Primary sequence `GGCUGCGUGUAGCCGGACUGUAUGCAGUC`.
All structure up to 30 kcal/mol above the global minimum have been processes, resulting in 382 macro states.
The figures display the temporal development of the population probability of the macro states. Macro states with core are shown as continuous lines, macro states without core are shown as dashed lines. For macro states populated more than 10 % their index and minimum core radius (in brackets) is displayed.

Figure 18 show plots based on the data generated by `treekin`. Macro states with core are represented as continuous lines, macro states without core as dashed lines. For macro states populated more than 10 % their index and minimum core radius (in brackets) is displayed. The plots reveal the sharpness of the different core definitions. We see the best results in lower plot showing that the (meta-)stable switching macro states 1 and 2 have a core, whereas the open chain 74 has not. Aside from the necessity of adjusting the upwards energy limit, we need to check why macro state 10 is higher populated, has no adaptive walk core, but a gradient walk core (maybe it is important for the switching initiation).

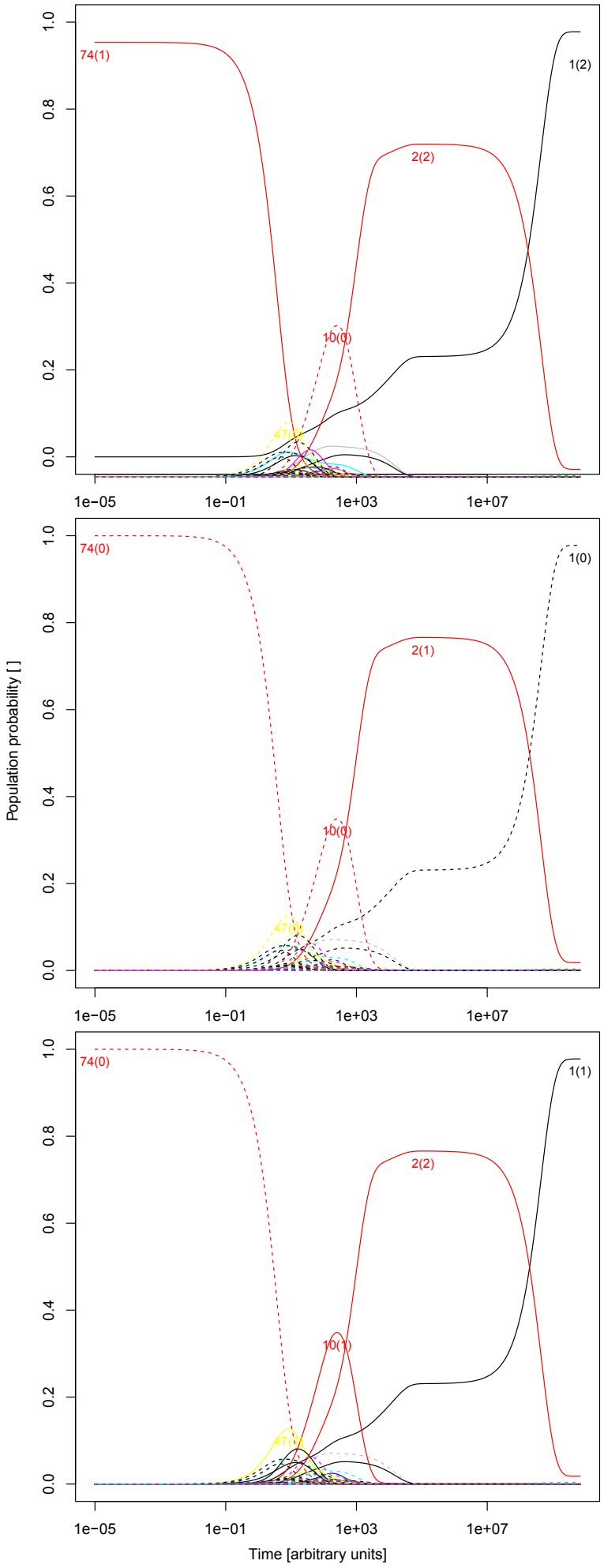

Figure 18: Macro states population versus time with minimum radii

Figure 19 shows the distribution of macro states on different minimum core radii for each core definition. The middle plot shows that the adaptive walk core definition is too hard for calculation of minimum core radii.

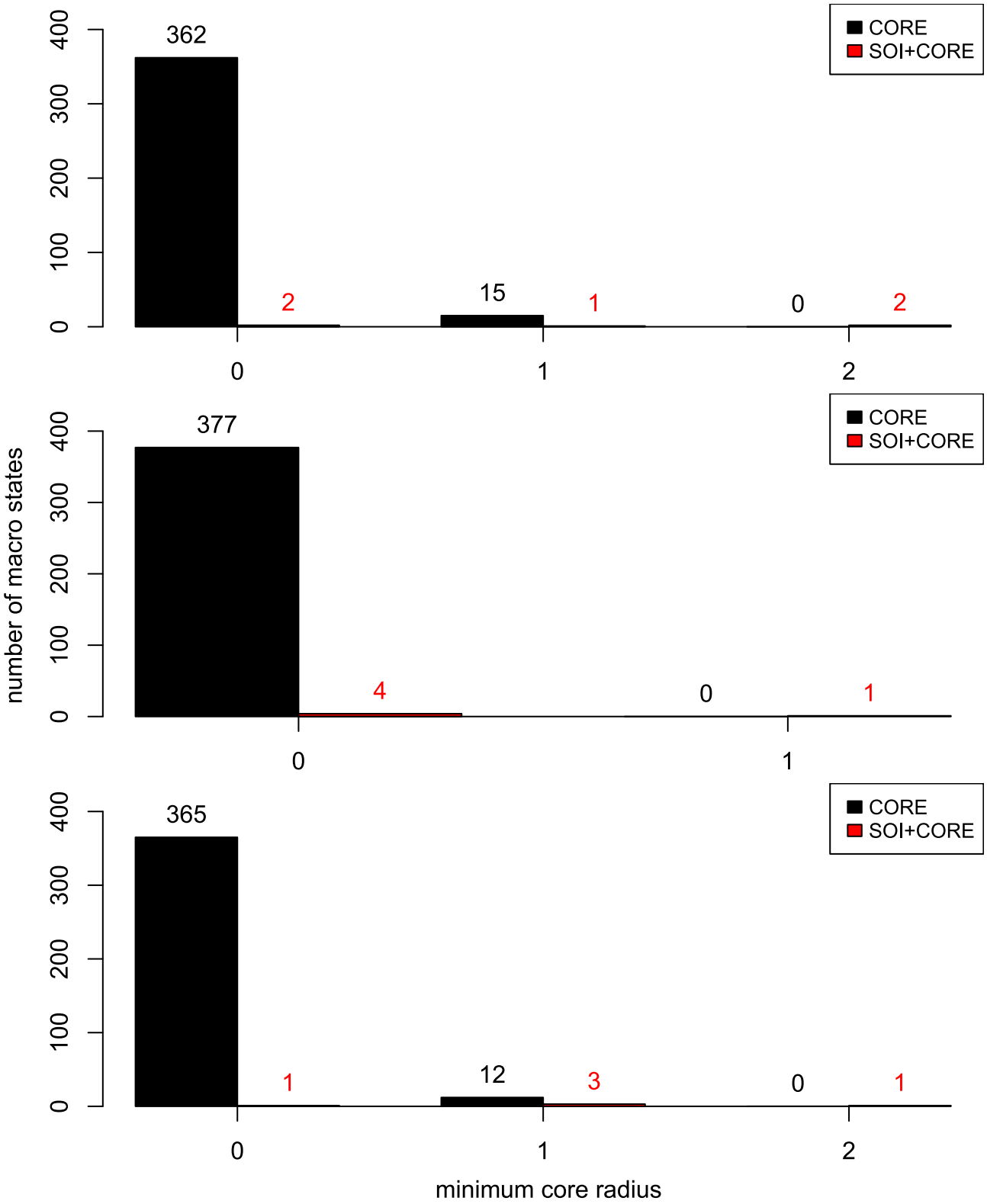

Figure 19: Minimum core radii of different methods

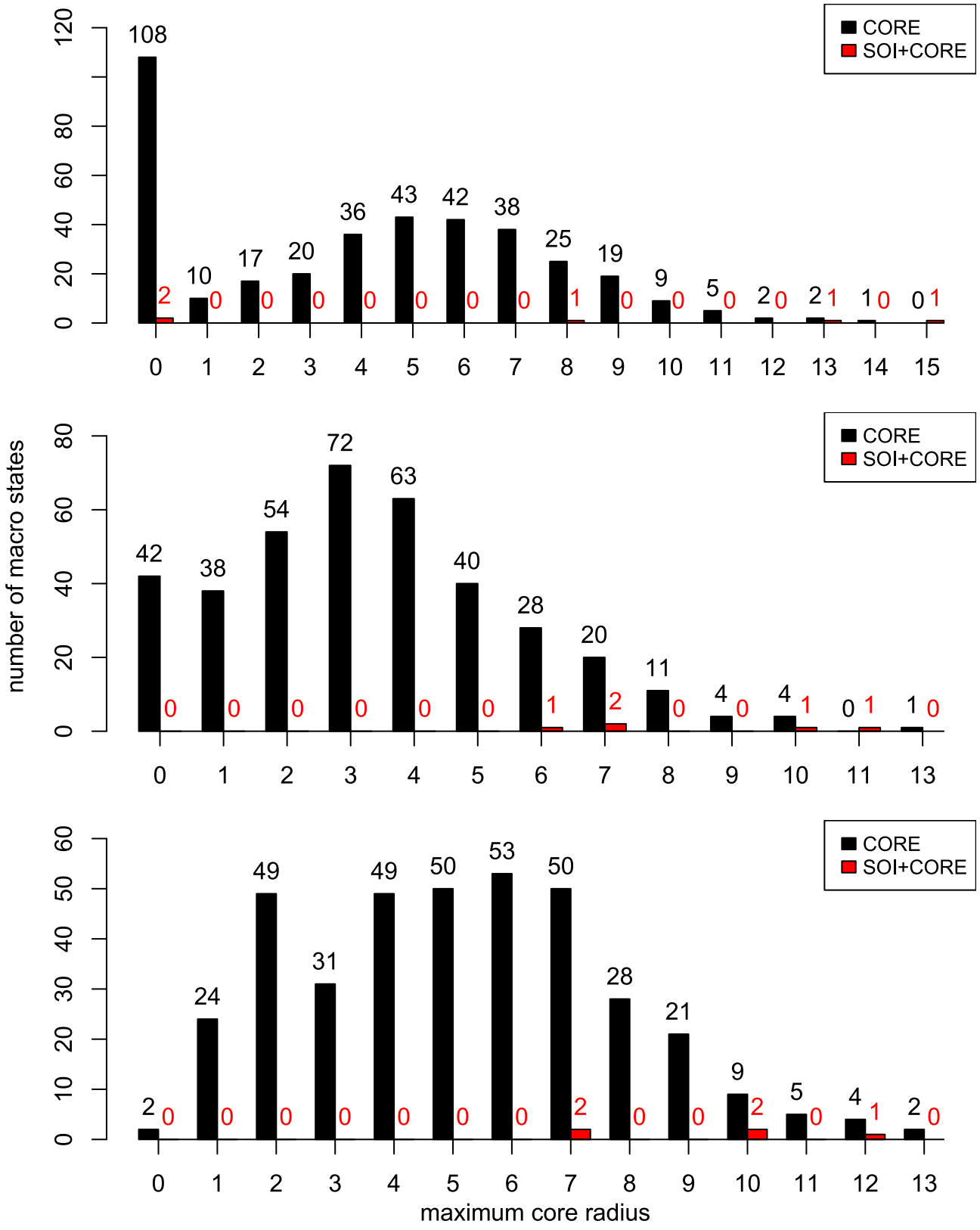

Figure 20: Maximum core radii of different methods

Figure 20 shows the distribution of macro states on different maximum core radii for each core definition. The influence of the core definition on the maximum radius is weak, only the weak gradient walk core definition can be seen as a slightly larger greatest maximum core radius.

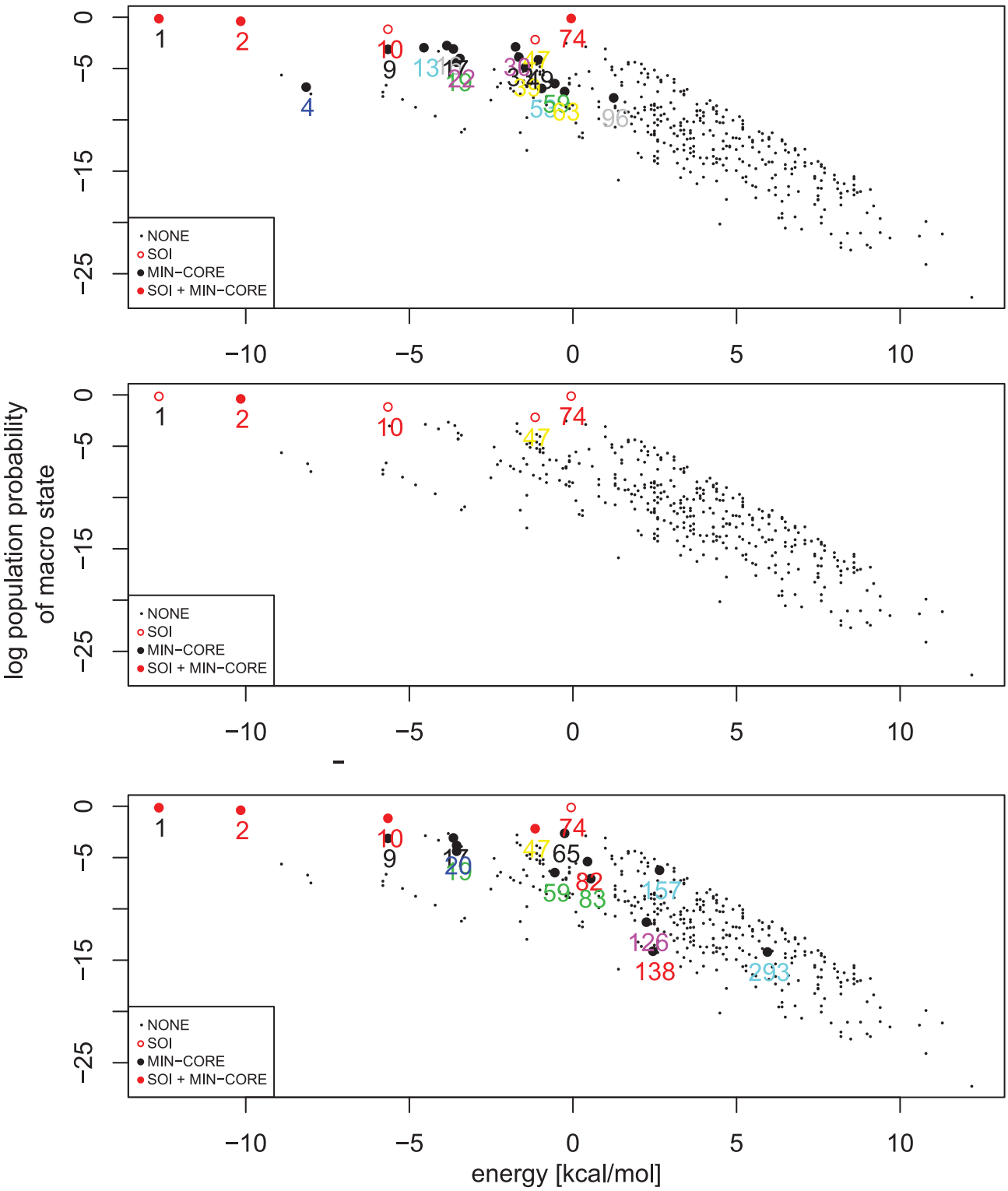

Figure 21: Energy versus logarithmic maximum population plots of different methods

Figure 21 are plots of energy versus logarithmic maximum population probability. Higher populated macro states are colored red, otherwise black. Macro states with core are disks, otherwise circles. In the middle plot we see again the hard adaptive walk core definition resulting in only one macro state with core, which is at least highly populated. The results of the gradient walk cores seems to be in better accordance (see upper plot), although the open chain (index 74) features also a core. In the lower plot the open chain has no core, but all other highly populated macro states. Although this is the best result, more in particular energetically unstable states also feature a core (indices 126,138,157,293).

Threshold	20%	10%	5%	2%	1%
Gradient walk	0.75	0.6	0.6363636	0.5263158	0.4137931
Adaptive walk	0.25	0.2	0.09090909	0.05263158	0.03448276
Adaptive walk energy check	0.75	0.8	0.5454545	0.4210526	0.3103448

Table 2: Percent of SOI macro states with $r_{min} > 1$

Threshold	20%	10%	5%	2%	1%
Gradient walk	0.75	0.6	0.7272727	0.7368421	0.6206897
Adaptive walk	1.0	1.0	1.00	1.00	0.9310345
Adaptive walk energy check	1.0	1.0	1.00	1.00	1.00

Table 3: Percent of SOI macro states with $r_{max} > 1$

Tables 2 and 3 summarize the proportion of SOIs which are also core elements. In table 2 we see good accordance and its decrease as the threshold goes lower. The values of the gradient walk method are too high, but the adaptive walk method does not look too strict here (actually the adaptive walk energy check method is too weak).
The weak accordance of the gradient walk method in table 3 is due to the fact, that there is only one gradient walk for each structure.

	Energy		**Maximum probability**	
	N_{min}	N_{max}	N_{min}	N_{max}
Gradient walk	-0.2967136	-0.1702163	0.8830793	0.5925826
Adaptive walk	-0.1844292	-0.3367850	0.4625818	0.3616787
Adaptive walk energy check	-0.2977745	-0.1254210	0.4427226	0.2834822

Table 4: Correlation of energy and maximum probability with N_{min} and N_{max} of macro states.

Table 4 summarizes the correlation of energy and maximum population probability with the number of core states. The correlation is weak or almost not existing. Only the gradient walk method, which represents best the size of the gradient basin, has a higher value. The degree of attraction of cores obtain by the other methods seems to be not so strong.

Significance of core radii

To test if the categorization of macro states in groups of different core radii is corresponding to the ranges of energy or maximum population probability we used the Analysis of Variance (ANOVA) or if the requirements are not met, the Kruskal–Wallis one-way analysis of variance. All results depict that r_{min} and r_{max} are not randomly assigned to macro states.
Figures 22 - 25 show the distribution of ranges for each core definition. The variance of the categories defined by minimum core radii is more significant than those defined by maximum core radii, but this is also due to the fact, that there are more maximum core radii.

Figure 22 and 23 show distribution of energy ranges.
Figure 24 and 25 show the distribution of maximum probability ranges. Mind the logarithmic scale of population probability. The best result can be seen on the lower plot of figure 25, which also reveals some outliers on maximum core radius 7.

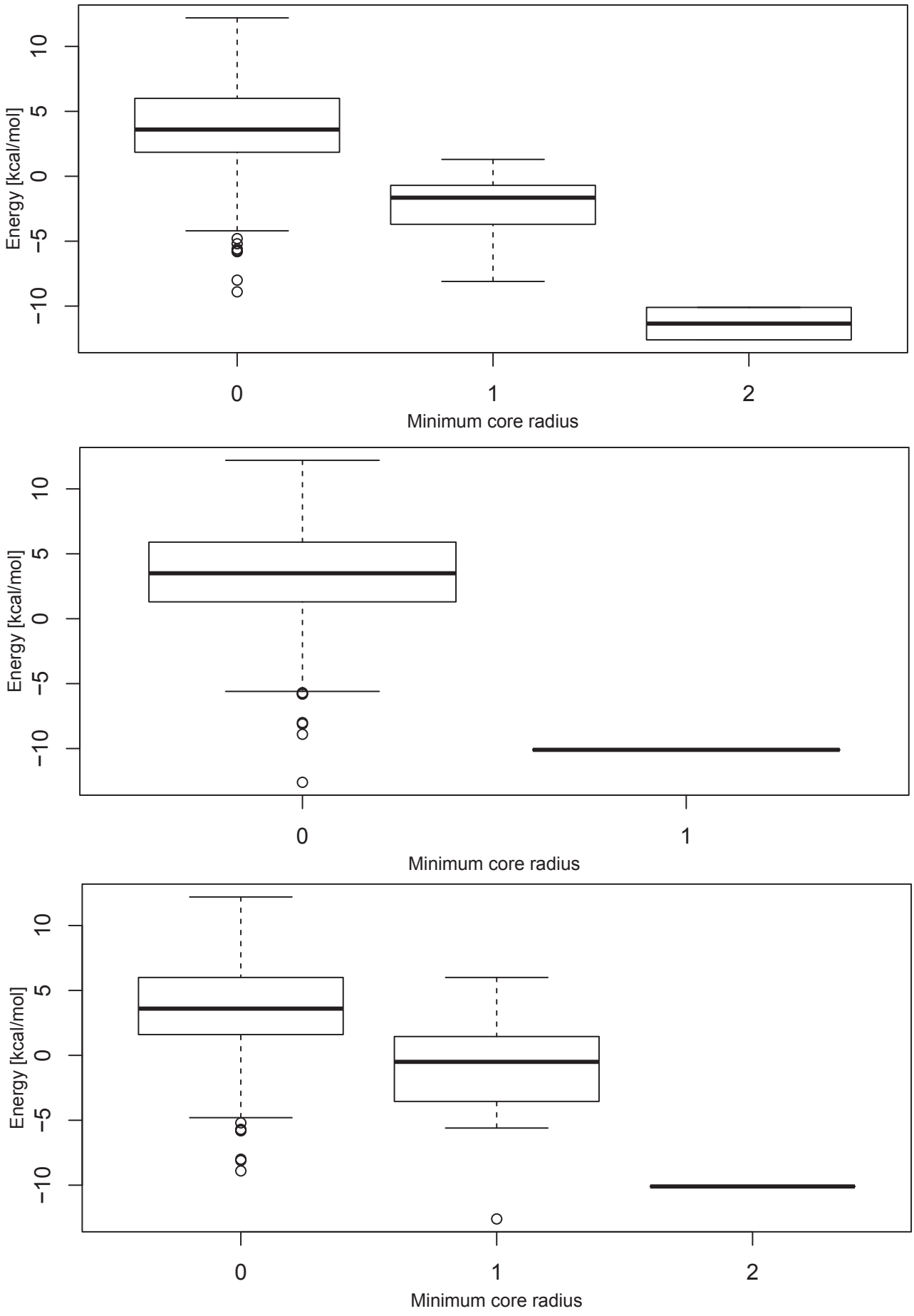

Figure 22: Energy ranges per minimum core radius

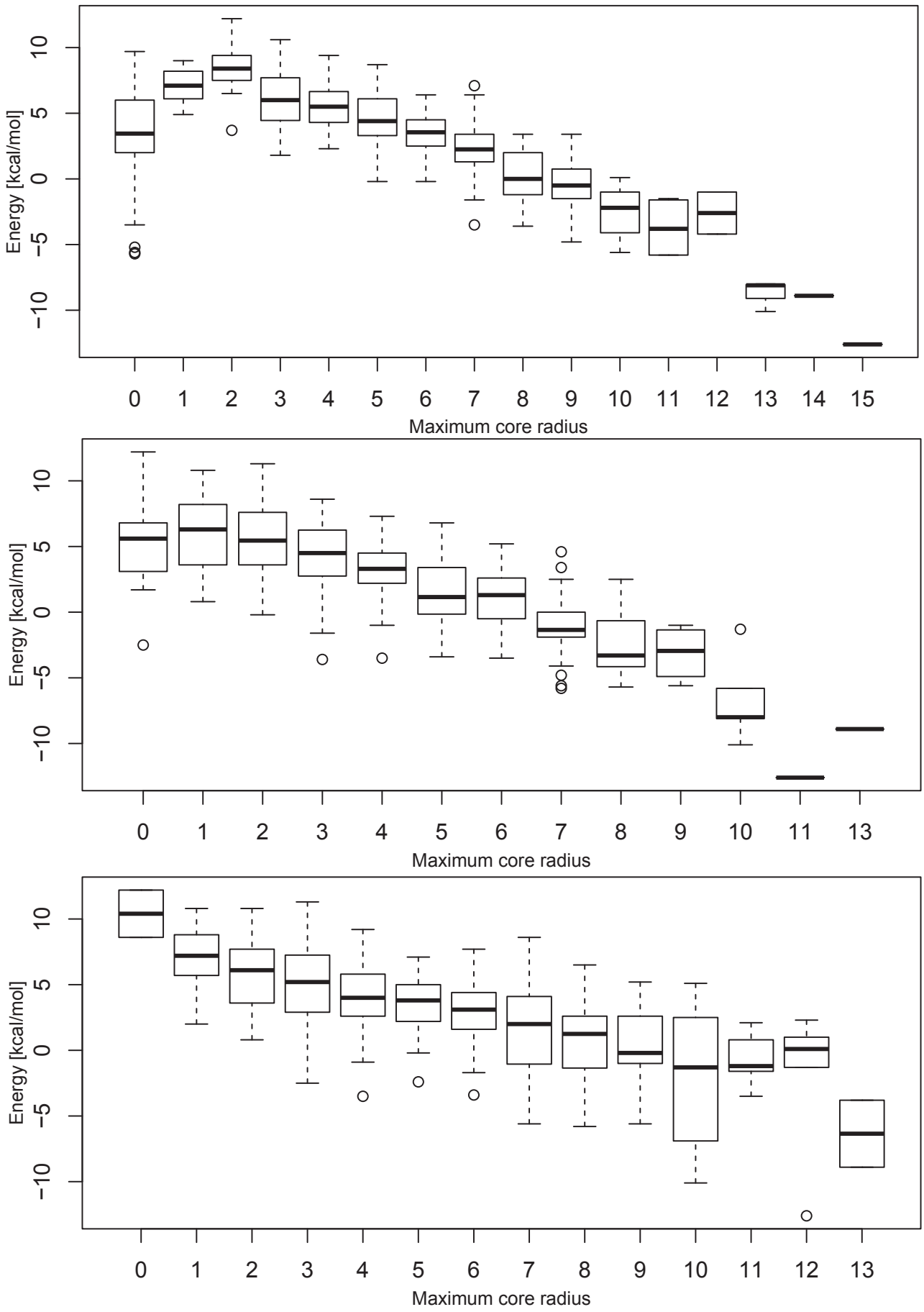

Figure 23: Energy ranges per maximum core radius

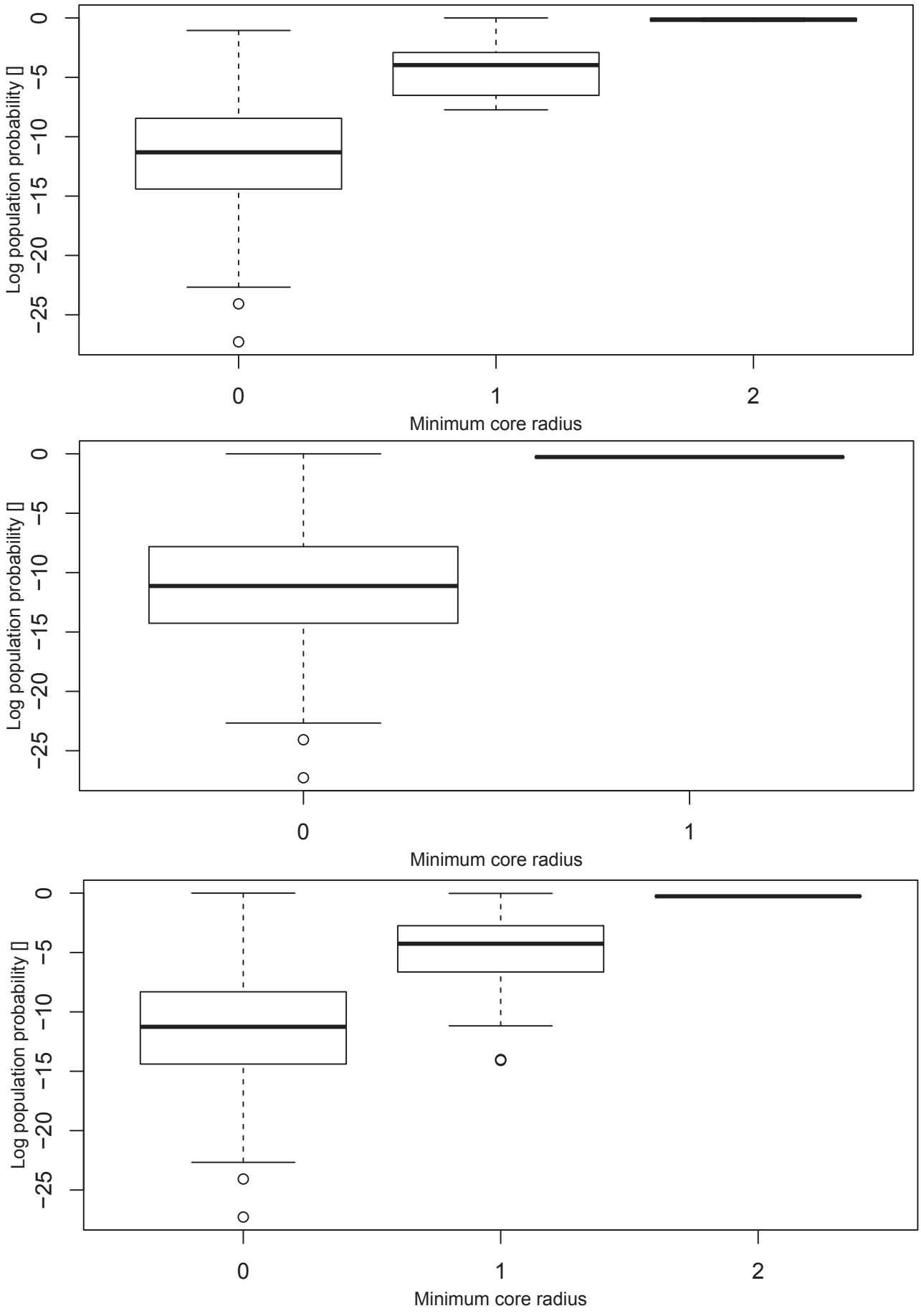

Figure 24: Log probability ranges per minimum core radius

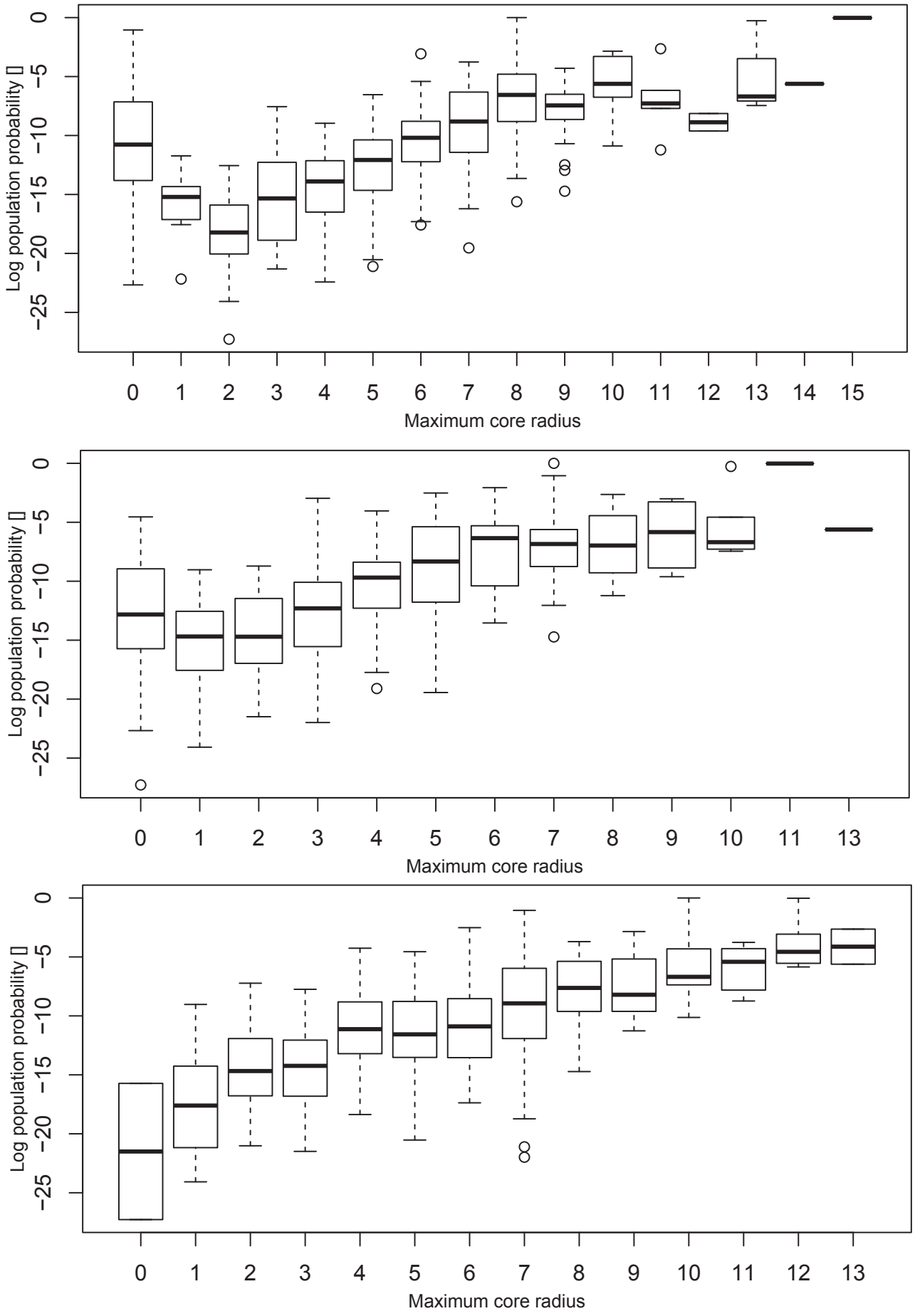

Figure 25: Log probability ranges per maximum core radius

7.3 Data analysis of many RNAs

To test the influence of properties of RNA on the core parameters we generated for each length of 20 to 30 base pairs 100 random RNA sequences.

The strongest correlation was between the GC content and the lowest energies of all macro states with core per sequence.

Minimum energy of macro states with greates r_{min}	-0.436738
Minimum energy of macro states with greates r_{max}	-0.3751044
Maximum probability macro states with greates r_{min}	0.03419664
Maximum probability macro states with greates r_{max}	-0.3751044

Table 5: Correlations of GC content obtained by random RNA sampling

The missing of correlation could be due to bad quality of the samples. The GC content follows a normal distribution, not a uniform distribution.

8 Conclusion and outlook

It has been shown that there are topological properties of the energy landscape which are related to the population probability of macro states. Our results show, that macro states with core tend to be more stable and hence higher populated than that without core. In particular macro states with $r_{min} > 0$ are always highly populated and macro states with $r_{max} = 0$ are never highly populated.
Different types of stability criteria have been used to search for core elements, whereby the adaptive walk with energy check yielded the best results in consideration of the fact, that the open chain (which was only highly populated in their function as start structure) had never and the (meta) stable chains had always a core. This revealed also that the energy parameter can of course not be neglected when testing the stability of macro states.

8.1 Computational improvement

The first step is the reduction of complexity of the computational implementation by avoiding the flooding ansatz. Particularly the algorithm of scanning the core complexes works by generating all neighbors recursively until no core element is found.
The current implementation of the algorithms is also suboptimal regarding memory management resulting in huge memory consumption. The problem is that one of the fastest methods to look up an element in a list consists of the use of associative arrays (hashes), which require thorough memory management. Currently we use an array to store elements of predefined size so that the pointers in the hash table point to addresses in the array. Instead some methods proposed in [25] should be used.
Even for longer sequences and/or greater energy ranges the implementation should make use of multi (core-)processor systems. The next step will be to obtain those macro states a priori without the use of any information provided by other programs, to really reduce the computational effort.

8.2 Methodical extension

The number of structures contained in a core (N_{min} and N_{max}) is not correlated to their radii, but we could calculate the free enthalpy of the partition function summation of only the core elements and use these values in the Arrhenius kinetics(see 4.4.1). The same idea can be applied to barrier tree kinetics (see 4.4.2) by using only core elements as transition structures (in the case of greater macro states only elements of the core complex will have transitions to other macro states). This would reveal if the folding dynamics can really be reduced to those few trajectories. Even the Monte Carlo method used in [9] (see 4.3) could be restricted to core elements. We will continue to look for additional adaptations of the core definitions to find algorithms, which are better capturing the stability of the macro states. Maybe it could even be possible to check some properties of the RNA molecule and then decide which algorithm is the best to use.

The models based upon reduced folding kinetics have been extended by parameters providing information about the stability of macro states. The restriction of the folding process on macro states which are depicted as stable seems to be a viable option to improve this method of RNA dynamics prediction. Although the model involves many assumptions and approximations, we discovered one more feature on the way asymptotically converging towards the perfect simulation of the RNA folding process.

Epilogue

Do not use LaTeX if you do not need to!
The quality of this work has been deteriorated due to the use of LaTeX. Half of the time have been spent dealing with problems caused by LaTeX. Although the concept behind LaTeX might not be bad, the realization is awful. LaTeXis like a program from the stone age of computation. For almost every function you need an additional package. The style which seems to have been introduced by LaTeX should apparently suggests more professionalism, but actually it is opposed to environmental protection and contradicts the methods of effective data storage.

Danksagung

Der Verfasser dieser Arbeit dankt allen die meinen etwas zu ihrem Gelingen beigetragen zu haben.
Einen besonderen Dank richte ich an meinen Hauptbetreuer Herrn Professor Wolfinger, der sich mit meinem ausgeprägten Individualismus herumschlagen musste und Herrn Professor Anton Beyer.
Weiters an Martin Mann für Seine Vorschläge und wertvollen Kommentare.
An Ronny Lorenz für Seine Hilfe in Fragen der Programmierung.
Nicht genug danken kann ich meiner Familie, die mir diesen Bildungsweg überhaupt erst ermöglicht hat.
Außerdem danke ich all den rechnenden Maschinen und Ihren Peripheriegeräten, insbesondere meinem Laptop, meinem PC und den Rechnern am Institut für Theoretische Chemie.

List of abbreviations and symbols

$\prec_{lex}$	Lexicographical predecessor
$\lll$	Energetic-Lexicographical predecessor
$x \overset{\eta}{\leftrightarrow} y$	Accessibility of y from x at height η
$\mathbb{A}$	Set of adaptive walks
$B(m)$	Basin of m
$\mathcal{B}(m)$	Gradient basin of m
$C_{aw}(m)$	Set of adaptive walk core elements (structures) of m
$C_{aw(\triangle L)}(m)$	Set of adaptive walk core elements (structures) with energy check of m
$C_{gw}(m)$	Set of gradient walk core elements (structures) of m
$\mathcal{C}(p)$	Conformation space of p
$d(x, y)$	Distance between x and y
DNA	Deoxyribonucleic acid
E	Potential function
$\mathcal{E}$	Set of edges
$\mathfrak{E}$	Energy landscape
F	Set of vertices with particular feature
$F(s)$	Free enthalpy of secondary structure s
$\hat{f}(x, y)$	Saddle height between x and y
g	Density of state
G	Free enthalpy / Gibbs (free) energy
$\mathbb{G}$	Set of gradient walks
$\mathcal{G}$	Graph
H	Enthalpy
k_B	Boltzmann constant
k_{ij}	Transition rate
L	Loop
	Limit (in the term $C_{aw(\triangle L)}(m)$)
μ	Move function
$\mathcal{M}_k$	Move set of size k
$\mathbb{M}$	Set of local minima
MFE	Minimum Free Energy
$N(x)$	Neighborhood of x
$\mathcal{N}(x)$	Set of adjacent vertices of x
N_A	Avogadro constant
N_{min}	Number of vertices within r_{min}
N_{max}	Number of vertices within r_{max}
Ω	Set of possible base pairings
p	Primary sequence / Primary structure
	Pressure (in the term pV)
p_{ij}	Transition probability
$\mathcal{P}$	Power set
	Probability (in the term $\langle\mathcal{P}\rangle_{eq}$)
P_t	Temporal distribution vector
$\mathbb{P}_{xy}$	Set of walks from x to y
r_{min}	Minimum core radius
r_{max}	Maximum core radius
R	Gas constant
$\mathfrak{R}(m)$	Radii set of m
RNA	Ribonucleic acid
rRNA	Ribosomal RNA

s	Secondary structure
S	Entropy (in the term $T\triangle S$)
$\mathcal{S}(p)$	Set of secondary structures compatible with p
$\mathfrak{S}_p(n)$	Set of primary sequences of length n
$\mathfrak{S}_s(p)$	Set of possible base pairings of p
$\bar{S}(m,n)$	Set of lowest saddle points between m and n
SOI	State of interest
T	Temperature
tRNA	Transfer RNA
u_{ij}	Transition matrix
U	Inner energy
V	Volume (in the term pV)
$\mathbb{V}(\bar{s})$	Valley of $\bar{s}$
$\mathcal{V}$	Set of vertices
$\mathbb{W}$	Set of walks
Z	Partition function

List of figures

List of tables

Bibliography

[1] American Standards Association. American Standard Code for Information Interchange. 1963.

[2] R.H. Austin, K.W. Beeson, L. Eisenstein, H. Frauenfelder, and I.C. Gunsalus. Dynamics of Ligand Binding to Myoglobin. *Biochemistry*, 14(24):5355--5373, 1975.

[3] Richard Bellman. *Dynamic Programming*. Princeton University Press, 1957.

[4] Richard Ernest Bellman. *Eye of the Hurricane: an autobiography*. World Scientific Singapore, 1984.

[5] Jan Cupal, Ivo L. Hofacker, and Peter F. Stadler. Dynamic programming algorithm for the density of states of RNA secondary structures. 1996.

[6] René Descartes. *Discours sur la méthode pour bien conduire sa raison et chercher la verité dans les sciences*. 1637.

[7] Manfred Eigen. The Hypercycle: A Principle of Natural Self-Organization. *International Journal of Quantum Chemistry*, 14(S5):219--219, 1978.

[8] Christoph Flamm. Kinetic Folding of RNA. 1998. PhD Dissertation, University of Vienna.

[9] Christoph Flamm, Walter Fontana, Ivo L. Hofacker, and Peter Schuster. RNA folding at elementary step resolution. *Rna*, 6(3):325--338, 2000.

[10] Christoph Flamm, Ivo L. Hofacker, Bärbel M.R. Stadler, and Peter F. Stadler. Saddles and Barrier in Landscapes of Generalized Search Operators. pages 194--212, 2007.

[11] Christoph Flamm, Ivo L. Hofacker, Peter F. Stadler, and Michael T. Wolfinger. Barrier trees of degenerate landscapes. *Zeitschrift für Physikalische Chemie International journal of research in physical chemistry and chemical physics*, 216(2/2002):155, 2002.

[12] Robert L. Forward. Dragon's Egg. 1980. Science Fiction.

[13] Hans Frauenfelder, Stephen G. Sligar, and Peter G. Wolynes. The energy landscapes and motions of proteins. *Science*, 254(5038):1598--1603, 1991.

[14] Crispin W. Gardiner et al. Handbook of Stochastic Methods. 3, 1985.

[15] W. Gilbert. The RNA World. *Nature*, 319:618, 1986.

[16] Johanne Hizanidis. The Master Equation. 2002.

[17] Ivo L Hofacker, Walter Fontana, Peter F Stadler, L Sebastian Bonhoeffer, Manfred Tacker, and Peter Schuster. Fast Folding and Comparison of RNA Secondary Structures. *Monatshefte für Chemie/Chemical Monthly*, 125(2):167--188, 1994.

[18] Karl Heinz Hoffmann and Paolo Sibani. Diffusion in hierarchies. *Phys. Rev. A*, 38:4261--4270, Oct 1988. (accessed on 2014-05-02).

[19] Bettina Huebner. Pruning strategies for large energy landscapes. 2013. Master's thesis, Albert-Ludwigs-Universität Freiburg.

[20] IUPAC. Compendium of Chemical Terminology, 2nd ed. (the "Gold Book"). Compiled by A. D. McNaught and A. Wilkinson. *Blackwell Scientific Publications*, 1997.

[21] Helmut G. Katzgraber. Introduction to Monte Carlo Methods. Technical report, 2010.

[22] Hannes Kochniß. Ein Hybridkinetik Ansatz für RNA Faltungswahrscheinlichkeiten. 2008. Master's thesis, Friedrich-Schiller-Universität Jena.

[23] Shu-Qun Liu, Xing-Lai Ji, Yan Tao, De-Yong Tan, Ke-Qin Zhang, and Yun-Xin Fu. Protein Folding, Binding and Energy Landscape: A Synthesis. *Protein engineering*, pages 207--252, 2012.

[24] R. Lorenz, S.H. Bernhart, C. Hoener zu Siederdissen, H. Tafer, C. Flamm, P.F. Stadler, and I.L. Hofacker. ViennaRNA Package 2.0. *Algorithms for Molecular Biology*, 6(1):26,

2011.

[25] Martin Mann, Marcel Kucharik, Christoph Flamm, and Michael T. Wolfinger. Memory-efficient RNA energy landscape exploration. *Bioinformatics*, 2014.

[26] Daniel Maticzka. Kinetiken von RNA-RNA Hybridisierungen. 2009. Master's thesis, Albert-Ludwigs-Universität Freiburg.

[27] John S. McCaskill. The Equilibrium Partition Function and Base Pair Binding Probabilities for RNA Secondary Structure. *Biopolymers*, 29(6-7):1105--1119, 1990.

[28] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of State Calculations by Fast Computing Machines. *The journal of chemical physics*, 21(6):1087--1092, 2004.

[29] Ruth Nussinov, George Pieczenik, R. Jerrold Griggs, and Daniel J. Kleitman. Algorithms for Loop Matchings. *SIAM Journal on Applied Mathematics*, 35(1), 1981.

[30] Wolfram Saenger. Principles of Nucleic Acid Structure. 1984.

[31] Bärbel M.R. Stadler and Peter F. Stadler. Combinatorial vector fields and the valley structure of fitness landscapes. *Journal of mathematical biology*, 61(6):877--898, 2010.

[32] Alejandro J. Vila. Plegamiento de proteinas. 2013. University of Rosario - Facultad de Ciencias Bioquímicas y Farmacéuticas.

[33] Michael Wolfinger. The Energy Landscape of RNA Folding. 2001. Master's thesis, University of Vienna.

[34] Michael Wolfinger. Energy Landscapes of Biopolymers. 2004. PhD Dissertation, University of Vienna.

[35] Michael T. Wolfinger, W. Andreas Svrcek-Seiler, Christoph Flamm, Ivo L. Hofacker, and Peter F. Stadler. Efficient computation of RNA folding dynamics. *Journal of Physics A: Mathematical and General*, 37(17):4731, 2004.

[36] Stefan Wuchty, Walter Fontana, Ivo L Hofacker, and Peter et alia Schuster. Complete Suboptimal Folding of RNA and the Stability of Secondary Structures. *Biopolymers*, 49(2):145--165, 1999.

[37] T. Xia, J. SantaLucia Jr., M.E. Burkard, R. Kierzek, S.J. Schroeder, X. Jiao, C. Cox, and D.H. Turner. Thermodynamic Parameters for an Expanded Nearest-Neighbor Model for Formation of RNA Duplexes with Watson-Crick Base Pairs. *Biochemistry*, 37(42):14719, 1998.

[38] M. Zuker. On Finding All Suboptimal Foldings of an RNA Molecule. *Science (New York, NY)*, 244(4900):48, 1989.

[39] Michael Zuker and David Sankoff. RNA secondary structures and their prediction. *Bulletin of mathematical biology*, 46(4):591--621, 1984.

[40] Michael Zuker and Patrick Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic acids research*, 9(1):133--148, 1981.