Entropy Functions and Rare Events in Disordered Systems by Transfer Matrix Calculations and Monte Carlo Sampling

Von der Fakultät V (Mathematik und Naturwissenschaften) der Carl von Ossietzky Universität Oldenburg zur Erlangung des Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

angenommene Dissertation

von

Stefan Wolfsheimer

geboren am 20.09.1977 in Frankfurt am Main

Gutachter:	Prof. Dr. Alexander K. Hartmann
Zweitgutachter:	Prof. Dr. Andreas Engel
Tag der Disputation:	27.02.2009

Contents

Zusammenfassung

In der vorliegenden Arbeit wurden Fragestellungen der Bioinformatik mit Monte Carlo Verfahren der statistischen Physik behandelt.

Beim Vergleich von molekularen Sequenzen (Sequenz Alignment) verwendet man statistische Tests, um die Signifikanz beobachteter Ähnlichkeiten zu quantifizieren. Verteilung der optimalen lokalen Alignment-Scores über Zufallssequenzen, insbesondere seltene Ereignisse, sind ein wichtiger Bestandteil solcher Tests. Ich erweiterte eine bereits bestehende Arbeit, in der große Abweichungen von der theoretisch vorhergesagten Gumbel-Verteilung gefunden wurde, auf weitere biologisch relevante Protein- und Score-Modelle. In den meisten Fällen konnten die Abweichungen durch eine heuristisch modifizierte Gumbel-Verteilung beschrieben werden. Sie sind teilweise so groß, dass einige signifikante Ähnlichkeiten in der bisherigen Praxis nicht als solche klassifiziert werden. Dies kann eintreten, wenn man ein Suchergebnis weiter verfeinern möchte. Zuerst betrachtete ich die Verteilung bezüglich des Standardmodells für Proteinsequenzen für verschiedene Alignment-Parameter. In einem zweiten Schritt untersuchte ich ein Modell, das Transmembran-Proteine beschreibt. Außerdem studierte ich Verteilungen freier Energien kanonischer Alignment-Ensembles. Die temperaturabhängige Form dieser Verteilungen deutete ich anhand des linear-logarithmischen Phasenübergangs, der in diesem Modell auftritt.

In einer ähnlichen Weise untersuchte ich RNA Sekundärstrukturen. Hier wurde die Verteilung der minimalen freien Energie ebenfalls über Zufallssequenzen bestimmt. Mit diesen Verteilungen konnte ich biologische RNA Sequenzen gegen Zufallsmodelle vergleichen. Dazu betrachtete ich mikrokanonische Sequenzensembles und verglich deren statistische Eigenschaften mit biologischen RNA Sequenzen aus einer Datenbank.

Auch für eine Studie der Monte Carlo Dynamik in komplexen Energielandschaften betrachtete ich RNA Sekundärstrukturen. Diese stellen für solche Fragestellungen ein ideales Modell dar, da es, im Gegensatz zu vielen anderen Modellen, exakt behandelt werden kann und gleichzeit komplexe glassartige Eigenschaften besitzt. Ich verglich dynamische Eigenschaften verschiedener Monte Carlo Algorithmen mit statischen Eigenschaften, die durch Transfermatrix-Berechnungen zugänglich sind.

Abstract

This thesis treats problems from bioinformatics with Monte Carlo methods from statistical physics.

Methods to compare molecular sequences (sequence alignment) make use of statistical tests to assess the significance of observed similarities. Distributions of optimal alignment scores over random sequences, particularly rare events, are integral parts of such tests. I extended an existing work where large deviations from the asymptotically predicted Gumbel distribution were found to further biologically relevant scoring and protein models. In most cases, deviations could be described by an heuristically modified Gumbel distribution. In some cases the deviations are so large that, in the previous praxis, some significant similarities are not properly classified, in particular when one wishes to refine a certain search result. First, I studied the score distribution for the standard protein model for different alignment parameters. In a second step, I investigated a model which describes transmembrane proteins.

Furthermore I studied free-energy distributions of canonical alignment ensembles.

I explained the temperature dependence of the shapes of these distributions with arguments of the linear-logarithmic phase transition that occurs in this model.

In a similar way, I studied RNA secondary structures. I obtained the minimum-free-energy distribution over random sequences. This distribution allowed me to compare biological RNA sequences against random models. For this purpose, I considered microcanonical sequence ensembles and compared their statistical properties to those of biological RNA sequences taken from a database.

I also used RNA secondary structures for a study of Monte Carlo dynamics in complex energy landscapes. This model is an ideal system for such purposes, because, in contrast to many other models, it can be treated exactly and, on the other side, it exhibits complex glassy properties. I compared dynamical properties of different Monte-Carlo algorithms to static properties which can be probed with transfer matrix calculations.

Chapter 1

Introduction

Computational molecular biology, or bioinformatics [CB05, RDM98], has arisen from different scientific fields, like molecular biology, computer science, probability theory and statistics. Since recently also many physicists have studied problems from bioinformatics and figured out that many bioinformatics models have an equivalent or a similar description in physics and vice versa. Just as the recent tendency of increasing interchange between computer science and physics [HR01, HW05], this point of view allows one to interchange methods and concepts or even results between bioinformatics and physics [LV02].

Analysis and classification of molecular biological data are challenging tasks in bioinformatics. Most data is stored in form of biological sequences in large databases (for example UniProt [Uni]). A biological sequence, also called primary structure, is the linearly ordered chain of monomers of a biopolymer, such as *deoxyribonucleic acid* (*DNA*), *ribosomal nucleic acid* (*RNA*) or *proteins*. They are usually encoded as character strings over finite alphabets, four letter alphabets in the case of DNA or RNA or the 20 letter amino acid alphabet for protein sequences.

It is commonly assumed that related organisms, so called *homologs*, share similarities on the molecular level. For this reason sequence comparison is a fundamental tool to detect homological relationships. Common search tools, like BLAST (*Basic Local Alignment Search Tool*) [BLA], are used to search a given query sequence against huge databases. In most cases search algorithms are based on *local pairwise sequence alignment* [CB05, RDM98]. It quantitatively measures similarities between a pair of sequences and detects corresponding regions in both sequences. The approach uses the *dynamic programing* paradigm [CLR02] (commonly known as *transfer matrix calculations* in physical literature). The algorithms return a *raw similarity score* that quantifies the similarity between the input objects. A more detailed introduction to sequence alignment is given in Chapter 3.

Unfortunately, the raw score is hard to interpret because one does not know the absolute scale of the score. An interpretation becomes possible when specifying a probabilistic null model for the input: Then the similarity score becomes a random variable S whose probabilities $\mathrm{Prob}(S = s)$ under the null model can be determined. Sometimes this can be done analytically [KA90, KD92, DKZ94], but usually one has to apply numerical simulations [AG96, ABOH01, RO99, ABOH01]. The *p-value* assigned to an observed score s is defined as $pval(s) := \mathrm{Prob}(S \geq s)$ in the null model and $-\log pval(s)$ is a measure of surprise (and hence a universally normalized score) for s. It is one fundamental problem in bioinformatics to find $\mathrm{Prob}(S)$ for a given

comparison method, a given scoring scheme, and a given null model.

Since true homological relationships usually exhibit large scores, the rare-event tail of the score distribution is particularly interesting. Rare-event tails are usually hardly accessible with naive "simple sampling" methods. Similar problems can be found in statistical mechanics, where one is interested in tails of ground-state-energy distributions of disordered systems with quenched disorder (for example [Pal03, ABM04, MG06, KKH06]). These are models with random interactions, where each realization of random interactions induces a physical ensemble on its own.

A fruitful solution to the problem of probing the rare-event tail of such distributions is to reinterpret the ensemble of realizations as a physical ensemble and make use of methods to compute the microcanonical entropy function, i.e. the logarithm of the density of states (the number of micro states for a given energy). Because feasible exact methods are not available in most cases, such problems are approached by Monte Carlo simulations, such as parallel tempering combined with reweighting techniques or generalized ensemble methods.

A few years ago Hartmann applied such a method to the alignment problem [Har02, HR04] and figured out that the accurate score distribution strongly differs from the analytically predicted score distribution in the rare-event tail. Unfortunately these results have not been considered in current database search tools, presumably because it was applied for only one case so far.

It is one aim of this thesis to extend these results to a broader range of scoring and protein models. Under the standard protein-sequence model, the effects of varying scoring parameters was studied. In a second step, the score statistics for a special class of proteins that are hardly described by the standard model was considered. Finally, the statistics of a finite-temperature version of the local alignment algorithm [Miy95, KL00] was investigated. The Monte Carlo algorithms that were used here are introduced in Chapter 2. The results for the local-alignment-score statistics are discussed in Chapter 4.

Another important problem in bioinformatics, molecular biology and biophysics is the prediction of the spatial conformation, the *tertiary structure*, of molecules from primary sequences. In contrast to the tertiary structure, a *secondary structure* describes the conformation on a topological level, i.e. the set of paired monomers.

Such higher order structures are important because they determine the molecule's function. The protein folding problem (the prediction of the three dimensional structure from the amino acid sequence) is probably the most prominent example. Beside protein structures, also RNA structures play an important role in living organisms. In order to fulfill a certain biological function, the molecule's structure is assumed to sit in a global minimum of the free energy in the structure space for a fixed sequence. Hence, many RNA structure prediction methods are based on free energy minimization. Fortunately, RNA structure prediction turned out to be much simpler than protein folding, because *secondary structures* without so called *pseudo knots* (topologically crossing pairs) describe the essential features already quite well [TB99]. Neglecting pseudo knots allows us to perform free-energy minimization in polynomial time by dynamic programing (transfer matrix calculations) [dG68, NPGK78, ZS81, Zuk89, HFS$^+$94, MSZT99]. Such algorithms are explained in Chapter 5.

Because biological RNA sequences are products of evolutionary processes they can hardly be seen as purely random objects. They rather sit in a local minimum of the map from the space of sequences to minimum free-energy [FHS99, CFKK05]. Statistical evidence of the non-randomness of a given sequence is often measured by the so called z-score. That is the distance of the minimum-free-energy value from the

mean of a distribution over a random sequence ensemble normalized by the standard deviation [SD99, WK99, CFKK05]. For this reason the free-energy distribution over random sequences is of interest. Considering the minimum-free-energy structure as the ground state, the problem is again equivalent to the ground-state-energy distribution over sequence ensembles and hence rare-event simulation methods can be applied.

In Chapter 6, I present some results on this distribution, in particular on properties of rare events in both tails, the one for unstable (large free energies) and the one for stable (low free energies) molecules. For this purpose I employed the same methodology as for the local-alignment-score distribution. This allowed me to compare properties of microcanonical-like ensembles of sequences, characterized by the minimum free-energy, to biological RNA sequences.

Interestingly, a simplified model of the RNA secondary structure is also of fundamental interest in statistical mechanics of systems with quenched disorder. When lowering the temperature the model exhibits a phase transition from a molten phase to a spin-glass-like phase [Hig96, PPRT00, Har01, BH02b, FKM02, LW06]. The latter one is characterized by rugged free-energy landscapes, where thermodynamic and ground-state properties exhibit large sample-to-sample fluctuations. In contrast to most other models featuring complex free-energy landscapes, probing static properties of RNA secondary structures is computationally feasible. One the other side, due to the glassiness of the model an interesting slow dynamics in the structure space can be expected. Such properties are crucial when performing Monte Carlo or molecular dynamics simulations. Hence, the model provides an ideal framework to study the relationship between static and dynamic properties.

Monte Carlo studies of this type are presented in Chapter 7. Static structure properties, such as the number of metastable states or the degree of ultrametricity, of random sequences were determined. The relationship of these properties to observations of different Monte Carlo methods, like the tunneling time in generalized ensembles, or sampling errors is worked out in detail. Finally, an improved sampling scheme that allows the Monte Carlo samplers to cross entropic barriers is presented.

Chapter 2

Monte Carlo methods

The general concept of the "Monte Carlo method" was proposed by Von Neumann, Metropolis, Ulam and others in the 1940s [MU49] and since then it was successfully enhanced and used in many different scientific fields [HH64, MEJN99, Liu02, LB05]. The idea is to estimate expectation values of observables over a state space by generating random states by means of computer simulations. The randomness of the procedure also gives the approach its name.

2.1 Simple sampling

In order to be more concrete, given a discrete state space χ, and an observable $A : \chi \rightarrow \mathbb{R}$ one wishes to estimate expectation values with respect to a probability mass function $p : \chi \rightarrow [0, 1]$. The idea of "Monte Carlo" is to estimate $\langle A(X) \rangle_p$ by drawing random samples $x_1, \ldots, x_n \in \chi$ according to p and estimate the expectation value from the sample average

$$\langle A(X) \rangle_p = \sum_{x \in \chi} A(x) \cdot p(x) \approx \frac{1}{n} \sum_{i=1}^{n} A(x_i) \tag{2.1}$$

Nearly all Monte Carlo algorithms can by classified as a *rejection-free* or as a *reject-accept method*. Prototypes of these approaches are the *inversion method* or the *reject-accept algorithm* [Dev86] respectively. The requirements of the inversion method are quite restrictive as it relies on the knowledge of the inverse of the cumulative distribution function F^{-1}. This requires also a kind of ordering of all states to be sampled. For each variate the algorithm uses an uniform variate $\xi \in [0, 1]$ and then returns $F^{-1}(\xi)$ [Dev86]. Note that in some cases it is possible to order the states hierarchically which allows direct sampling with an inversion-like approach. This idea is realized in the Boltzmann sampling of RNA secondary structures and finite-temperature alignments (see Appendix A.1 and Appendix A.2).

In many cases this method is infeasible because the underlying state space is an high dimensional object and the inversion of F requires much information about the system. The reject-accept algorithm is more flexible. Suppose we wish to sample from the distribution p, and we were able to compute $p(x)$ up to a global normalization constant (partition function). In order to draw one sample with the reject-accept method, one repetitively draws $\xi \in [0, 1]$ and states x from χ according to an arbitrary distribution q until $\xi c \frac{q(x)}{p(x)} \leq 1$, where $c \in \mathbb{R}$ is a free parameter. In order to apply the reject-accept

```
procedure metropolis_update(x,w)
begin
   propose y ∈ N(x)
   if w(E(y)) > w(E(x)) or w(E(y))/w(E(x)) > rand() then
      x ← y
    end
   return x
end

procedure metropolis(x^init,w)
begin
   x ← x^init
   repeat
      x ← metropolis_update(x, w)
   until x is in equilibrium

   for i = 1 ... n do
      x0 ← x
      repeat
         x ← metropolis_update(x,w)
      until x and x0 are decorrelated
      x^sample[i] ← x
   done
   return x^sample
end
```

Algorithm 2.2.1: The Metropolis algorithm. Only equilibrated and decorrelated states should be sampled (see Sec. 2.5)

algorithm efficiently, the distribution q and the number c should be chosen, such that the ratio $c\frac{q(x)}{p(x)}$ is large.

Furthermore, Monte Carlo methods can be classified whether the generated outcomes are *correlated* or *uncorrelated*. Uncorrelated means that the Monte Carlo procedure returns random objects that are not correlated to the outcomes of previous calls. This will be denoted as "simple sampling".

Because simple sampling is hardly possible in many application, indirect ways such as Markov chain Monte Carlo (MCMC) methods, in particular the Metropolis-Hastings algorithm, had become very popular and successful. This algorithm is explained next.

2.2 The Metropolis-Hastings algorithm

The Metropolis-Hastings algorithm [MRR$^+$53, Has70] is a general reject-accept Monte Carlo method that is suitable for problems, where simple (uncorrelated) sampling is hardly possible. It is based on a discrete time Markov chain in the state space χ with transition matrix $P_{x,y}$. The transition matrix $P_{x,y}$ is constructed such that its stationary distribution converges to the desired sampling distribution p, i.e. $\sum_{x\in\chi} p(x) \cdot P_{x,y} = p(y)$ for all states $y \in \chi$.

The algorithm (see Algorithm 2.2.1) starts with some inital state $x^{\text{init}} \in \chi$. At each

time step of the simulation a new state in the local neighborhood of the current state $y \in \mathcal{N}(x)$ is proposed with the "proposal density" $Q_{x,y} = \mathrm{Prob}(X = y|X = x)$. The proposal is then accepted (y is used at the next time step) with the probability

$$\alpha_{x,y} = \min\left\{1, \frac{p(y) \cdot Q_{y,x}}{p(x) \cdot Q_{x,y}}\right\}. \tag{2.2}$$

Otherwise the proposal is rejected and x is kept for the next iteration. The rate $\alpha_{x,y}$ in Eq. (2.2) is constructed such that detailed balance,

$$\alpha_{x,y} Q_{x,y} p(x) = \alpha_{y,x} Q_{y,x} p(y),$$

is guaranteed. Furthermore the chain needs to be ergodic. This means each state must be reachable from any other state by a finite chain of transitions. If these conditions are fulfilled the chain converges towards the stationary distribution p and one may approximate expectation values by averages over the visited states.

In the case of symmetric proposals $Q_{x,y} = Q_{y,x}$ the Metropolis-Hastings algorithm in its general form Eq. (2.2) [Has70] simplyfies to the Metropolis algorithm with acceptance rate

$$\alpha_{x,y} = \min\left\{1, \frac{p(y)}{p(x)}\right\}, \tag{2.3}$$

which was proposed by Metropolis et. al. in their famous article from 1953 [MRR$^+$53].

Usually the sampling distribution depends on a macroscopic observable, which is the energy in most physical applications. We shall write $p(x) \propto w(E(x))$, and hence

$$\alpha_{x,y} = \min\left\{1, \frac{w(E(y))}{w(E(x))}\right\}. \tag{2.4}$$

In order to implement this algorithm, the weights w need to be known up to normalization. In the canonical ensemble,

$$w(E) \propto e^{-E/T},$$

the ratio in Eq. (2.4) only depends on the change of the energy, that is associated with the proposal,

$$\alpha_{x,y} = \min\left\{1, \exp\left[-\Delta E/T\right]\right\}. \tag{2.5}$$

2.3 The dynamics of the N-fold way

In a few discrete models, such as Ising spin systems or the RNA secondary structure, it is possible to use a rejection-free dynamics and determine the correct averages by accounting for so called *waiting times* at each step of the simulations. This is referred as *N-fold way*. So as to follow the classification scheme that has been described above, the N-fold way is a rejection-free Monte Carlo method, that produces correlated states. It requires following assumptions:

1. there must be a small number of possible energy changes $\{\Delta E_i\}$, $i = 1 \ldots N$ and

2. at each step there must be an efficient way to classify all possible proposals $y \in \mathcal{N}(x)$ by the associated energy change ΔE. These sets are denoted as *classes of proposals*

$$\mathcal{C}(x, \Delta E) = \{y \in \mathcal{N}(x)|E(y) = E(x) + \Delta E\}$$

Each step in the N-fold way requires two random selections. First a class is chosen with a probability that corresponds to the Metropolis algorithm and secondly one member from that class is selected with equal probability. This proposal is then used in the next iteration with probability one. There are two timescales involved, the *computer time* measuring the number of MC steps and the *MC time* associated with a physical time scale of the random walker. The first plays the major role in the performance analysis of the algorithms and the latter one gives the correct weight to the visited states. In order to choose the class with the "Metropolis weight" (i.e. the probability of choosing $y \in \mathcal{N}(x)$ times the acceptance rate in Eq. (2.4)), the cumulative weights

$$Q(k) = \min\left\{1, \frac{w(E+\Delta E_i)}{w(E)}\right\} |\mathcal{C}(x, \Delta E_k)| + Q(k-1)$$

with the boundary condition $Q(0) = 0$ have to be determined for $k = 1 \ldots N$. Consequently $Q(k)/Q(N)$ is the probability that a flip occurs in the k first classes. The inversion method allows choosing a class k with the same probability as it would have been chosen with the Metropolis algorithm. Within that class a proposal $y \in \mathcal{C}(x, \Delta E)$ is chosen at random and accepted in any case. When computing expectation values from the chain of visited states $x_i \ldots x_n$ in the form of Eq. (2.1), the time that the random walker would have stayed in each state x_i before an acceptance from x_i to x_{i+1} (the *waiting time*) has to be taken into account.

It can be computed by the following considerations: Let p be the probability that a proposal in the first $k-1$ classes is selected, given that the random walker sits in state x, i.e. $p = Q(k-1)/Q(N)$. Then the probability that the Metropolis algorithm had selected class k after m trials is given by

$$p(m) = p^m(1-p). \tag{2.6}$$

The probability of staying at most τ time steps can be evaluated via geometric progression:

$$P(\tau) = \mathrm{Prob}\,[m \le \tau] = \sum_{m=0}^{\tau} p^m \cdot (1-p) = 1 - p^{\tau+1} \tag{2.7}$$

In order to assign a waiting time to the current state one has to draw a random number according to the discrete distribution Eq. (2.7), i.e.

$$\tau = \lfloor \ln(\zeta)/\ln(p) \rfloor,$$

where ζ is an uniformly distributed random number and $\lfloor x \rfloor$ denotes rounding down to the next integer. This completes one MC step. Expectation values are then approximated by

$$\langle A \rangle \approx \frac{1}{n + \sum_i \tau_i} \sum_{i=1}^{n} (\tau_i + 1) A(x_i)$$

In Sec. 7 a variant of the N-fold way is discussed. This will be refereed as "semi-rejection free".

2.4 Parallel tempering

Metropolis Coupled Markov Chain Monte Carlo (MCMCMC) was first invented by Charles Geyer [Gey91] and then reinvented by Hukushima and Nemoto [HN96] under

the term *exchange Monte Carlo*. In the physical literature MCMCMC is often denoted as *parallel tempering*. The method has become a standard tool in disordered systems that feature a rough (free) energy landscape [ED05]. These rough energy landscapes are characterized by high energy barriers and can be found for problems like protein folding [ZBG01], nucleation [AF01], spin-glasses [MPRL98, KPY01] and other models characterized by rare events [Har01, KKH06]. In the last decade it turned out that MCMCMC accelerates equilibration and mixing remarkably.

In the framework of MCMCMC m copies $\chi^{(1)}, \ldots, \chi^{(m)}$ of a system are simulated in parallel. The sampling distributions $p_{\Theta_1} \ldots p_{\Theta_m}$ belong to a single-parameter family of distributions. This means one samples from the product of the state space χ^m weighted with the joint distribution with weights $\prod_{j=1}^{m} p_{\Theta_j}$.

In most applications where one wishes to sample from a Gibbs-Boltzmann distribution, the parameter can be identified with the temperature, i.e. $\Theta_j = T_j$ and hence the sampling distribution is a product of m canonical ensembles. Without loss of generality we will denote the parameter as "temperature" and assume $T_1 < \ldots < T_m$ in the following. The parallel tempering algorithm is designed to exchange configurations between different neighboring temperatures during the simulation. For this purpose let us define the space of all possible mappings from the m configuration to the m temperatures as *temperature space*.

During the simulation mainly each of the replicated configurations will evolve independently according to the underlying MCMC scheme characterized by the Boltzmann weight $\exp(-\frac{1}{T_j}E(x))$ at its current temperature T_j, i.e. according to Eq. (2.5).

In addition to this evolution, every t_{exc}*th* step (for each replicated configuration) a flip between two neighboring replicas k and $k+1$ ($k \in \{1, \ldots, m-1\}$) is attempted. If an attempt is successful, the configurations $x^{(k)}$ and $x^{(k+1)}$ are exchanged (denoted by $x^{(k)} \leftrightarrow x^{(k+1)}$), i.e. the configuration which has previously evolved at temperature T_k will now evolve at temperature T_{k+1} and vice versa. This exchange is accepted with the probability

$$\alpha\left(x^{(k)} \leftrightarrow x^{(k+1)}\right) = \min\left\{1, \frac{p_{T_k}(x^{(k+1)})}{p_{T_k}(x^{(k)})} \cdot \frac{p_{T_{k+1}}(x^{(k)})}{p_{T_{k+1}}(x^{(k+1)})}\right\}. \tag{2.8}$$

In the canonical ensemble this ratio depends on the difference of the inverse temperature $\Delta\beta_k = \frac{1}{T_{k+1}} - \frac{1}{T_k}$ and on the energy difference $\Delta E = E(x^{(k+1)}) - E(x^k)$,

$$\alpha\left(x^{(k)} \leftrightarrow x^{(k+1)}\right) = \min\left\{1, \exp\left[\Delta\beta_k \Delta E\right]\right\}, \tag{2.9}$$

This leads to a "random walk in the temperature space".

The parallel tempering approach has the advantage over the standard Metropolis algorithm that the different configurations are not confined to a fixed temperature, but perform a random walk in temperature space, i.e. visit all temperatures several times. Thus, mixing is accelerated and hence fewer Monte Carlo steps are required.

It is suitable for at least three purposes:

- optimization, i.e. finding low energy states in rugged energy-landscapes,
- approximating canonical averages over the ensemble with the lowest temperature T_1 and
- determine canonical averages for *any* temperature within the interval $[T_1, T_m]$.

For the first two applications the high temperatures ensembles are only auxiliary for the sake of decreasing equilibration time. For the latter application, data from all chains are relevant for the data analysis. In particular, when choosing $T_m = \infty$ together with a broad range of temperatures $T_1, \ldots, T_{m-1}$, the density of states (DOS) can be determined. The methodology of reweighting such mixtures of empirical distributions is described in detail in Sec. 2.6.

2.5 Convergence

Due to the fact that the Metropolis-Hastings algorithm generates *correlated* states, some care has to be taken when computing averages in the form of Eq. (2.1) for two reasons. Firstly, because the inital configuration might be far away from the equilibrium of the sampling distribution p. For this reason usually the first steps of the chain (called *burnin* or *equilibration* time) have to be ignored in the estimators. Secondly the generated states are correlated, which is crucial for the estimation of the statistical error. To avoid this the chain is usually thinned out, i.e. only every n_{thin}*th* visited state is considered for data production (see Ref. [CC96] for an review).

2.5.1 Equilibration

The estimation of the equilibration time is not always trivial and depends strongly on the model.

A visual way, which has been proven to be appropriate to our purpose (see Sec. 4.2), is to compare the convergence of two chains starting from two different initial configurations. For instance, if we consider to simulate a physical system in the canonical ensemble at temperature T and we were able to generate low-temperature configurations (for example ground states), then it is possible to choose two distinct starting configurations. When starting from a random configuration (for example $T = \infty$, i.e. a disordered configuration) and equilibrating the system, the system can reach in principle most regions of the energy landscape at the beginning. Hence, typically the energy decreases or stays the same during the simulation with only few energy fluctuations. In contrast, when starting from a low energy, i.e. a minimum of the energy landscape, the reverse process is possible. One can use this fact to verify, whether a system has equilibrated, i.e. whether it is able to overcome the typical barriers in the energy landscape. This is the case when the average energy for two runs, one starting with a disordered configuration and one starting with a "ground-state" configuration, have converged to the same value (within fluctuations). If the temperature is too small, this is not possible in many glassy systems. An example of this approach is shown in Fig. 4.3.

2.5.2 Relaxation

So as to estimate the thinning interval many different approaches are available [MEJN99, CC96, Jan02]. To estimate the times scales over which the simulation decorrelates, we considered the autocorrelation function

$$\xi(t) = \frac{\langle E(t_0)\, E(t_0+t)\rangle_{t_0} - \langle E(t_0)\rangle_{t_0}^2}{\langle E(t_0)^2\rangle_{t_0} - \langle E(t_0)\rangle_{t_0}^2}\,, \tag{2.10}$$

$\langle \cdots \rangle_{t_0}$ denoting the average over different times and independent runs. The typical time scale, over which correlation vanish is the correlation time τ defined via $\xi(\tau) =$

$1/e$. The correlation time increases with decreasing temperature, which corresponds to a growth of the equilibration time with decreasing temperature. However, by the generation of the histograms the correlations will average out, but estimates of the errors are more complicated when the data is correlated. A common rule of thumb is to choose $n_{\text{thin}} \approx 2\tau$ as thinning interval.

Another method that aims at a direct computation of the statistical error is Flyvbjerg and Petersen's blocking method [FP89], which has two main advantages. Firstly it is computationally less complex than the computation of the autocorrelation function, and secondly a generalization to multidimensional observables is straightforward. For a given Monte Carlo data set of correlated data of some observable A, $a_1, \ldots a_n$, the sample error $\widehat{\epsilon}(a_1, \ldots, a_n)$ is a lower bound of the "true" error of the uncorrelated data. The blocking method uses a series of blocking transformation, where the actual dataset $a_i^{(k)}$ ($i = 1 \ldots n^{(k)}$) is transformed into a coarse grained set according to the rule

$$\begin{aligned} a_i^{(k+1)} &= \frac{1}{2}\left(a_{2i-1} + a_{2i}\right) \quad (i = 1 \ldots n^{(k)}/2) \\ n^{(k+1)} &= \frac{n^{(k)}}{2} \end{aligned}$$

This transformation keeps the expectation value $\langle A \rangle$ and the true statistical error $\epsilon(A)$ invariant. The transformation is repeated until $n^{k_{\max}} = 2$. When the block size is at least as large as the intrinsic correlation time the empirical error of the blocked data approaches a plateau (within the statistical error) of constant values for $k \geq k_0$ for some k_0. This fixed point serves as an consistent estimate of the statistical error of the correlated data.

2.6 Sampling of rare events I: Importance sampling and reweighting

Suppose that we wish to estimate the probability distribution of an observable A, i.e. $P(a) = \text{Prob}(A(X) = a)$. Since distributions of this form can always be formulated as expectation values of indicator functions, it is possible to estimate probabilities via Monte Carlo sampling,

$$P(a) = \langle \delta_{A(X),a} \rangle_p \approx \frac{1}{n} \sum_{i=1}^{n} \delta_{A(x_i),a}.$$

This simple sampling approach allows probing the region of the distribution where $P(a)$ is large. If the probability to be estimated is small, say $\approx 10^{-9}$, we need about 10^{12} samples to estimate it with reasonable precision. For very rare events, this "naive" sampling quickly becomes infeasible.

Importance sampling generates the "interesting" events more often by sampling from a different distribution and correcting for this bias afterwards, which results in a more accurate estimate with a reasonable number of samples. Let p be the "target distribution" and q be an alternative distribution over χ, the so called "sampling distribution". Consequently samples from q, in the following denoted as $x'_1, \ldots, x'_n$, allow for estimates of the expectation value of an observable A with respect to the target

distribution using the importance sampling formula

$$\begin{aligned} \langle A(X)\rangle_p &= \sum_x A(x)\cdot p(x) \\ &= \sum_x A(x)\cdot\frac{p(x)}{q(x)}\cdot q(x) \\ &\approx \frac{1}{n}\sum_{i=1}^{n} A(x_i')\cdot\frac{p(x_i')}{q(x_i')}, \end{aligned} \tag{2.11}$$

or, to estimate the probability $P(a)$,

$$P(a) = \left\langle \delta_{A(X),a}\right\rangle_p \approx \frac{1}{n}\sum_{i=1}^{n}\delta_{A(x_i'),a}\cdot\frac{p(x_i')}{q(x_i')}. \tag{2.12}$$

To successfully apply importance sampling, q has to fulfill three properties:

- it needs to put high probability on the region of interest,
- we need to be able to sample according to q and
- we need to be able to compute the correcting weight $p(x)/q(x)$.

For the estimator of rare-event probabilities it is not sufficient to put high probability on the rare event alone, because then the distribution's normalization remains undetermined. Instead the entire range from high probabilities (where $P(a)$ is large) down to rare events have to be sampled. Since this range is very broad it is hard to find a good guess for the sampling distribution q a priori.

Torrie and Valleau developed a technique called "umbrella sampling" [TV77], which was originally used to estimate free-energy differences. The method makes use of a parameterized family of sampling distributions $\{q_{\Theta_k}\}$, $k = 1,\ldots,m$ and requires that the mixture covers the entire range of interest. That can be for example the mixture of canonical distributions ($\Theta_k \equiv T_k$ and $q_{T_k}(x) \propto \exp\left[-\frac{1}{T_k}E(x)\right]$), that is involved in the parallel tempering algorithm.

Furthermore, we consider that the target distribution p is one member of the family, without loss of generality we set $p = q_{\Theta_m}$ and we shall write $q_k \equiv q_{\Theta_k}$. Ferrenberg and Swendsen [FS89] proposed a data analysis procedure, which can be seen as a generalization of Eq. (2.12) for a mixture of distributions.

Later, Geyer developed a related method under the term *reverse logistic regression* [Gey91]. Meng and Wong [MW96] reviewed the basic concept in the framework of Bayesian inference and proposed different recipes to obtain the relative normalization constants (partition functions) which play a central role in this methodology. We use the selfconsistent method proposed in [MW96] to derive the normalization constants from a mixture of Monte Carlo data.

Consider the family $\{q_k\}$ covering the region of interest and n_k independent[1] Monte Carlo samples $\{x_{ki}, k = 1\ldots m, i = 1\ldots n_k\}$ from each distribution q_k. Furthermore we assume that each q_k is only known up to (global) normalization constants $c_k = \sum_x \tilde{q}_k(x)$, i.e.

$$q_k(x) = \frac{\tilde{q}_k(x)}{c_k} \tag{2.13}$$

[1] for correlated data one has to account for autocorrelation times and consider a thinned sample as described in Sec. 2.5

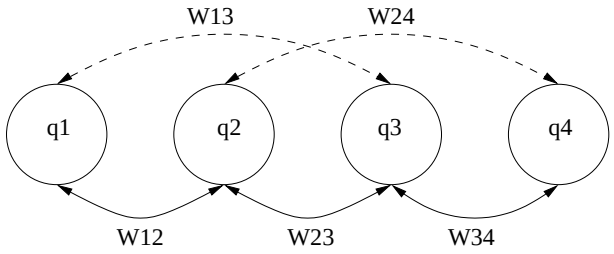


Figure 2.1: Sketch of the graph of overlapping distributions $q_1, ..., q_4$. Distant distributions have weak overlaps.

Since the support of the mixture distribution is broader than each of the particular distributions, not all pairs of distributions q_k and q_l overlap in general. The overlaps of the empirical data can be measured by the matrix

$$w_{kl} \approx \frac{1}{\frac{1}{2}(n_k + n_l)} \sum_{x \in \chi} \sum_{i=1}^{n_k} \sum_{j=1}^{n_l} \delta_{x,x_{ki}} \cdot \delta_{x,x_{lj}} \tag{2.14}$$

and the set of distributions can be represented by a graph (V, E) with vertices being the distributions $V = \{q_1, ..., q_m\}$ and the set of all overlaps being the weighted edges $E = \{w_{kl}\}$ with $w_{kl} > 0$ (see Fig. 2.1). We require, that the so constructed graph is connected.[2] This criterion was used in the study of the local alignment statistics discussed in Sec. 4.3

Geyer's idea is to generalize Eq. (2.11) to mixtures by "forgetting" from which distribution each sample was drawn and assume that it was drawn from the mixture. This is done by replacing each q by a "mixture weight" q_{mix},

$$\langle A \rangle_p \approx \frac{1}{n} \sum_{k=1}^{m} \sum_{i=1}^{n_k} A(x_{ki}) \cdot \frac{p(x_{ki})}{q_{\text{mix}}(x_{ki})}, \tag{2.15}$$

with $n = \sum_{j=1}^{m} n_j$ The mixture weight q_{mix}

$$q_{\text{mix}}(x) = \sum_{k=1}^{m} \frac{n_k}{n} \cdot \frac{\tilde{q}_k(x)}{c_k}$$

is known up to the normalization constants c_k that have to be determined consistently from *all* Monte Carlo data. This is possible up to a global (trivial) constant by considering ratios of these constants

$$\frac{c_k}{c_l} = \sum_{x \in \chi} \frac{\tilde{q}_k(x)}{c_l} = \sum_{x \in \chi} \frac{\tilde{q}_k(x)}{\tilde{q}_l(x)} \cdot q_l(x) = \left\langle \frac{\tilde{q}_k(X)}{\tilde{q}_l(X)} \right\rangle_l$$

(because $c_l = \frac{\tilde{q}_l(x)}{q_l(x)} \ \forall x$, see Eq. (2.13)). Each pair of ratios c_k/c_l are in principle accessible from Monte Carlo data, if the distributions are not too distant. In general, each distribution of interest should have a finite overlap with q_{mix} which ensures that reweighting becomes possible on the full support.

[2] In practice one must find paths between each pair of distributions with not too small weights.

There might be incompatible (due to finite sample sizes) estimates of c_k/c_l that are computed directly or indirectly $\frac{c_k}{c_l} = \frac{c_k}{c_{l'}} \cdot \frac{c_{l'}}{c_l}$ using the "bridge distribution" $q_{l'}$. One wish to estimate these ratios consistently using *all* data that is available from the mixture, i.e. using $\frac{c_k}{c_l}$, $\frac{c_k}{c_{l'}}$ and $\frac{c_{l'}}{c_l}$ in the case of three distributions. A method to obtain self-consistent estimates of these normalization constants is discussed in the following.

2.6.1 Estimation of relative normalization constants

Geyer introduced a *profile log-likelihood*, which depends on the unknown normalization constants given the Monte Carlo data. Since there is a global trivial normalization constant the estimation reduces to $(m-1)$ ratios of normalization constants with respect to an arbitrary reference distribution, say q_1. Hence the aim is to estimate the vector $\mathbf{r} = (r_2, \ldots, r_m) \in \mathbb{R}^{m-1}$ with $r_k = c_1/c_k$ from the Monte Carlo mixture. Additionally $r_1 = 1$ by definition and we do not consider it as a free parameter in the following. Using the probability that x has been sampled from the k*th* distribution $p_k(x, \mathbf{r}) = \frac{\tilde{q}_k(x) \cdot r_k}{\sum_l \tilde{q}_l(x) \cdot r_l}$ one can construct a log-likelihood (a function of the unknown normalization constants) the for the complete data set

$$\mathcal{L}(\mathbf{r}) = \sum_{k=1}^{m} \sum_{i=1}^{n_k} \log p_k(x_{ki}, \mathbf{r}) \tag{2.16}$$

and obtain the normalization constants by maximazing $\mathcal{L}$ with respect to $\mathbf{r}$, i.e. $\hat{\mathbf{r}} = \text{argmax}_{\mathbf{r}} \mathcal{L}(\mathbf{r})$. In other words, the relative normalization constants are determined by a maximum likelihood estimator.

In practice one may implement the Newton-Raphson method or an iterative procedure. Meng and Wong [MW96] proposed a reliable selfconsistent method to obtain $\mathbf{r}$, which is easy to implement on one side and stable on the other side. This approach is explained in the following.

Let $\alpha_{kl} : \Omega \to \mathbb{R}$ be a set of arbitrary functions with $\alpha_{kl} = \alpha_{lk}$ and

$$0 < \left| \sum_{x \in \Omega} \alpha_{kl}(x) \cdot q_k(x) \cdot q_l(x) \right| < \infty.$$

The average of $\tilde{q}_k(x) \cdot \alpha_{kl}(x)$ equals to

$$\begin{aligned} \langle \tilde{q}_k(x) \cdot \alpha_{kl}(x) \rangle_l &= c_k \sum_{x \in \Omega} q_k(x) \cdot \alpha_{kl}(x) \cdot q_l(x) \\ &= \frac{c_k}{c_l} \sum_{x \in \Omega} q_k(x) \cdot \alpha_{kl}(x) \cdot \tilde{q}_l(x) \\ &= \frac{c_k}{c_l} \langle \tilde{q}_l(x) \cdot \alpha_{kl}(x) \rangle_k \end{aligned}$$

which can also be written in the form

$$\frac{c_k}{c_l} = \frac{\langle \tilde{q}_k(x) \cdot \alpha_{kl}(x) \rangle_l}{\langle \tilde{q}_l(x) \cdot \alpha_{lk}(x) \rangle_k} \tag{2.17}$$

that Meng and Wong called the "key identity". By using Eq. (2.17) in terms of $r_k =$

c_1/c_k and summing over l

$$\begin{aligned}\underbrace{\sum_{l\neq k}\langle\tilde{q}_k(x)\cdot\alpha_{kl}(x)\rangle_l}_{b_{kk}}\cdot r_k &= \sum_{l\neq k}\langle\tilde{q}_l(x)\cdot\alpha_{lk}(x)\rangle_k\cdot r_l \\ &= \sum_{l\neq k,l\neq 1}\underbrace{\langle\tilde{q}_l(x)\cdot\alpha_{lk}(x)\rangle_k}_{b_{kl}}\cdot r_l + \underbrace{\langle\tilde{q}_1(x)\cdot\alpha_{1k}(x)\rangle_k}_{b_{k1}}\end{aligned}$$

it becomes clear that $\mathbf{r}$ satisfies the following linear system

$$\begin{pmatrix} b_{22} & -b_{23} & \dots & -b_{2m} \\ -b_{32} & b_{33} & \dots & -b_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ -b_{m2} & -b_{m3} & \dots & -b_{mm}\end{pmatrix}\begin{pmatrix} r_2 \\ r_3 \\ \vdots \\ r_m\end{pmatrix} = \begin{pmatrix} b_{21} \\ b_{31} \\ \vdots \\ b_{m1}\end{pmatrix} \tag{2.18}$$

$$\text{with } \begin{cases} b_{kk} = \sum_{l\neq k}\langle\tilde{q}_k(x)\cdot\alpha_{kl}(x)\rangle_l & , 2\leq k\leq m \\ b_{kl} = \langle\tilde{q}_l(x)\cdot\alpha_{lk}(x)\rangle_k & , k\neq l\end{cases}$$

The next step is to replace the b_{lk} in Eq. (2.18) by the sample average and choose

$$\alpha_{lk}(x) = \frac{n_l\, n_k}{n^2}\frac{1}{q_{\text{mix}}(x)}.$$

Since q_{mix} and therefore the b_{kl} as well depend on the unknown ratios $\mathbf{r}$ Eq. (2.18) is not a true linear system in $\mathbf{r}$, but it is possible solve the equation self-consistently by iterating the equation

$$\hat{B}(\mathbf{r}^{(t)})\cdot\mathbf{r}^{(t+1)} = \hat{\mathbf{b}}(\mathbf{r}^{(t)}),$$

where $\hat{B}(\mathbf{r}^{(t)})$ and $\hat{\mathbf{b}}(\mathbf{r}^{(t)})$ are the sample estimates of b_{kl} ($k,l = 2\dots m$) and b_{k1} ($k = 2\dots m$) depending on the ratios $\mathbf{r}$ in the t*th* iteration. The solution of $\mathbf{r}$ in the $(t+1)$*th* iteration, $\mathbf{r}^{(t+1)}$ is obtained by solving the above linear system. This approach converges to the maximizer of Eq. (2.16) [MW96].

2.6.2 Illustration: Reweighting probability distributions

The approach to obtain relative normalization constants for probability distributions is essentially the same as for expectation values. This is so, because this is a special case of an expectation value. The reweighting equation Eq. (2.15) for this special "observable" is given by

$$P(a) = \text{Prob}(A=a) \approx \sum_{k=1}^{m}\sum_{i=1}^{n_k}\delta_{A(x_{ki}),a}\frac{p(x_{ki})}{q_{\text{mix}}(x_{ki})}. \tag{2.19}$$

As illustration we consider a mixture of Gaussian deviates with standard deviation $\sigma = 1$ and mean values $\mu = \{-3,-2,-1,0,1,2,3\}$ (see Fig. 2.2) The PMF q in the above statements is substituted by a probability density function (PDF) and $p = q_{\mu=0}$. The x_{ki} are drawn from $\mathcal{N}(\mu_k,1)$, which can be easily generated by a Box-Müller transform [PFTV92]. For each distribution $10,000$ samples were generated (see the upper plot in Fig. 2.2). The data had been reweighted to the distribution $\mathcal{N}(0,1)$ using the iterative scheme described above (Fig. 2.2, lower).

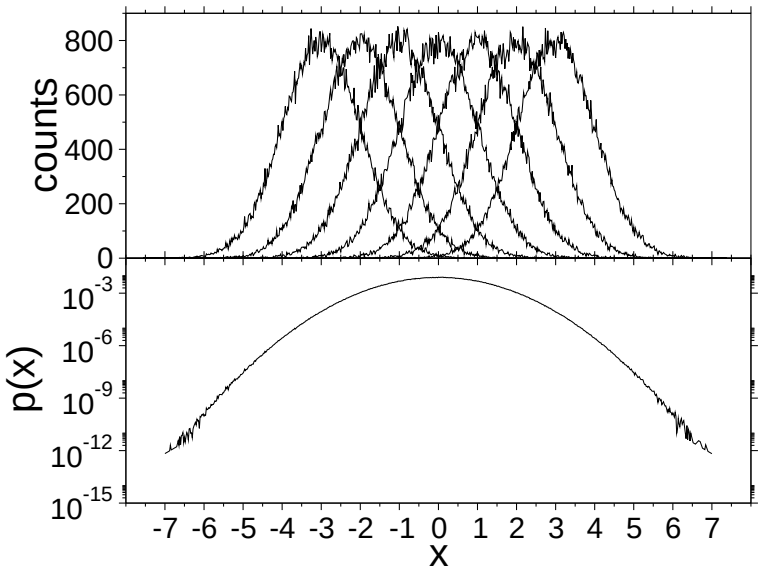


Figure 2.2:
top: Histograms of a Gaussian mixture $\mathcal{N}(\mu_k, 1)$ with different mean values, obtained via Box-Müller sampling. The bin size was 0.02.
bottom: Reweighted data. The distribution of empirical Gaussian deviates is very broad. The tail is accessible via reweighting of a mixture.

In Fig. 2.3 the estimate of q_{mix} as well as the relative error of the empirical estimate $\hat{p}$ with respect to the exact distribution

$$\epsilon(x) := \frac{\hat{p}(x) - p^{\text{exact}}(x)}{p^{\text{exact}}(x)}$$

is illustrated. Obviously the data become noisy beyond $|x| > 3$ where the support is governed by the mixture only weakly. This example is trivial because the normalized distributions are known and there is no need for simulations at all. For this reason it provides a practical test of the reweighting procedure and also an instructive illustration.

2.7 Sampling of rare events II: Generalized ensemble methods

Generalized ensemble methods have become a popular tool in statistical physics since the early 1990s. Similar to parallel tempering their usefulness shows up in problems where distributions over a broad parameter range have to be sampled and a usual Boltzmann sampling gets stuck in local minima of the energy landscape. This usually happens close to critical points, where the correlation time increases with a power law. When sampling from a broad distribution instead of from the narrow Boltzmann distribution, the sampler is allowed to escape from such local minima. A second advantage is that these methods aim at approximating the DOS, and hence obtain the thermo-

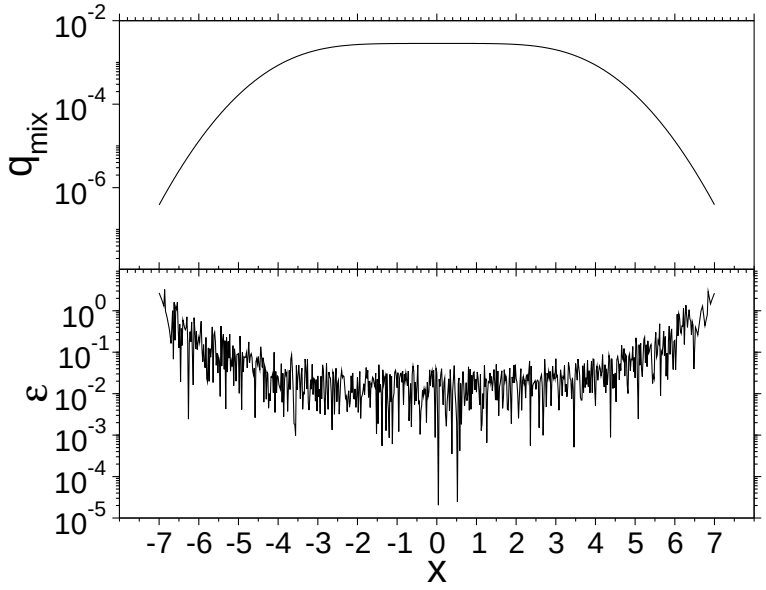


Figure 2.3:
top: $\hat{q_{\text{mix}}}$ estimated from the mixture
bottom: The relative error with respect to the true distribution $\mathcal{N}(0,1)$

dynamics at any temperature, similar as reweighting of mixtures. In practice the data analysis turns out to be simpler. There is no need to determine relative normalization constants, because a single chain or multiple independent chains cover the entire range. Whereas the parallel tempering algorithm requires various parameters, that have to be chosen in advance (the number of temperatures and their values), generalized ensemble methods, especially the Wang-Landau algorithm (Sec. 2.7.1), essentially require an energy interval $[E_{\min}, E_{\max}]$ as input. Hence only little a priori knowledge is needed.

The generalized ensemble method consists of two stages. First, an heuristic algorithm, like the multicanonical iteration [BN92], entropic sampling [Lee93], Wang-Landau sampling [WL01], transition matrix Monte Carlo [Wan99b, WTS99, WL00] or the ParQ [HH05, AHM$^+$88] algorithm, approximates the DOS to a sufficient degree of accuracy. Sec. 2.7.1 and Sec. 2.8.2 explain more details on the Wang-Landau and the ParQ algorithm. In the second stage the Metropolis algorithm produces the relevant data. The weights are chosen as $w(E) \equiv w^{\text{flat}}(E) := 1/\hat{g}(E)$, where $\hat{g}(E)$ denotes the approximated DOS from the first stage. Instead of the energy, the DOS might depend on other observables, for example order parameters. This is referred as "joint density of states". An implementation of the generalized ensemble method for such two-dimensional objects is directly possible. In contrast, it is complicated for the parallel tempering algorithm.

There is still the freedom to choose a global normalization constant. Frequently the DOS is normalized to 1 in order to interpret it as the energy-distribution in the canonical ensemble at infinite temperature. Alternative, in the case that the degeneracy of at least one level is known, it is possible to determine the degeneracy of each energy level. Then $\ln g(E)$ equals the microcanonical entropy function.

procedure wang_landau(x^{init},$E_{\min}$,$E_{\max}$,ϕ^{init})
begin
  $x \leftarrow x^{\text{init}}$
  $\phi \leftarrow \phi^{\text{init}}$
  for $i = E_{\min} \ldots E_{\max}$ **do**
    $w[i] \leftarrow 1$
    $h[i] \leftarrow 0$
  done
  repeat
    repeat
      $x \leftarrow$ metropolis_update(x,w)
      $w[E(x)] \leftarrow w[E(x)]/\phi$
      $h[E(x)] \leftarrow h[E(x)] + 1$
    until h is flat
    $\phi \leftarrow \sqrt{\phi}$
    for $i = E_{\min} \ldots E_{\max}$ **do**
      $h[i] \leftarrow 0$
    done
  until $\phi \leq \phi^{\text{final}}$
  return w
end

Algorithm 2.7.1: The Wang-Landau algorithm.

2.7.1 Wang-Landau sampling

The algorithm of Wang and Landau [WL01] provides an efficient and easy to implement way to estimate the weights for the flat histogram ensemble. Provided that the specific energy function is available, the algorithm can be implemented in a few lines of programming code (see Algorithm 2.7.1).

Instead of sampling from a distribution with fixed weights w, the weights are updated dynamically such that the random walker is biased towards states that have been sampled rarely so far. It employs an histogram $h(E)$ and the weights $w(E)$. After each step the histogram entry of the current state is incremented by one and the weights are changed according to

$$w(E) \leftarrow w(E)/\phi,$$

where ϕ is the modification factor for the weights. Once the histogram has become "approximately flat" ϕ is reduced via the rule

$$\phi \leftarrow \sqrt{\phi}$$

and the histogram is reseted to 0 while the weights w are kept for the next iteration. This procedure is repeated until ϕ is close to one. In total there are the following parameters that have to be tuned:

- the energy interval $[E_{\min}, E_{\max}]$,
- the initial and final value of ϕ, ϕ^{init} and ϕ^{final} and
- the flatness criterion for the histogram.

Usually the weights $w(E)$ are underestimated in the low energy region in the early stage of the algorithm. This bias is successively corrected at each step of iteration. When ϕ^{init} is chosen too large, this kind of underestimation is very large, and a lot of computational effort has to be put for the correction in the following iterations. On the other side, using a small value of ϕ^{init} yields a long simulation time in the first iteration. In the applications here, the simulation of sequence alignments, as well as for the minimum (free-) energy distribution of the RNA secondary structure, a value of $\phi^{\text{init}} = e^{0.1}$ was a suitable compromise. The flatness criterion is not as crucial for the algorithm. Wang and Landau [WL01] suggested the requirement, that the number of visits of each energy level is at least 80% of the expected number for an uniform energy distribution, i.e.

$$h(E) > \epsilon^{\text{flat}} \frac{N}{k} \quad \forall E \in [E_{\min}, E_{\max}],$$

where k is the number of distinct energy levels on the interval and $N = \sum_E h(E)$ the total number of events. The choice of ϵ^{flat} depends on the problem, the choice of ϕ^{init} on the dimensionality of the weights to be estimated. In many cases it is sufficient to guarantee that the random walker has cycled several times from $E_{\max}$ to $E_{\min}$ in the energy space. The choice of ϕ^{final} requires some experimentation. Because the weights $w(E)$ are systematically underestimated for small energies (close to the ground state) the first hint for convergence is the change of the normalized weights between two iterations. The quality of convergence can be seen easily in the second part of the generalized ensemble simulation. If Wang-Landau sampling was stopped too early the random walker does not mix very well and get stuck in the low energy region. That can be detected by few Monte Carlo steps in the generalized ensemble.

Another reason for slow convergence in the Wang-Landau algorithm is due to the choice of the energy interval. If it was chosen too broad in the energy landscape, the sampler might also get stuck in local minima. This also happens in the parallel tempering approach as discussed above. the algorithm is applied.

2.7.2 Optimized ensembles

Perfectly flat histogram ensembles are only optimal in the sense, that all macrostates are visited with equal probability. There might remain large correlations due to the fact that the random walker stays in local minima for a long time. Especially near phase transitions, where the specific heat diverges, a huge amount of computation time is spent. This effect is known as *critical slowing down*.

This is also related to the *regeneration* of Markov chains in the following sense: A Markov chain is regenerative if there are times t_i, such that the process after t_i becomes independent from times before t_i.

The paths between regeneration points are called *tours*. Usually the distribution of tour lengths exhibits an heavy tail and only a very small fraction of tours hit one of the ground states. The *first-passage time* (also called *tunneling time*) is the time the random walker needs to hit the ground state starting at its last regeneration point. This is an extremal event and, hence the distribution of first-passage times might be, at least approximately, a generalized extreme-value distribution exhibiting a heavy tail.

Small first-passage times increase mixing and performance of the sampler. We will also use the *round-trip time*, which is the tunneling time plus the time needed to go back to regeneration. Since the turn from regeneration to the ground state is much longer than the turn back, first-passage time and round trip time are approximately equal.

Trebst et. al. [THT04] developed an iterative algorithm to optimize round-trip times in a generalized ensemble. Instead of giving all macrostates the weight $w(E) \propto 1/g(E)$ a different weight function $w^{\text{opt}}(E)$ is chosen, such that the *number of round trips* on an energy interval E_+ and E_- is maximized.

The equilibrium distribution of the optimized ensemble is proportional to $w^{\text{opt}}(E) \cdot g(E)$, which is not a flat histogram in general. The method works for both, for Metropolis and n-fold dynamics. In the iteration that is proposed in Ref. [THT04], on need to know the the fraction of visits $f(E)$ at energy E, where the last visit to E_+ has occurred more recently than to the state E_-. A sample estimate for f can be made by labeling the random walker with two different labels "$+$" or "$-$", depending on whether it has visited the state E_+ or E_- most recently. During the simulation a separate energy histogram $H_\pm(E)$ for each label is updated and an approximation of f for the given weights is given by

$$\hat{f}(E) = \frac{H_+(E)}{H_+(E) + H_-(E)}.$$

The derivative df/dE can be approximated by a polynomial interpolation of $f(E)$ and numerical derivation.

By diffusion arguments one shows that the weights of the optimized ensemble can be obtained iteratively [THT04]. The feedback iteration is given by

$$w^{k+1}(E) = w^k(E) \cdot \sqrt{\frac{1}{H_+(E) + H_-(E)} \cdot \frac{df}{dE}}, \tag{2.20}$$

where the histograms $H_\pm$ and the derivate $\frac{df}{dE}$ for the weights of iteration $(k+1)$ is obtained empirically from iteration k.

For the n-fold way, or as in the case of a semi rejection-free dynamics (see Chapter 7), the iteration has to account for the two intrinsic time scales. Since one aims at optimizing the computer time the iteration scheme Eq. (2.20) is modified by the factor $\tau(E)$, which is the accumulated waiting time at energy level E, in other words

$$w^{k+1}(E) = w^k(E) \cdot \sqrt{\frac{1}{H_+(E) + H_-(E)} \cdot \frac{df}{dE} \cdot \frac{1}{\tau(E)}}. \tag{2.21}$$

After each iteration the number of MC steps, which is used to accumulate the histograms, is doubled.

This method will be compared with other Monte Carlo methods in Chapter 7. Fig. 2.4 illustrates the convergence of $\hat{f}$ for the model of the RNA secondary structure. Details of the model will be introduced in Chapter 5, but it is not essential in the general description of the method in this chapter. A similar convergence behavior has also been observed in other systems such as Ising models [THT04]. In the first iteration the random walker spends much time for tours from energy level 0 towards low energy levels. After the optimization the fraction of tours in positive and negative direction become more balanced. Convergence is achieved after only 4 iterations. The inset shows the decrease of the round-trip time, which has already converged after the first iteration.

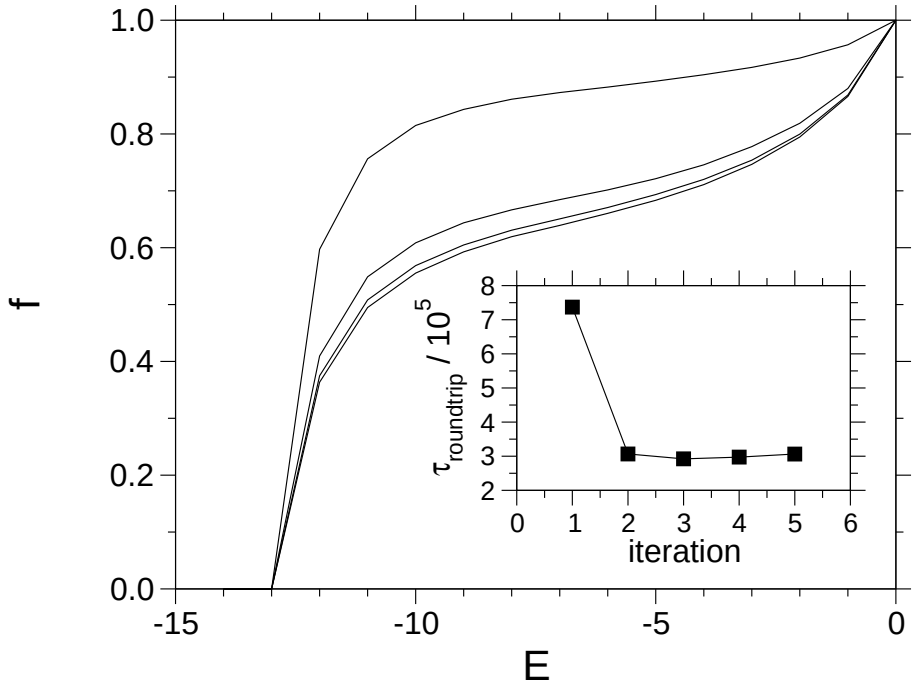


Figure 2.4: Convergence of $f(E)$ using iteration scheme Eq. (2.21), shown by lines connecting the data points, for better visibility. Between iteration 4th and 5th no significant difference of $f(E)$ is visible.
Inset: Convergence of round-trip times. Lines in the inset are guides to the eyes only.

2.8 Sampling of rare events III: Evaluation of the number of potential moves

During the course of a generalized ensemble or a parallel tempering simulation, more information than just the visited states is available. Instead of estimating the DOS from Eq. (2.12), matrix based methods aim at estimating the DOS from the *number of potential moves* (to be specified below) during a course of a Monte Carlo simulation. It is not crucial to use a particular ensemble. The only requirements are that the entire energy range is covered by the simulation and secondly that each state within each microcanonical ensemble $\chi_E = \{x \in \chi | E(x) = E\}$ is visited with equal probability. Simulations with detailed balance guarantee this *microcanonical property* automatically [Wan99a].

2.8.1 The density of states by transition matrix estimates

The connection between the transition matrix and the DOS can be made as follows:

Consider an infinitely long simulation in the canonical ensemble at infinite temperature, where *all* attempts are allowed. In entropic systems, like the RNA secondary structure, there might be *forbidden* configurations $E(x) = \infty$. Proposals that yield infinite energy are not accepted even at infinite temperature.

The discrete time and state master equation for the so constructed chain on the level of the macrostates is given by

$$p(E_j, t+1) = \sum_{E_i} Q_{i,j} \cdot p(E_i, t), \tag{2.22}$$

where $p(E_i, t)$ denotes the probability of finding macrostate state E_i at time t and $Q_{i,j}$ is the macrostate transition matrix, i.e. the probability of jumping to a state with energy E_j, given that the random walker sits in a state with energy E_i. Since $Q_{i,j}$ is stochastic we require that the columns sum to one, i.e. $\sum_j Q_{i,j} = 1$ for all i. The stationary distribution of Eq. (2.22) is the desired DOS $g(E)$. For a known infinite temperature transition matrix Q the stationary distribution can either be computed via solving the eigenvalue problem

$$g(E_j) = \sum_{E_i} Q_{i,j} \cdot g(E_i).$$

Alternatively one can also obtain $g(E)$ iteratively. This means one starts with some inital guess for $g(E)$ and applies Eq. (2.22) until the relative change of $g(E)$ is sufficiently small ($\approx 10^{-10}$) for all energy values.

The microstate infinite temperature transition matrix of the Metropolis algorithm is defined as

$$\Gamma_{x,y} = \begin{cases} 1 & \text{if } y \in \mathcal{N}(x) \text{ and } E(y) < \infty \\ 0 & \text{otherwise} \end{cases},$$

i.e. it equals 1 if the state y is a neighbor of x. Furthermore, let us define the number of neighbors of x with energy $E(x) + \Delta E$, or the number of potential moves, as

$$N(x, \Delta E) := \begin{cases} |\{y \in \mathcal{N}(x) | E(y) = E(x) + \Delta E\}| & \text{if } \Delta E \neq 0 \\ |\{y \in \mathcal{N}(x) | E(y) = \infty\}| & \text{if } \Delta E = 0 \end{cases}$$

The macrostate transition matrix Q can be written as a microcanonical expectation value of Γ,

$$\begin{aligned} Q_{i,j} &= \frac{1}{C_i} \sum_{x \in \chi} \left(\sum_{y \in \chi} \Gamma_{x,y} \cdot \delta_{E(y),E_j} \right) \cdot \delta_{E(x),E_i} \\ &= \frac{1}{C_i} \sum_{x \in \chi} N(x, E_j - E(x)) \cdot \delta_{E(x),E_i} \qquad (2.23) \\ &= \frac{1}{C_i} \langle N(x, E_j - E(x)) \rangle_{E_i} \qquad (2.24) \end{aligned}$$

where $\delta_{i,j}$ and the C_i's are chosen such that $Q_{i,j}$ is stochastic.

A sample estimate of $Q_{i,j}$ can be made [AHM$^+$88, WL00, HH05] from MC data on attempted moves: In the matrix $\hat{W}_{i,j}$ we count the number of attempted moves from i to j if the energy change is finite. Otherwise $\hat{W}_{i,i}$ is incremented by 1. Hence, the microcanonical average in Eq. (2.23) can be obtained from simulations in other ensembles than the microcanonical one.

Suppose we have data of m different simulations (a mixture of canonical ensembles or even a combination of generalized and canonical ensembles is possible), yielding transition matrices $\hat{W}^k_{i,j}$ with $k = 1 \dots n$. Then all data are added into one master matrix $\hat{W}_{i,j} = \sum_{k=1}^{n} \hat{W}^k_{i,j}$ and the DOS is determined from $\hat{Q}_{i,j} = \frac{1}{\sum_j \hat{W}_{i,j}} \hat{W}_{i,j}$.

2.8.2 The ParQ algorithm

Simulated annealing [KGV83, JJS06] is a stochastic optimization method that is inspired by the physical annealing process. The cost function is translated to the energy of the corresponding physical system and hence the temperature can be seen as parameter that controls the sub-optimality of the system. The Metropolis algorithm Algorithm 2.2.1 with Boltzmann weights $w(E) \propto \exp(-\beta E)$ and a time-dependent temperature schedule $\beta(t)$ provides the dynamics of the annealing procedure. By lowering the temperature successively, the system approaches low energy states and the simulation will end in a state close to the ground state. Geman and Geman [GG84] where able to proof the convergence of simulated annealing.

In most applications of simulated annealing, only the final state is of interest and the data during the course of the simulation is irrelevant. However, as has been illustrated above, information about the number of potential moves provides an estimate of the infinite temperature transition matrix and hence an estimate of the DOS [AHM$^+$88, HH05].

The *ParQ* method [HH05] aims at estimating the infinite temperature transition matrix (the letter "Q" in the acronym ParQ) from a parallel ("Par") run of independent simulated annealing simulations. The data from *all* simulations are collected afterwards and evaluated as described above.

It is an open question under which conditions the estimate converges toward the true DOS when the number of simulations, at a fixed number of MC steps each, tends to infinity. In section Chapter 7 we will examine convergence properties of the ParQ method for the RNA secondary structure and show that the microcanonical property is explicitly violated.

For the other limit, one simulated annealing run subject to infinitely slow cooling, the convergence can be conjectured by the convergence theorem for simulated anneal-

ing [GG84] and by the microcanonical property of the Metropolis algorithm, which is the limiting case of simulated annealing.

Chapter 3

Sequence alignment

Comparative genomics [Har03] is a young, growing research field that aims at studying the relationship between genetic information and functions across different organisms. Regions of coding DNA sequences, that are conserved between organisms play a key role in this framework, because conserved genetic information is also translated to related proteins which characterize the organisms' functions. For this reason the study of conserved regions in the sequences of protein molecules is a widespread approach.

Also DNA sequences that are responsible for gene regulation are conserved between related species. In analogy to the terms "genome" and "genomics", "protenome" and "protenomics" are often used, when looking at properties of the complete set of proteins in organisms.

Since the introduction of new biochemical methods in the 1970s the amount of molecular biological sequence information, which is the basis of comparative genomics, has increased dramatically. The history of genome sequencing projects began with the discovery of the chain termination method of DNA sequencing by Sanger and its application to the complete genome of $\Phi X174$ bacteriophage[SNC77]. Sanger determined this genome consisting of 5,386 bp (base pairs) manually. Over the years the method was improved towards computer aided sequencing [SKSH86] and further genomes with increasing sizes could be sequenced. Some milestones are the full genomic sequence of Saccharomyces cerevisiae (yeast) in 1996 (12,070,000 bp) [GBB+96] and Escherichia coli (a prokaryotic model organism) in 1997 (4,290,000 bp) [BIB+97]. Almost the complete genome of the model organism Drosophila melanogaster (fruit fly, 180,000,000 bp) was sequenced using the whole-genome shutgun method in 2000 [VAS+98, ACH+00, MSD+00] and finally a "working draft" of the human genome (2,700,000,000 bp) was obtained in 2001 [Int01] and finalized in 2003.

Large databases of DNA, proteins or RNA are available and international collaborations try to synchronize and standardize information and make them available by access tools. The information that is stored in the records of those databases ranges from sequences of bases or amino acids to citations of the corresponding publications. Also crosslinks (for example between genes and translated proteins) are very important.

An important resource is maintained by the *International Nucleotide Sequence Database Collaboration (INSDC)* [INS], which consits of three sub-organizations, the *European Molecular Biology Laboratory (EMBL)* [EMB], the *GenBank* hosted by the *National Center for Biotechnology Information (NCBI)*, USA [NCB] and the *DNA*

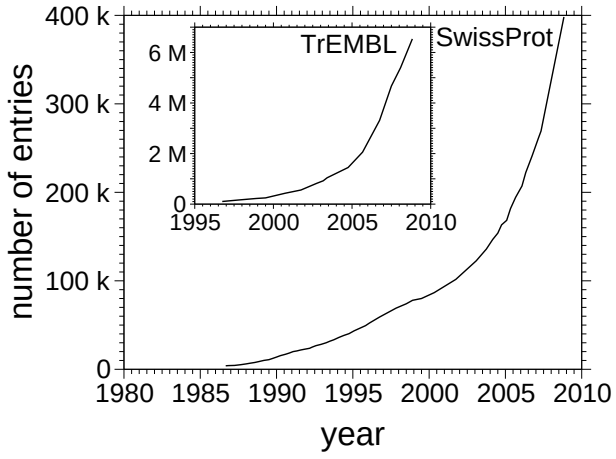


Figure 3.1: Growth of the SwissProt and TrEMBL database over the years

Data Bank of Japan (DDBJ) [DDB].

On the protein level the databases of the Universal Protein Resource Knowledgebase (UniProtKB) [Uni], *UniProtKB/SwissProt* and *UniProtKB/TrEMBL*, are popular resources. In order to give an impression of the increasing size of these databases, the number of entries (proteins) over the years is illustrated in Fig. 3.1. The SwissProt database contains manually checked and updated information on each entry, implying an high degree of usability. In contrast the "supplement" TrEMBL is generated by all EMBL nucleotide sequence entries by automatic translation, expect for those that are already integrated in SwissProt (TrEMBL stands for translated from EMBL). Most of the TrEMBL entries are seen as possible candidates for SwissProt. Because the TrEMBL entries are generated automatically much more sequences are stored in that database.

This exploding database sizes require computational tools that are able to analyze data, in particular searching for so called homological relationships [1], i.e. relationships due to common ancestry, between sequences.

This chapter is dedicated to *sequence alignment*, which is the workhorse of comparative genomics / protenomics. It is a method to quantify the similarity between two (*pairwise alignment*) or more (*multiple alignment*) biological sequences. Furthermore sequence alignment can be literally translated to a classical physical model with quenched disorder and, hence, many methods and concepts from statistical mechanics can be adopted to this problem. Some of them will be discussed in this chapter, others in the following one.

Basic definitions are introduced in Sec. 3.1 and after that the statistical inference of scoring parameters are outlined in Sec. 3.2. Sec. 3.3 treats optimization algorithms for pairwise alignment. Variants that also consider sub-optimal alignments are discussed in Sec. 3.4 followed by an outline of the so called *linear-logarithmic phase transition* in Sec. 3.5. Illustrative biological examples in Sec. 3.6 will close this chapter.

[1] In fact homology is a much broader term that describes similarities on different levels from entire organisms to the molecular biological level.

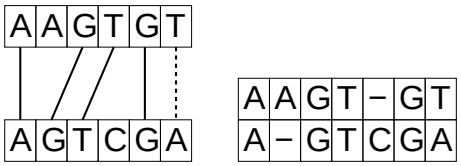


Figure 3.2: Two representations of an alignment. Matches are shown as solid lines and dashed lines indicate mismatches.
Left: Topological representation. Aligned letters are connected by lines.
Right: Bioinformatics representation. Aligned letters are connected by vertical bars and gaps are indicated by the symbol "-".

3.1 Notation of sequence alignment

As mentioned above pairwise alignment (in the following simply referred as "alignment") aims at measuring the similarity between two molecular sequences. Although we will focus on protein alignment alone later on, the presentation here will be general. This means amino acids or bases are referred as *letters* and the set of possible letters as *alphabet*. This is the four letter alphabet $\Sigma = \{A, T, G, C\}$ in the case of DNA and the 20 letter amino acid alphabet for proteins.

An alignment relates letters from one sequence to the second sequence. These relationships should reflect correlated regions between the sequences, that exhibits a large fraction of conserved letters in the evolutionary process. Since deletion or insertion of letters at certain positions in the sequences occur as well the concept of so called gaps is crucial. Formally we define an alignment by

Definition 3.1.1 *Let* $\mathbf{a} = a_1 \dots a_L \in \Sigma^L$ *and* $\mathbf{b} = b_1 \dots b_M \in \Sigma^M$ *a pair of sequences over* Σ.

(a) An alignment $\mathcal{A}$ *of* $\mathbf{a}$ *and* $\mathbf{b}$ *is an ordered set of pairings* $\mathcal{A} = \{(i_1, j_1), \dots, (i_N, j_N)\}$ *with* $1 \le i_k < i_{k+1} \le L$ *and* $1 \le j_k < j_{k+1} \le M$ *for* $k = 1 \dots N-1$. *The set of all alignments of* $\mathbf{a}$ *and* $\mathbf{b}$ *is denoted as* $\chi^{global}_{\mathbf{a},\mathbf{b}}$.

(b) Letters a_i *and* b_j *with* $(i,j) \in \mathcal{A}$ *are called aligned to each other and* a_i *and* b_j *aligned. If* $a_i = b_j$ *the pair* (i,j) *is denoted as match otherwise as mismatch. Letters that are not aligned are called unpaired or gaped.*

(c) Alignments with gaped letters are also called gaped.

(d) If $i_{k+1} = i_k + 1$ *and* $j_{k+1} = j_k + 1 + l$ *with* $l > 0$ *and* $(i_k, j_k), (i_{k+1}, j_{k+1}) \in \mathcal{A}$, *then* $\mathbf{b}$ *is said to contain a gap of length* l *between* i_k *and* i_{k+1} *and likewise for the sequence* $\mathbf{a}$. *If* $j_1 = l + 1 > 2$, *then* $\mathbf{b}$ *is said to have a gap of length* l *at the begin, if* $j_N = M - l < M$, *then* $\mathbf{b}$ *has a gap of length* l *at the end and likewise for the sequence* $\mathbf{a}$.

The conditions for the order of the aligned letters ensure that the relationships are not crossing, which implies polynomial alignment algorithms, see Sec. 3.3. Alignments can be represented in different ways. For example in Fig. 3.2 the alignment $\mathcal{A} = \{(1,1), (3,2), (4,3), (5,5), (6,6)\}$ of the input sequences $a = AAGTGT$ and $b = AGTCGA$ is shown in two different representations. In bioinformatics the symbol "-"

is commonly used to indicate gaps. Gaps are illustrated by inserting "-" between the corresponding positions. Note that each bioinformatics representation can be translated into the set of paired letters, whereas the reverse mapping is not unique. For example, the (suboptimal) alignment $\mathcal{A} = \{(1,1),(5,5),(6,6)\}$ can either be translated to

$$\begin{array}{l} A---AGTGT \\ |\qquad\quad|\,| \\ AGGC---GT \end{array} \quad \text{or to} \quad \begin{array}{l} AAGT---GT \\ |\qquad\quad|\,| \\ A---GGCGT \end{array}$$

amongst other representations. For this purpose we define a canonical bioinformatics representation by disallowing gaps in the first sequence $\mathbf{a}$ that follow a gap in second sequence $\mathbf{b}$. This implies that the second representation in the above example is not in its canonical form. This distinction is relevant not only for visualization of alignments, but also for the purpose to exclude redundant alignments from the partition function of suboptimal alignments (see Sec. 3.4).

The objective is a measure for the similarity or the degree of conservation between the sequences or regions of the sequences. The classical way is to assign a *score* for each alignment via an *objective function* $\mathcal{S} : \chi^{\text{global}}_{\mathbf{a},\mathbf{b}} \to \mathbb{R}$ and then maximizing $\mathcal{S}$ among all alignments

$$\begin{aligned} \mathsf{S}_0(\mathbf{a},\mathbf{b}) &= \max_{\mathcal{A}} \mathcal{S}(\mathcal{A};\mathbf{a},\mathbf{b}) \\ \mathcal{A}^{\text{opt}} &= \operatorname{argmax} \mathcal{S}(\mathcal{A};\mathbf{a},\mathbf{b}). \end{aligned} \tag{3.1}$$

For the choice of the objective function and its parameters we need to know

(i) whether we are looking at locally conserved region or whether the entire sequences should be considered,

(ii) how matches and mismatches should be evaluated and

(iii) how gaps should penalize the overall score.

To address the first issue there are in principle two types of objective functions, namely *optimal local alignment scores* $\mathsf{S}_0^{\text{local}}$ and *optimal global alignment scores* $\mathsf{S}_0^{\text{global}}$. Optimal global alignment scores involve contributions from all matches, mismatches and gaps. The optimal local alignment score is the optimum over all global alignments of all subsequences of $\mathbf{a}$ and $\mathbf{b}$,

$$\mathsf{S}_0^{\text{local}}(\mathbf{a},\mathbf{b}) = \max_{\substack{1\le i'<i\le L \\ 1\le j'<j\le M}} \mathsf{S}_0^{\text{global}}(a_{i'}\dots a_i, b_{j'}\dots b_j). \tag{3.2}$$

Alternatively $\mathsf{S}_0^{\text{local}}(\mathbf{a},\mathbf{b})$ can be seen as a global alignment where gaps at the begin or end of the sequences are not penalized. Formally we define the state space of local alignments by (see also Fig. 3.3)

Definition 3.1.2 *Let* $\mathbf{a} = a_1\dots a_L \in \Sigma^L$ *and* $\mathbf{b} = b_1\dots b_M \in \Sigma^M$ *be a pair of sequences over* Σ*. The set of local alignments of* $\mathbf{a}$ *and* $\mathbf{b}$ *is given by*

$$\chi^{\textit{local}}_{a,b} = \bigcup_{\substack{1\le i'<i\le L \\ 1\le j'<j\le M}} \{\chi^{\textit{global}}_{a_{i'}\dots a_i, b_{j'}\dots b_j} | a_{i'}, b_{j'} \textit{ and } a_i, b_j \textit{ aligned to each other}\} \bigcup \{\}$$

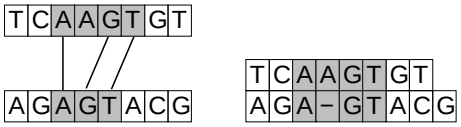


Figure 3.3: Representations of a local alignment.
Left: Topological representation.
Right: Bioinformatics representations. Aligned letters are connected by vertical bars and gaps are indicated by the symbol "-".

This definition is introduced in order to avoid overcounting of local alignments that have a gap at the begin or the end of either sequence, because all these alignments have an unique topological representation.

The second issue requires the knowledge of a relationship between the letters of the underlying alphabet. This is usually realized by so called *substitution or score matrices* that assign each pair of letters a real number, i.e. $\sigma : \Sigma \times \Sigma \to \mathbb{R}$. In generic studies often a very simple score matrix is used. This matrix assigns all mismatches the same number $-\mu$ ($\mu > 0$) and all matches 1,

$$\sigma(a,b) = \begin{cases} 1 & \text{if } a = b \\ -\mu & \text{otherwise} \end{cases} .$$

However, for the case of protein alignment this setup is strongly oversimplified, because different types of amino acid substitution are more or less functionally conservative. That means certain substitutions affect physical and chemical properties of the protein, than others, because the amino acids themselves share similar properties. Mutations between such related amino acids occur more likely. In most cases the score matrices are derived by so called log-odds ratios that compare probabilities between two models

$$\sigma(a,b) = \log \frac{\mathrm{Prob}(a,b \,|\, a \text{ and } b \text{ are related })}{\mathrm{Prob}(a,b \,|\, a \text{ and } b \text{ are unrelated })},$$

see Sec. 3.2 for details. The contribution of all letters due to matches or mismatch is then a sum over all these contribution.

Regarding the gaps one compromises between a computational feasible and biological evident penalty function g. That means each gap Γ of length l_Γ yields a negative contribution of $-g(l_\Gamma)$ to the overall score, which is then defined as

$$\mathcal{S}(\mathcal{A};\, \mathbf{a}, \mathbf{b}) = \sum_{(i,j)\in\mathcal{A}} \sigma(a_i, b_j) - \sum_{\Gamma} g(l_\Gamma) \tag{3.3}$$

Usually g is a monotonously increasing function of the length. The alignment algorithms for gaped alignments with arbitrary gap penalties exhibit a cubic time complexity ($\mathcal{O}(\max(L,M)^2 \min(L,M))$). In practice *affine gap* cost functions

$$g(l_\Gamma) = \alpha + \beta\,(l_\Gamma - 1), \qquad \text{with } \alpha > \beta \tag{3.4}$$

are commonly used, because the computational complexity reduces to $\mathcal{O}(LM)$ [Got82]. The opening parameter α penalizes an opening of a gap, i.e. it is a general contribution for the existence of a gap at all. The extension parameter β is usually chosen to be smaller than α, hence

(i) longer gaps are penalized more than shorter ones and

(ii) opening a gap is always more expensive than extensions.

The affine gap penalty Eq. (3.4) is based on the empirical observation that the size distribution of insertions and deletions in evolutionary processes is well described by a power-law distribution [GCB92, GL95]

$$P(l_\Gamma) = P(1)\, l_\Gamma^{-b}, \quad l_\Gamma = 1, 2, \ldots$$

with $b \approx 1.8$. Together with the ratio a of insertions or deletions to substitutions or conservations the natural gap penalty has a log-affine form

$$g(l_\Gamma) \propto \log a + b \log l_\Gamma,$$

which is computational more expensive than linear-affine gap costs. However in a recent study the commonly used affine form Eq. (3.4) has been justified by benchmarking the accuracy of logarithmic $g(l_\Gamma) = \beta \log l_\Gamma$, log-affine $g(l_\Gamma) = \alpha + \beta \log l_\Gamma$ and linear-affine $g(l_\Gamma) = \alpha + \beta l_\Gamma$ gap costs [Car06]. The result is that the practical choice of linear-affine gap costs approximates the realistic case very well. Pure logarithmic gap costs are significantly worse. In praxis some heuristics for the optimal choice of scoring matrices and corresponding gap costs is required (see for example [RP02, VEA95]).

Before describing in Sec. 3.3 the optimization procedure to perform the maximization Eq. (3.1) for global and local alignment in polynomial time, more about the choice of the protein scoring matrices σ is said in the next section.

3.2 Scoring models

Now, as we have fixed the notation of sequence alignment, it is possible to describe the methods, which had been used to derive scoring matrices σ for protein alignment. There are three approaches relevant in this scope, the `PAM` [DSO78] and `BLOSUM` matrices [HH92], which are most common, and a special purpose scoring matrix for transmembrane proteins `SLIM` [MRR01].

3.2.1 The `PAM` family

The starting point of the derivation of the `PAM` matrices [DSO78] are *phylogenetic trees* of closely related proteins (see Fig. 3.4). Usually only sequence data of the leafs of such tree is available and the rest of the tree is constructed by parsimony methods [RDM98, CB05]. At each edge of the tree of proteins a mutation due to a change in the gene, i.e. in the coding region of the DNA, occurred. This allows one to construct such trees not only for DNA but also for proteins.

Trees of sequences such as in Fig. 3.4 can be described as a series of mutations, which are defined as follows.

Definition 3.2.1 *Let Σ be an alphabet. An accepted point substitution is an operation on a sequence $\mathbf{a}$ over Σ*

$$\Sigma^n \to \Sigma^n : a_1 \ldots a_{i-1} a_i a_{i+1} \ldots a_n \to a_1 \ldots a_{i-1} a_i^* a_{i+1} \ldots a_n \in \Sigma^n$$

with $1 \leqslant i \leqslant n$ and a letter $a_i^* \neq a_i$.

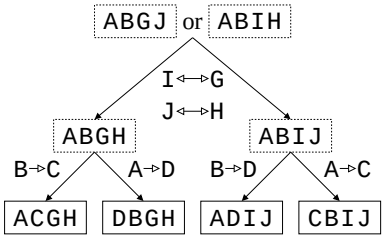


Figure 3.4: Sketch of a phylogenetic tree of a protein family. Such trees are usually derived from the sequences that are located at the leafs (bordered by solid lines) At each edge the mutations are counted. Note that mutations occurred less frequently in the data set used in Ref. [DSO78] than in this illustration.

Dayhoff, Schwartz and Orcutt [DSO78] have chosen 71 families of closely related proteins. That means each pair of sequences does not differ in more than 15% of letters. This implies the reasonable assumption that each site is changed at most once. Also insertions or deletions of letters are not considered in this study.

From the constructed trees the total number of inferred accepted point mutations from letter $a \in \Sigma$ to $b \in \Sigma$ are counted in an matrix $A_{a,b}$. This matrix is constructed symmetrically which means whenever a mutation from a to b is observed $A_{a,b}$ and $A_{b,a}$ is incremented by 1. The normalized frequencies of all amino acids f_a, $\sum_{a\in\Sigma} f_a = 1$, in all branches are estimated from the data. This allows a definition of the so called *mutation probability matrix* via

$$P_{a,b} = \Lambda \frac{A_{a,b}}{\sum_{c\neq a} A_{a,c}} \text{ for } a \neq b \quad \text{and} \quad P_{a,a} = 1 - \Lambda \text{ for } a = b,$$

where Λ is a tunable parameter that is specified below. This matrix exhibits following properties:

- the rows are normalized to one, $\sum_b P_{a,b} = 1$, i.e. P is stochastic and
- the average fraction of amino acids that are changed is given by $\sum_{a\neq b} f_a\, f_b\, P_{a,b}$.

The last property allows one to choose the scale parameter Λ in a practical way based on the following definition:

Definition 3.2.2

(a) Two sequences $\mathbf{a}, \mathbf{b} \in \Sigma^L$ are said to have an evolutionary distance of k PAM (point accepted mutations), if they have evolved by a series of $100 \times k/L$ accepted point substitutions, i.e. k is the average number of substitutions per 100 residues. The unit PAM defines an evolutionary distance.

(b) The matrix $P^{(1)}_{a,b}$ is said to be the $1PAM$ mutation probability matrix, if Λ is chosen such that $\sum_{a\neq b} f_a\, f_b\, P^{(1)}_{a,b} = 0.01$.

The definition of the $1PAM$ mutation probability matrix is the starting point of the derivation of matrices for larger distances, when considering the sequence evolution as a Markov process. That means that an initial composition $\hat{f}_a$ will evolve to $\hat{f}_b = \sum_a \hat{f}_a P^{(1)}_{a,b}$ for an one PAM process. Larger distance can be obtained by repeating the

process k-times. This yields the $kPAM$ mutation probability matrix $P^{(k)}$ as a power of the $1PAM$ matrix.

$$P^{(k)} = \underbrace{P^{(1)}\ P^{(1)}\ \dots P^{(1)}}_{k \text{ times}} = \left[P^{(1)}\right]^k .$$

In order to translate the probability matrix P into a score matrix σ, let us consider a gapless global alignment of two sequences $\mathbf{a}$ and $\mathbf{b}$ (or a gapless local alignment of $\mathbf{a}$ and $\mathbf{b}$). We want to discriminate the hypothesis that both sequences (or subsequences in the case of local alignment) stem from a common ancestor according to the process described above (model M_1) against the *null hypothesis* which states that both are unrelated and their composition is i.i.d. according to the background frequencies f_a (model M_0). For model M_1 we further assume that all sequences have an average composition f_a and that is equally likely to replace a with b or vise versa. Hence the probability of observing the pair a, b is given by $\hat{P}_{b,a} = \hat{P}_{a,b} = f_a P_{a,b}$. . The probabilities that the observation $\mathbf{a}$ and $\mathbf{b}$ are described by the model of M_0 or M_1 are given by

$$\begin{aligned} \mathrm{Prob}(\mathbf{a}, \mathbf{b}|M_0) &= \prod_{i=1}^{N} f_{a_i} \prod_{i=1}^{N} f_{b_i} \quad \text{and} \\ \mathrm{Prob}(\mathbf{a}, \mathbf{b}|M_1) &= \prod_{i=1}^{N} \hat{P}_{a_i,b_i} \end{aligned}$$

respectively. A common measure of discrepancy between two models is the so called log-odd ratio,

$$\mathsf{S} = \log \frac{\mathrm{Prob}(\mathbf{a}, \mathbf{b}|M_1)}{\mathrm{Prob}(\mathbf{a}, \mathbf{b}|M_0)} = \log \prod_{i=1}^{N} \frac{\hat{P}_{a_i,b_i}}{f_{a_i}\, f_{b_j}} = \sum_{i=1}^{N} \log \frac{\hat{P}_{a_i,b_i}}{f_{a_i}\, f_{b_j}}.$$

This also justifies the fact that the contributions for matches and mismatches in Eq. (3.3) was chosen additive and we can identify

$$\sigma^{\mathsf{PAM}}(a, b) = \log \frac{\hat{P}_{a,b}}{f_a\, f_b}$$

Apparently, σ depends on the PAM distance. For this reason different PAM score matrices for different purposes have been derived. This matrices are denoted as "PAMk". Hence the acronym "PAM" has two meanings, the evolutionary time unit as well as the name of the matrix family.

Popular matrices are for example `PAM30` or `PAM250`. These matrices are usually scaled by an factor ($3/\log 2$ in the case of `PAM250`) and then rounded to integer. The more distant a pair of sequences is suspected to be the larger the PAM value should be chosen. Hence the methods requires some experience to choose the best substitution matrix. To be on the safe side, it is also possible to use a combination of different matrices for a single biological question [FFB04].

3.2.2 The `BLOSUM` family

The PAM approach is most powerful for shorter distances because it relies on an extrapolation of the 1 PAM matrix. Fourteen years after the introduction of the PAM family,

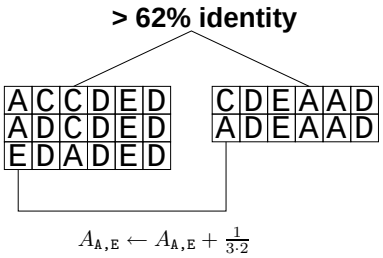


Figure 3.5: Construction of the BLOSUM family from clusters of highly conserved blocks.

Heinkoff and Heinkoff derived another family of substitution matrices, the BLOSUM family [HH92], which is most commonly used today, in particular to detect distant relationships.

This matrices have been derived from known so called ungapped multiple alignments. That are blocks of conserved regions of equal length which are written in top of each other, similar as in pairwise alignment. Each column represents a position and each row is a fragment of a protein sequence. These blocks are clustered according to the percentage of identical letter at each site (see Fig. 3.5). The clusters are constructed such that for each sequence within $k\%$ cluster there is at least one sequence in the same cluster with $k\%$ or more identical letters..

A matrix $A_{a,b}$ counts the weighted frequencies of letter a and b in different clusters, i.e. each time letter a is observed in a cluster and letter b is observed in the same column but different cluster $A_{a,b}$ is incremented by $1/n_1 n_2$, where n_1 and n_2 are the sizes (number of sequences) of the respective clusters. In order to avoid overcounting of many highly similar sequences, when estimating the amino acid background frequencies f_a, Heinkoff and Heinkoff have provided a unbiased estimate via $A_{a,b}$, instead of simply counting amino acids,

$$f_a = \frac{\sum_b A_{a,b}}{\sum_{c,d} A_{c,d}}.$$

Again, the score matrix is a log-odd score of pair probabilities $\frac{A_{a,b}}{\sum_{c,d} A_{c,d}}$ and the background frequencies, i.e.

$$\sigma^{\mathtt{BLOSUM}}(a,b) = \log \frac{P_{a,b}}{f_a\, f_b}$$

By the construction of $A_{a,b}$ the score matrix $\sigma^{\mathtt{BLOSUM}}(a,b)$ is symmetric. The percentage threshold value that defines the cluster is the analogue of the PAM distance. The difference is that a large value, i.e. a large fraction of identical letters, yields a matrix that is more sensitive for distant proteins. The standard matrix from that family is BLOSUM62, where all blocks with at least 62% identical letters are clustered.

3.2.3 Position specific scoring for transmembrane proteins using the SLIM family

Transmembrane proteins are important players in the molecular biology of the cell [AJL$^+$08]. They extend from one side of the cell membrane (a so called lipid bilayer)

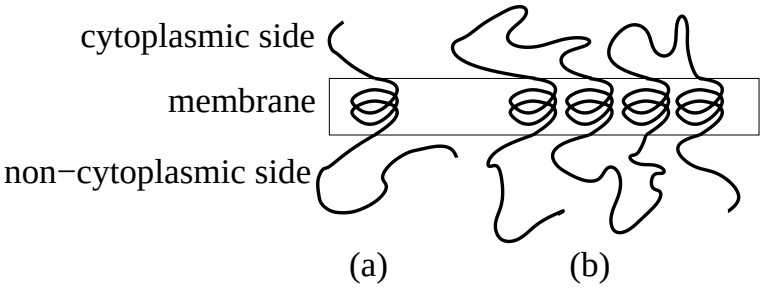


Figure 3.6: Transmembrane proteins crossing the lipid bilayer once (a) or several times (b). The transmembrane region is characterized by helical structures in most cases.

to the other one either once (see Fig. 3.6 (a)) or several times (Fig. 3.6 (b)). The amino acid composition and the three dimensional structure strongly depends on the structural domain, i.e. the sub sequence that is part of a certain element of the three dimensional structure. Domains inside the cell, the so called *cytoplasmic side*, and domains outside the cell are characterized by a typical hydrophilic amino acid composition. Whereas the membrane domains exhibits a hydrophobic composition resulting in helical structures in most cases [AJL$^+$08].

The strong difference in the composition for transmembrane domains and the rest of the molecule implies that the standard matrices such as the PAM or the BLOSUM families are less powerful because they rely on general background frequencies and pair probabilities. A score matrix to align two transmembrane domains has been derived by Jones [JTT94]. This kind of alignment is desirable when one is interested in finding good alignments between two transmembrane proteins rather than discriminating between different sequences. For this purpose Ng, Heinkoff and Heinkoff [NHH00] derived a special purpose matrix, PHAT (*Predicted Hydrophobic And Transmembrane*), which accounts for hydrophobic bias and is designed to use in database search.

On this basis Müller, Rahmann and Rehmsmeier derived the SLIM (*Scorematrix Leading to Intra-Membrane domains*) family [MRR01] that is designed to align transmembrane regions against "general" regions, as it occurs in cases where transmembrane queries are searched against huge general protein databases. It is explicitly nonsymmetric by construction,

$$\sigma^{\texttt{SLIM}}(a,b) = \frac{3}{\ln(2)} \ln\left(\frac{T_{a,b}}{f_a^{\text{TM}}\, f_b}\right).$$

The construction of the matrix T is based on a generalization of Dayhoff's PAM approach to continuous time Markov processes [MV00] based on the pair probabilities of the PHAT matrix, details can be found in [MRR01]. The subject frequencies f_b have been taken from the general purpose matrix VTML [MV00], whereas the frequencies $\left\{f_a^{\text{TM}}\right\}$ stem from the software tool PHDhtm [RFC96], that allows predicting transmembrane helical regions by a neural network approach.

A typical application for a database search requires the prediction of the positions of the transmembrane regions[2] of the query. With this data the database is searched

[2]Beside PHDhtm, there are numerous approaches for the same purpose available [vH92, NK92, HBCM98, TS98, CWS$^+$97, PA97, SvHK98, KLvHS01, DWL$^+$01, AMI$^+$04, KKS04, KKS05, Jon07]. I

with the *position specific scoring scheme*

$$\mathcal{S}(\mathcal{A},\ \mathbf{a},\mathbf{b}) = \sum_{(i,j)\in\mathcal{A}} \begin{cases} \sigma^{\mathtt{slim}}(a_i,b_j) & \text{if } i \text{ is a transmembrane position} \\ \sigma^{\mathtt{blosum}}(a_i,b_j) & \text{otherwise} \end{cases} - \sum_{\Gamma} g(l_\Gamma) \tag{3.5}$$

instead of Eq. (3.3) for general purpose scoring.

3.3 Optimal alignment

In the following, the optimization algorithms by Needleman and Wunsch [NW70] and Smith and Waterman [SW81] for the global and local alignment are described.

3.3.1 Global alignment

The optimization problem to find the optimal global pairwise alignment allowing gaps can be solved by *dynamic programming* (known as transfer matrix method in statistical physics). It is referred as *Needleman-Wunsch algorithm* [NW70], which was originally designed for linear gap costs, where $\alpha = \beta$ in Eq. (3.4). A modified version also allows for general affine gap costs, where $\alpha > \beta$ within the same time complexity class of $\mathcal{O}(LM)$ [Got82]. The computation requires three auxiliary matrices of size $L \times M$. In fact the computation can also be performed in linear space $\mathcal{O}(\max(L,M))$ [Hir75], but memory efficiency is not essential here.

Following the paradigms of dynamic programming [CLR02], we divide the problem in subproblems. Let us define the matrix elements $D_{i,j}$, $P_{i,j}$ and $Q_{i,j}$ by the optimal score of the subproblem $a_1 \dots a_i$ and $b_1 \dots b_j$, given that a_i and b_j are aligned, given that a_i is gaped and given that b_j is gaped respectively, i.e.

$$\begin{aligned} D_{i,j} &:= \mathsf{S}_0^{\text{global}}\left(a_1\dots a_i, b_1\dots b_j;\ \text{given} \begin{pmatrix} a_i \\ | \\ b_j \end{pmatrix}\right) \\ P_{i,j} &:= \mathsf{S}_0^{\text{global}}\left(a_1\dots a_i, b_1\dots b_j;\ \text{given} \begin{pmatrix} a_i \\ - \end{pmatrix}\right) \\ Q_{i,j} &:= \mathsf{S}_0^{\text{global}}\left(a_1\dots a_i, b_1\dots b_j;\ \text{given} \begin{pmatrix} - \\ b_j \end{pmatrix}\right) \end{aligned} \tag{3.6}$$

The case that both sequences end up in a gap is not possible according to the topological definition Def. 3.1.1.

We assume that the matrix elements of D, P, Q are known for all indices $(i',j') < (i,j)$, where "$<$" denotes *lexicographic ordering*[3]. In particular this applies to the indices $(i-1,j-1)$, $(i-1,j)$, $(i,j-1)$. Then the so far unknown matrix elements

have used TMHMM (Transmembrane Hidden Markov Model) [SvHK98, KLvHS01], because recent benchmark results are convincing [CDS05] (see Sec. 4.5)

[3] $(i',j') < (i,j)$ if and only if $i' < i$ or $i' = i$ and $j' < j$

$D_{i,j}$, $P_{i,j}$,$Q_{i,j}$ can be determined by the recursion relations

$$D_{i,j} = \sigma(a_i, b_j) + \max \begin{cases} D_{i-1,j-1} \\ P_{i-1,j-1} \\ Q_{i-1,j-1} \end{cases} \tag{3.7a}$$

$$P_{i,j} = \max \begin{cases} D_{i-1,j} - \alpha \\ Q_{i-1,j} - \alpha \\ P_{i-1,j} - \beta \end{cases} \tag{3.7b}$$

$$Q_{i,j} = \max \begin{cases} D_{i,j-1} - \alpha \\ Q_{i,j-1} - \beta \end{cases} \tag{3.7c}$$

with the boundary conditions

$$\begin{aligned} D_{0,0} &= 0 \\ P_{0,0} &= -\infty \\ Q_{0,0} &= -\infty \\ D_{i,0} &= -\infty && \text{for } i = 1 \dots L \\ D_{0,j} &= -\infty && \text{for } j = 1 \dots M \\ P_{i,0} &= -\alpha - (i-1)\beta && \text{for } i = 1 \dots L \\ Q_{0,j} &= -\alpha - (j-1)\beta && \text{for } j = 1 \dots M \end{aligned}$$

and the final result

$$\mathsf{S}_0^{\text{global}}(a_1 \dots a_L, b_1 \dots b_M) = \max\{D_{L,M}, P_{L,M}, Q_{L,M}\}\,.$$

After filling the matrices a trace back procedure can be applied to determine an optimal alignment $\mathcal{A}^{\text{opt}}$. Because a gap in the second sequence cannot follow a gap in the first one by definition, only two cases occur in the recursion Eq. (3.7c). This is not essential for optimal alignments, but turned out to be important for finite temperature alignments.

Because optimal alignments might be exponentially degenerate (in particular in the linear phase, see Sec. 3.5) one has to distinguish, whether

(i) one is interested in an arbitrary optimal alignment, or whether

(ii) one (ore more) alignment is to be chosen with the correct statistical weight, where each optimal alignment is equally likely.

The second task requires additional matrices N^D, N^P, N^Q that account for the degeneracy of the corresponding subproblems. These matrices can be computed in a similar way as the recursion Eq. (3.7),

$$\begin{aligned} N^D_{i,j} &= N^D_{i-1,j-1}\,\delta_{D_{i,j},\,D_{i-1,j-1}+\sigma(a_i,b_j)} + \\ &\quad N^P_{i-1,j-1}\,\delta_{D_{i,j},\,P_{i-1,j-1}+\sigma(a_i,b_j)} + \\ &\quad N^Q_{i-1,j-1}\,\delta_{D_{i,j},\,Q_{i-1,j-1}+\sigma(a_i,b_j)} \\ N^P_{i,j} &= N^D_{i-1,j}\,\delta_{P_{i,j},\,D_{i-1,j}-\alpha} + \\ &\quad N^Q_{i-1,j}\,\delta_{P_{i,j},\,Q_{i-1,j}-\alpha} + \\ &\quad N^P_{i-1,j}\,\delta_{P_{i,j},\,P_{i-1,j}-\beta} \\ N^Q_{i,j} &= N^D_{i,j-1}\,\delta_{Q_{i,j},\,D_{i,j-1}-\alpha} + \\ &\quad N^Q_{i,j-1}\,\delta_{Q_{i,j},\,Q_{i,j-1}-\beta} \end{aligned}$$

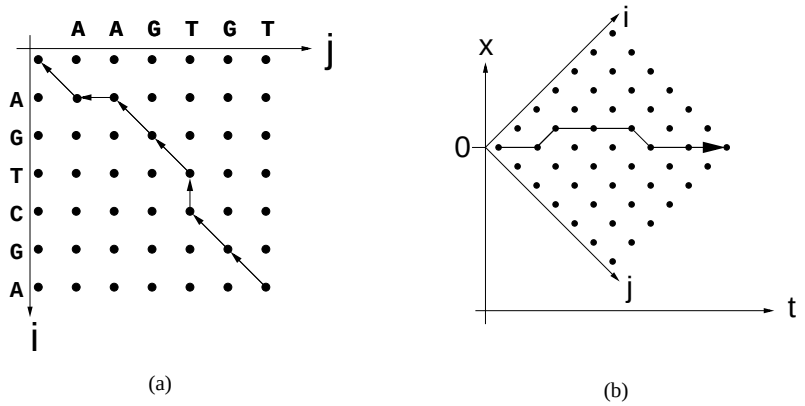


Figure 3.7:
(a) Path graph representation of a global alignment.
(b) Interpretation as DPRM

To perform the backtracing we set up a rectangular lattice of size $(L+1)\times(M+1)$ (see Fig. 3.7(a)) Each pair $(0,0) \leq (i,j) \leq (L,M)$ corresponds to a site. The optimal alignment corresponds to a *minimum weighted directed path* from the site (L,M) to $(0,0)$. This yields third alignment representation denoted as *path graph* (see Fig. 3.7(a)).

The weights depend on the choices that have been made during the dynamic programming procedure in forward direction and that are stored in the matrices D, P and Q. Initially (i,j) is set to (L,M) and we define $F_{i,j} = \max\{D_{i,j}, P_{i,j}, Q_{i,j}\}$ as the unconditioned optimal socres of the subproblems. At each step it is determined which of the three matrix elements is chosen as maximum and go back in $(-1,-1)$, $(-1,0)$ or $(0,-1)$ direction depending on whether $F_{i,j} = D_{i,j}$, $F_{i,j} = P_{i,j}$ or $F_{i,j} = Q_{i,j}$ respectively. In the case that the choice is not unique the direction is chosen randomly weighted with the corresponding degeneration that are stored in N^D, N^P and N^Q. To translate the path graph representation into the standard bioinformatics representation (see Fig. 3.7(b)) the symbols

$$\begin{pmatrix} a_i \\ | \\ b_j \end{pmatrix} \quad \begin{pmatrix} - \\ \\ b_j \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} a_i \\ \\ - \end{pmatrix}$$

are attached in front of the alignment, depending on the direction that has been chosen. That means a step in diagonal direction corresponds to a match or mismatch, horizontal or vertical steps to gaps in either sequence.

Assuming that $u_{i,j} \equiv \sigma(a_i, b_j)$ are independent, local alignment with linear gap costs $\delta \equiv \alpha = \beta$ is equivalent to one of the best studied models in statistical physics, *directed paths in random media in (1+1) dimensions* (DPRM) [HH85, Kar87, Mez90, FH91, Kar94] (see Fig. 3.7(b)). This model describes a path that is directed in a positive temporal dimension, which is related to (i,j) via $t = i + j - 1$, and may fluctuate in one spatial dimension $x = i - j$. The energy of the path corresponds to the negative score and δ can be interpreted as a "line tension", which forces the path to follow a straight line. In contrast, $u_{i,j}$ corresponds to a random potential that is responsible for fluctuations in the spatial dimension. This analogy has inspired researchers to study

global sequence alignment from a statistical mechanics perspective and look at properties like localization-delocalization transition [HL96], percolation [SAY05] or apply methods like replica calculations [Yu04]. In order to study suboptimal alignments also a finite-temperature version has been formulated, see [Miy95, ZM95, KL00, MHS02] and Sec. 4.6.

3.3.2 Local alignment

At the first glance the optimization problem for local alignment Eq. (3.2) is much more complex. Interestingly it can be solved within the same complexity as the global alignment problem. The matrices P and Q are defined in the same way as in Eq. (3.7) except that the global objective function $\mathsf{S}_0^{\text{global}}$ is replaced with $\mathsf{S}_0^{\text{local}}$. Note that an integral difference between local and global alignment is that gaps at the begin and end of a or b are not penalized. In particular the null alignment $\mathcal{A}^0 = \{\}$ has score 0 and it is always better to choose $\mathcal{A}^0$ instead of an alignment with negative score. Hence we define

$$D_{i,j} := \mathsf{S}_0^{\text{local}} \left(a_1 \dots a_i, b_1 \dots b_j; \text{ given } \begin{pmatrix} a_i \\ | \\ b_j \end{pmatrix} \right)$$

Consequently, the recursion of for the local alignment is given by

$$D_{i,j} = \sigma(a_i, b_j) + \max \begin{cases} 0 \\ D_{i-1,j-1} \\ P_{i-1,j-1} \\ Q_{i-1,j-1} \end{cases} \tag{3.8a}$$

$$P_{i,j} = \max \begin{cases} D_{i-1,j} - \alpha \\ Q_{i-1,j} - \alpha \\ P_{i-1,j} - \beta \end{cases} \tag{3.8b}$$

$$Q_{i,j} = \max \begin{cases} D_{i,j-1} - \alpha \\ Q_{i,j-1} - \beta \end{cases} , \tag{3.8c}$$

which is almost identical to the global alignment problem. Due to the fact that alignments with gaps at the begin of a or b are never chosen as optimal and because optimal local alignments can start at any point the boundary conditions are given by

$$\begin{aligned} D_{i,0} = P_{i,0} = Q_{i,0} &= -\infty \qquad \text{for } i = 0 \dots L \\ D_{0,j} = P_{0,j} = Q_{0,j} &= -\infty \qquad \text{for } j = 0 \dots M \end{aligned}$$

This is defines the famous Smith-Waterman algorithm, which was originally proposed for linear gap costs $\alpha = \beta$ [SW81].

The end of a local alignment can be any point in $(i, j) \leq (L, M)$ and hence the optimum is

$$\mathsf{S}_0^{\text{local}}(a_1 \dots a_L, b_1 \dots b_M) = \max \left\{ \max_{(i,j) \leq (L,M)} \{D_{i,j}\}, 0 \right\}.$$

The back-tracing procedure is quite similar to that of global alignment. Initially the end point is set to the position of the maximum (instead of (L, M) for global alignment). Then the traceback is performed in the same way as global alignment until a point with $D_{i,j} = \sigma(a_i, b_j) = \max\{D_{i,j}, P_{i,j}, Q_{i,j}\}$ is reached, which is then the starting point of the alignment.

3.4 Finite-temperature local alignment

Sometimes it is also desirable to consider alternative sub-optimal alignments. One approach to detect alternative local alignments beyond the optimum is to consider a canonical ensemble of local alignments for each pair of sequences. Hence, in the jargon of statistical mechanics, sequence alignment is a disordered system with quenched disorder. Each pair of sequences corresponds to one realization of the disorder inducing a canonical ensemble on its own. We interpret the score $\mathcal{S}$ as a negative energy and, hence, the optimal alignment S_0 reads as the ground state of the system.

The partition function of the canonical ensemble at temperature T of all local alignments is given by

$$Z = \sum_{\mathcal{A} \in \chi_{a,b}} e^{\mathcal{S}(\mathcal{A};a,b)/T}.$$

The temperature plays the role of a control parameter which gives suboptimal alignments more weight with increasing value. In the infinite temperature limit all alignments have equal weight, i.e. the entropy dominates. The free energy $F = T \log Z$ is the finite temperature analogue of the optimal score and

$$\lim_{T \to 0} F = \mathsf{S}_0^{\text{local}}$$

The partition function version of the alignment problem was proposed by Zhang and Marr [ZM95] as well as by Miyazawa [Miy95] at about roughly the same time. The first authors suggest to use the partition function formalism for an algebraic expansion in the scoring parameters in order to investigate the parametric dependence of the free energy [ZM95]. A similar approach for the optimal alignment was proposed by Waterman [Wat94]. The connection between information theory and reliability of finite temperature alignments has been worked out by Kschischo and Lässig [KL00].

As the Smith Waterman algorithm (Eq. (3.8)), the partition function version of the Smith-Waterman algorithm requires three auxiliary matrices. That is the partition function of all local alignments ending at (i, j), $Z^D_{i,j}$ and two matrices of all non-canonical alignments that end in a gap in either sequence. The corresponding recursion relation reads as

$$Z^D_{i,j} = \left(1 + Z^D_{i-1,j-1} + Z^P_{i-1,j-1} + Z^Q_{i-1,j-1}\right) \cdot e^{\sigma(a_i,b_j)/T} \tag{3.9a}$$

$$Z^P_{i,j} = \left(Z^D_{i-1,j} + Z^Q_{i-1,j}\right) \cdot e^{-\alpha/T} + Z^P_{i-1,j} \cdot e^{-\beta/T} \tag{3.9b}$$

$$Z^Q_{i,j} = Z^D_{i,j-1} \cdot e^{-\alpha/T} + Z^Q_{i,j-1} \cdot e^{-\beta/T} \tag{3.9c}$$

with the boundary conditions

$$\begin{aligned} Z^D_{i,0} = Z^P_{i,0} = Z^Q_{i,0} &= 0 \qquad \text{for } i = 0 \ldots L \\ Z^D_{0,j} = Z^P_{0,j} = Z^Q_{0,j} &= 0 \qquad \text{for } j = 0 \ldots M. \end{aligned}$$

Since an alignment may start anywhere and may also include the empty alignment, the full partition function is given by

$$Z = 1 + \sum_{i=1}^{L} \sum_{j=1}^{M} Z^D i, j.$$

Note that the contributions from Z^P and Z^Q are explicitly excluded because they are auxiliary only and contain non-canonical alignments.

Mückstein, Hofacker and Stadler proposed a stochastic backtrace procedure that allows sampling local alignments from the Gibbs-Boltzmann distribution in a direct fashion [MHS02]. The algorithm is illustrated in Appendix A.1.

Instead of expanding the partition function in all score parameters, as proposed by Zhang and Marr [ZM95], we shall perform a numerical *high temperature expansion* to obtain the *density of states* (DOS) of all alignments for a fixed pair of sequences. This is by far less costly than parametric expansion for realistic protein alignment, because there are 210 possible entries in the score matrix plus two parameters for the affine gap costs.

This expansion is feasible when all scoring parameters are integers, which is usually the case, because the scoring matrices are rounded to closest integers. The partition functions in Eq. (3.9) are replaced by polynomials in the expansion parameter $z = e^{1/T}$, e.g. $\hat{Z}^D_{i,j}(z)$ instead of $Z^D_{i,j}$, and all additions and multiplications are operations on polynomials. The full partition function is also a polynomial in z,

$$\hat{Z}(z) = 1 + \sum_{i=1}^{L}\sum_{j=1}^{L} \hat{Z}^D i, j(z) = \sum_n c_n z^n.$$

When re-substituting $z = e^{1/T}$ in $\hat{Z}(z)$ the DOS can easily identified with the coefficient of the expansion

$$\hat{Z}(z) = \sum_n c_n e^{n/T} = \sum_{\mathsf{S}} g(\mathsf{S}) e^{\mathsf{S}/T}.$$

Applications of this methodology are discussed in Sec. 3.6 after brief statements about the so called ”linear-logarithmic“ phase transition.

3.5 The linear-logarithmic phase transition

The correspondence between the physical model of DPRM and sequence alignment was outlined in Sec. 3.3.1. Hence there is a connection between statistical physics and sequence alignment. The study of sequence alignment from that perspective yields interesting results that have improved the optimal choice of parameters of sequence alignment. The *linear-logarithmic phase transition* [WGA87, AW94, BH00] is the most important aspect regarding this issue.

The name stems from the fact that there is a continuous, parameter-driven transition between phases where the average local score (or the ”length“ of a local alignment) grows either linearly or logarithmically with sequence length. The main mechanism of this transition can be seen by looking at the DPRM analog of sequence alignment and consider the local growth of the local alignment score of random sequences. In the linear regime, where gaps are penalized only weakly, the score grows essentially unbounded, because a mismatch can be easily circumvented by gaps with low penalty. In the biological relevant logarithmic phase the growth is essentially bounded by restart condition of the Smith-Waterman algorithm (the first case in Eq. (3.8)).

In the logarithmic regime, when looking at the path graph of local alignments or the dynamic programming matrix $D_{i,j}$ one observes an ensemble of isolated islands of positive scoring segments (see Fig. 3.5(a)).

In contrast, in the linear phase there is essentially one large cluster of the order of the alignment lattice (see Fig. 3.5(b)). This means the transition can be mapped on a

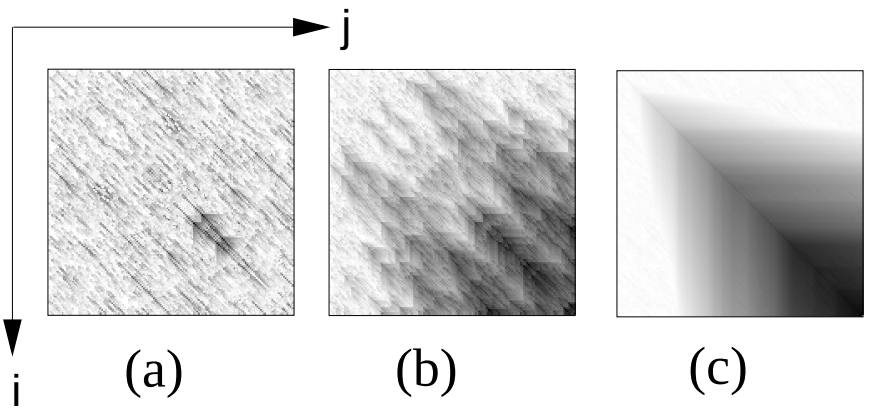


Figure 3.8: The score landscapes (the matrix $\{D_{i,j}, 0\}$) of optimal local alignment of two protein sequences in different phases. Darker greyscale means higher score. The `blosum62` matrix and different gap penalties have been used as parameters.
(a) The landscape of two random sequences in the logarithmic phase ($\alpha = 12, \beta = 1$). This phase is characterized by isolated high score islands.
(b) The linear phase ($\alpha = 6, \beta = 1$) exhibits a single growing cluster.
(c) A pair of non-random related proteins aligned (see Sec. 3.6) with the parameters of the logarithmic phase. The growth rate is comparable to the linear phase.

percolation problem [SAY05][4] (see Sec. 4.6).

From the biological point of view the linear phase is not desirable because large high scoring alignments occur even for random sequences and meaningfull alignments can hardly be distinguished from random ones. On the other side, (non-random) strong relationships show a long "percolating" optimal alignment even in the logarithmic regime (see Fig. 3.5(c) and Sec. 3.6). The main difference to alignments of random sequences in the linear regime is that this optimal alignment is unique, or only weakly degenerate, whereas optimal alignments of random sequences in the linear phase are highly degenerate.

Scaling laws close to the transition line have been studied in a similar fashion as it is commonly done in the field of critical phenomena [HL96, HL98, RO99, DHL00] [5]. An important implication for weakly related sequences has been drawn from the scaling behavior: The optimal parameter set is close to the boundary of the transition on the logarithmic side. In Chapter 4 another aspect of this transition, regarding the alignment score distribution, is discussed.

In the following section some illustrative biological examples of optimal and finite temperature alignments are presented.

[4] A perfect percolation of the alignment-path from (L, M) to $(0, 0)$ occurs rarely due to the geometry of the lattice and the lack of periodic boundary conditions. Instead the path is of the order of the lattice size in the linear phase.

[5] Although the phase transition is essentially parameter driven the extension to finite temperature alignment is also possible [KL00]. Results for this model are discussed in Sec. 4.6

3.6 Thermodynamics of local alignments by biological examples

In order to illustrate the alignment methods I picked out several examples of more or less related proteins. All data presented in this section are based on local alignment using `blosum62` score matrix and affine gap costs with the standard gap cost parameters $\alpha = 12$ and $\beta = 1$. For random subjects the `blosum62` background frequencies had been used.

		subjects			
acc.no.	protein	organism	length	identity	S^{local}
P68873	Hemoglobin subunit beta	Pan troglodytes (Chimpanzee)	147	100%	780
P18989		Procyon lotor (Raccoon)	146	90%	709
P02088		Mus musculus (Mouse)	147	80%	638
P84792		Aythya fuligula (Tufted duck)	147	70%	558
P10060		Sphenodon punctatus (Hatteria)	146	63%	496
Q90486		Danio rerio (Zebrafish)	148	51%	417
O13077		Gadus morhua (Atlantic cod)	147	41%	326
P56692		Dasyatis akajei (Red stingray)	142	33%	200
P02042	Hemoglobin subunit delta	Homo sapiens (Human)	147	93%	727
P02100	Hemoglobin subunit epsilon		147	75%	607
Q8WWM9	Cytoglobin (Histoglobin)		190	28%	173
B4DUI1	cDNA FLJ55163		136	23%	93
	random		147		29

Table 3.1: A list of proteins that are related to the human Hemoglobin subunit beta protein (accession number `P68871`). The accession number of the SwissProt database, the protein name, the species and the length of the protein is shown for each subject. The similarity is measured by number of identical residues and the similarity score.

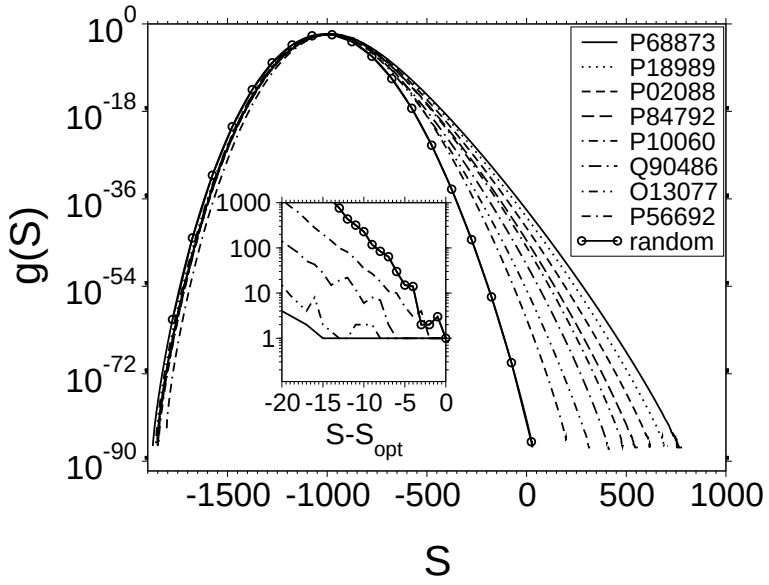


Figure 3.9: The DOS of local sequence alignments of human beta-globin to the beta-globin of eight different species (see Tab. 3.1).

3.6.1 Strong homologs

If we submit the human hemoglobin subunit beta (or beta-globin), a protein that is responsible for oxygen transport, against a current protein database [Uni] we receive a list of homological related proteins. Some of these results are listed in Tab. 3.1. The first block shows the related proteins of other species ordered by the similarity, whereas the second block contains a list of results of human proteins from the globin family. The resulting scores can be compared with the optimal score of the human beta globin aligned against a random sequence. It is no surprise that mammalian (chimpanzee, raccoon and mouse) hemoglobin show highest similarity to the human one, whereas fishes (zebrafish, Atlantic cod and red stingray) only have a weaker related beta-globin. Birds (tufted duck) and Reptiles (hatteria) are intermediate.

For that purpose to look at the "thermodynamics" of the canonical ensemble of all local alignments, I first determined the density of states $g(\mathsf{S})$ of all alignments because the partition function Z_T, free energy F_T and any moment of the score distribution $\langle \mathsf{S}^m \rangle_T$ at any temperature is in principle known. The resulting normalized DOS for the eight alignments in the first block in Tab. 3.1 is shown in Fig. 3.9. The optimum is positive by construction, but most alignment scores of the state space of all local alignments are negative. Of particular interest is the increase of the microcanonical entropy $S(\mathsf{S}) = \log g(\mathsf{S})$, when going towards lower score values (higher excitations in terms of physics). The inset of Fig. 3.9 displays a close up of the unnormalized DOS that has been shifted horizontally such that the ground states match at 0. The entropy increases faster for relative weak homologs and the fastest growth is observed for the random subject. Small entropies imply that there are only few variations of high score alignments, implying a high degree of reliability of the optimal solution.

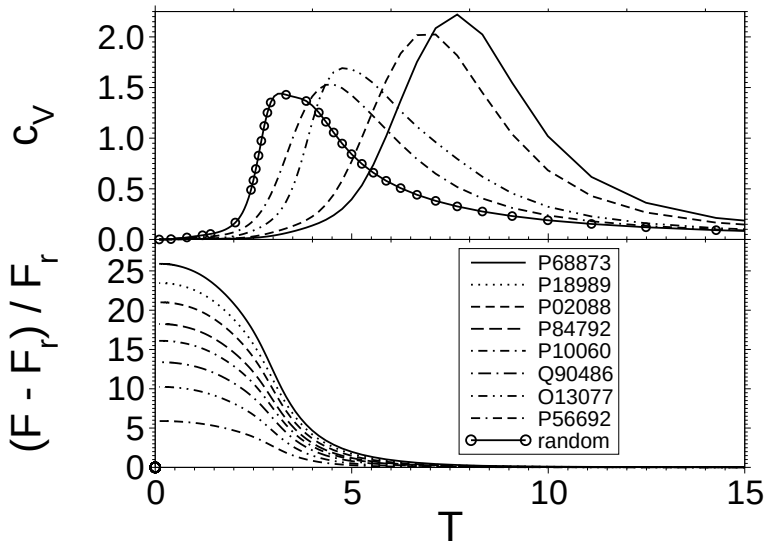


Figure 3.10:
top: Specific heat of the canonical ensemble of alignments with fixed sequences of different degree of homology.
bottom: Relative free energy differences between sequence alignments of fixed sequences against homologs and the same sequence against a random one.

Thermodynamic quantities, such as the specific heat

$$C_V = \frac{\langle \mathsf{S}^2 \rangle_T - \langle \mathsf{S} \rangle_T^2}{L}$$

and the free energy

$$F = T \, \log Z_T = T \, \log \sum_{\mathsf{S}} g(\mathsf{S}) \exp(\mathsf{S}/T)$$

can be computed from the DOS. As mentioned above the temperature controls the balance between the optimal solution and the entropy. The optimal solution emerges below the peak of the specific heat (see top of Fig. 3.10). This can be seen, when looking at the relative free energy difference [KL00]

$$F = \frac{F - F_r}{F_r}$$

between the free energy F of the human beta-globin sequence aligned against homologs and the free energy F_r of the same human sequence aligned against a random one as a function of the temperature. This ratio is increasing with decreasing temperature which means that the zero temperature limit is most powerfull to discriminate this kind of homological relationships against random similarities. Hence the optimal alignment score is the best quantity in this case and nothing is gained when looking in suboptimal alignments.

query			subjects			score
PDBID	organism	L	PDBID	organism	L	S^{local}
1SMV:A	Sesbania mosaic virus	266	1BBT:3	Foot-and-mouth disease virus	220	34
			random		220	27
1BDM:A	Thermus thermophilus	327	1DIH	Escherichia coli	273	49
			random		273	38
1ISZ	Streptomyces olivaceoviridis	436	1FWU	Mus musculus	134	68
			random		134	30
2DOR	Lactococcus lactis	311	1MUC	Pseudomonas putida	373	32
			random		373	27

Table 3.2: Four pairs of proteins that are known to have similar structures. Their optimal sequence similarity score S^{local} is small. BDBID is the ID of the protein data bank [PDB]. Examples 1SMV:A – 1BBT:3 and 1BDM:A – 1DIH have been inspired by Ref. [JLG02] and 1ISZ – 1FWU and 2DOR vs. 1MUC by Ref. [KKK04].

3.6.2 Weak homologs

When looking in the biological literature it is possible to find examples where suboptimal alignments are important and increase accuracy. For instance higher order structures (secondary structure or even the three dimensional conformation) show up higher similarity than the primary sequences, in particular for weak homologs (also called "twilight zone"), because structures are more conservative than sequences during the course of evolution.

Therefore known structures are used in benchmarks of alignment methods [VEA95, JLG02]. Algorithms to compare structures have become more important. One example is the "Combinatorial extension" (CE) algorithm [SB98].

I took four examples from literature [JLG02, KKK04] that are known to show similar structures, see Tab. 3.2. The sequence similarity score is still larger in comparison to optimal sequence alignments of the queries against random sequences, but the discrepancy is not as large as for the examples above, hence this examples should illustrate typical behavior in the twilight zone.

Here, the differences of the DOS of the alignment ensembles of the pair of weakly homolog sequences and the random reference are much smaller than in the example of beta-globin, see Fig. 3.11. The microcanonical entropy functions close to the optimum (inset of Fig. 3.11) are approximately linear with the same slope. Surprisingly the entropy at score values below the optimum score is larger for homologs, which means that there might be many suboptimal alignments with high score.

Also the behavior of the specific heat and relative free-energies (see Fig. 3.12) differ strongly from strong homologs. The specific heat exhibits a richer structure featured with more than one peak. Each peak may indicate the emergence of a set suboptimal local alignments. When looking at the relative free energies, that are defined in the same way as above, one observes that they are not just monotonously decreasing func-

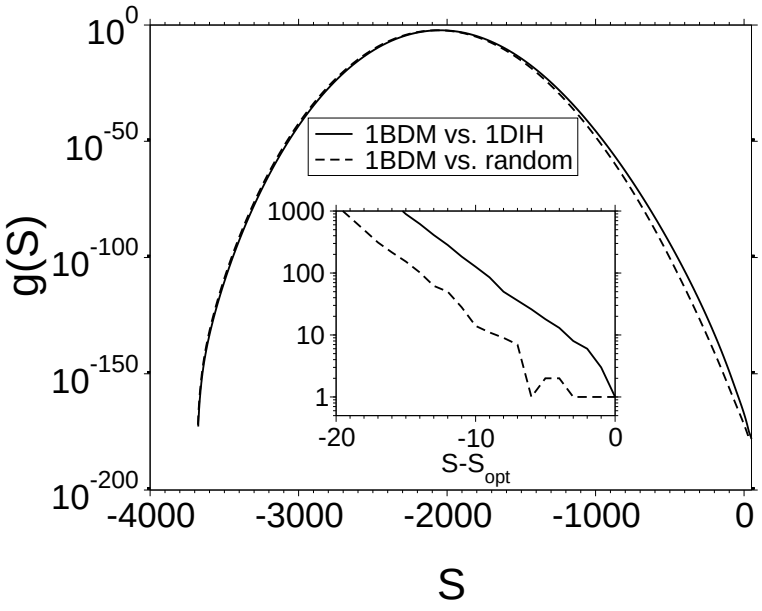


Figure 3.11: The normalized DOS of local sequence alignments of the protein `1BDM:A` and `1DIH` [JLG02] in comparison of the alignments of `1BDM:A` against a random sequence. Inset: The DOS close to the optimum.

tions of the temperature but exhibit a local maximum between $T = 2.5$ and 3.5 for the pairs `1BDM:A`–`1DIH`, `1ISZ`–`1FWU` and `2DOR`–`1MUC` and between $T = 1$ and 2 for `1SMV:A`–`1BBT:3`. In three out of the four examples, the relative free energy at the peaks was even larger than the in the zero temperature limit. Note that this was not observed in the analysis of Kschischo and Lässig [KL00], where artificial homologs where generated by a Markov process and a simple scoring model had been used.

The locii of the peaks are interesting from the statistical mechanics perspective because they can be associated with the linear-logarithmic phase transition (see Sec. 3.5 and Sec. 4.6). Kschischo and Lässig [KL00] have observed that the finite temperature algorithm overestimates the length of a related segment in the sequences, when the temperature is increased.

At first glance the peaks in specific heat and free energy ratio suggest that there are optima above $T = 0$. However some care has to be taken if there is a real biological enhancement at these temperature values. A look at the expectation value of the score $\langle \mathsf{S} \rangle_T$ suggest that alignments above a temperature of approximately $2 - 3$ become meaningless because they have an negative expected score.

For this reason I studied the interesting temperature range from 0 to approximately 1.8 for the pair `1BDM:A`–`1DIH` in more detail. First an ensemble of finite-temperature alignments have been drawn from the Gibbs-Boltzmann distribution using the stochastic backtrace procedure (Algorithm A.1.1 described in Appendix A.1).

Each pair of alignments $\mathcal{A}_i$ and $\mathcal{A}_j$ have been compared quantitatively by a distance measure on the state space

$$d : \chi_{\mathbf{a},\mathbf{b}} \times \chi_{\mathbf{a},\mathbf{b}} \to [0,1] : \mathcal{A}_i, \mathcal{A}_j \mapsto d(\mathcal{A}_i, \mathcal{A}_j) := 1 - \frac{|\mathcal{A}_i \cap \mathcal{A}_j|}{\max\left(|\mathcal{A}_i|, |\mathcal{A}_j|\right)}, \quad (3.10)$$

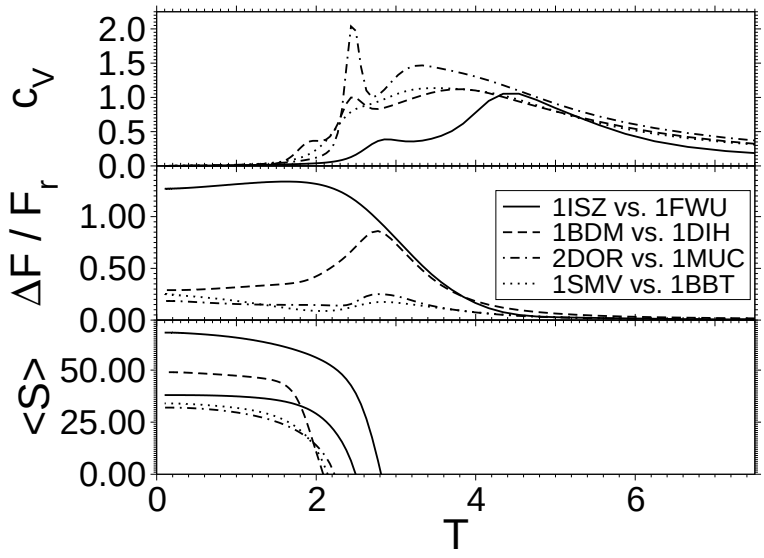


Figure 3.12:
top: Specific heat of the canonical ensemble of alignments of the sequences that are listed in Tab. 3.2. Three of them exhibits more than one peak.
middle: Relative free-energy differences between sequence alignments of the weak homologs and the random reference.
bottom: Thermal expectation value of the score as a function of temperature. Negative values are not shown.

i.e. the number of common pairs in both alignments normalized by the larger one. The distance $d(\mathcal{A}_i, \mathcal{A}_j)$ equals 1 if both alignments have no common base pair and 0 if two identical alignments are compared. A comparison of all pairs of alignments yields to a distance matrix $\Delta_{i,j} := d(\mathcal{A}_i, \mathcal{A}_j)$ which can be used to visualize the structures of the ensemble at different temperatrues. The states have been clustered according to the distances, i.e. similar alignments are grouped in hierarchical clusters. The method is described in Appendix A.3. Fig. 3.13 shows sorted distance matrices, where abscissa and ordinate represent the sampled states and the gray-scale gives the distance between the states. A darker color means smaller distance, hence the diagonal is allways black because two identical alignments are compared. 200 sampled states at $T \approx 2.8$ and $T = 1.7$ for the sequence pair `1BDM:A`–`1DIH` and for `1BDM:A` against a randomly generated sequence are illustrated.

The matrix close to the point, where the expected score decays ($T \approx 1.7$) for the non-random alignment exhibits a rich structure. There are several groups of "similar alignments" that are slight variations of each other either in length or in aligned residues. When the temperature is lowered the emergence of the cluster that contains the ground state, i.e. the optimal alignment, is observed. At $T = 0$ only one cluster persists, if the ground state is not degenerate, which is the case here. In the entropy dominated infinite temperature limit only a light gray area (expect of the diagonal) remains. Hence all alignments have equal probability to occur and are therefore meaningless. It is remarkable that close to the position of the maximum of the relative free energy ratio between the homolog pair and the random one (see Fig. 3.12) most infor-

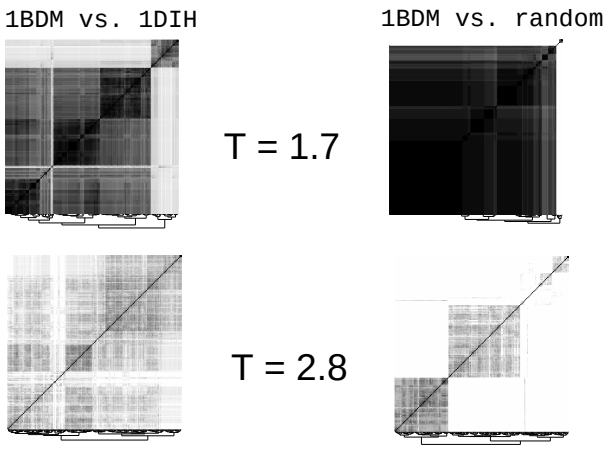


Figure 3.13: Distance matrices at the temperatures after applying the clustering (see Appendix A.3) with the distance measure Eq. (3.10).

mation is already destroyed, which can be seen by the poor structure of the distance matrix at that temperature, $T \approx 2.8$. In comparison with the random alignment, one learns that in the interesting temperature region, $T \approx 1.7$, the ensemble of the homologous pair exhibits a rich structure. In contrast, the ground-state dominates the ensemble of the random pair. This viewpoint might give more insight of the reliability of finite temperature alignment.

Next a comparison of structural, the optimal and the finite-temperature alignment is provided by looking at the paths graphs of these alignments. The solid line in Fig. 3.14 indicates the structural alignment obtained by the CE-algorithm [SB98] and is assumed to the "standard of truth" [JLG02, VEA95]. The dashed line shows one alignment taken from the finite temperature alignment ensemble.

The reliability of the optimal alignment is bad because it is essentially too short. The finite temperature algorithm yields better results, as the alignments become longer and predict matched segments better. However, it also fails to predict the last segments of the structural alignment, which deviate from diagonal. This effect was also observed in Ref. [JLG02], where an hybrid algorithm has been employed. This algorithm combines an iterative algorithm [SS91] and a parametric algorithm that uses several score parameters at the same time [JPG98, WEL92]. A recent study [KKK04] shows how this drawback can be circumvented by introducing periodic boundary conditions. A more detailed study on this issue has not be done here, as it is beyond the main scope of this thesis.

Instead, the statistical significance of local alignment in different variants that have been discussed in this chapter are subject of the next chapter. A discussion of the examples of this section is provided in Sec. 4.4 and Sec. 4.6.

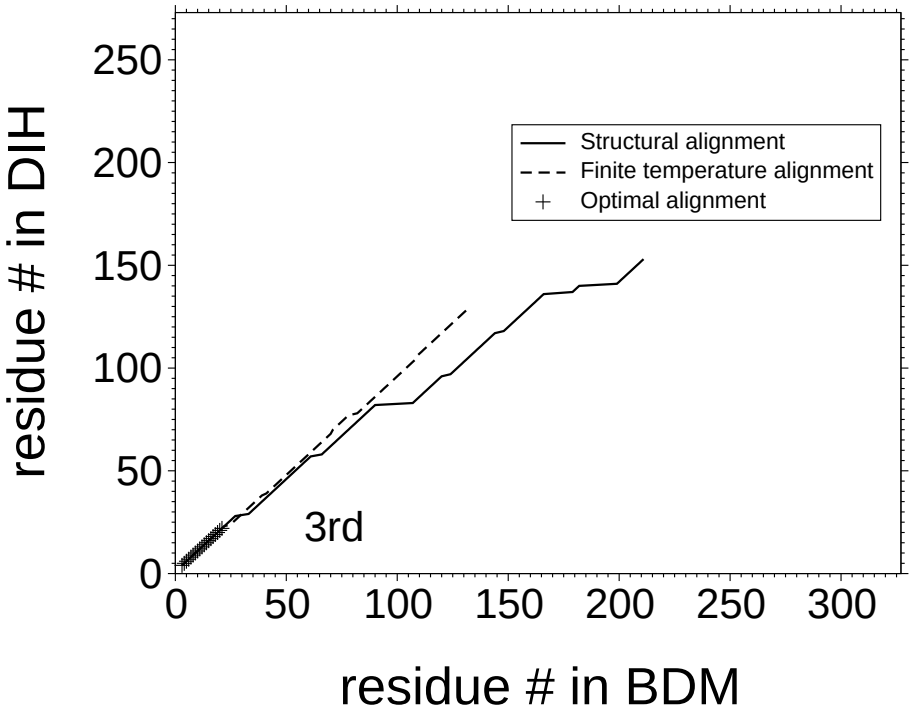


Figure 3.14: Comparison of the alignment path of the structural alignment with the optimal alignment and a finite temperature alignment at $T = 1.7$. Note that the optimal alignment covers the structural alignment and the finite temperature alignment only partly.

Chapter 4

Statistics of local sequence alignment

Pairwise alignment algorithms find optimum alignment scores $\mathsf{S}_0^{\text{local}}(\mathbf{a}, \mathbf{b})$ and associated alignments of two sequences $\mathbf{a}$ and $\mathbf{b}$ for a given scoring system. Different variants of local alignment algorithms were introduced and illustrated in the previous chapter. We have also seen that the optimal score of two sequences is always positive even for random sequences. This leads to problems when one wish to distinguish between *true* and *false positives*. Positives are hits (results from a database query) that are reported as homologe to the input sequence by the search program. True positives are caused by a true biological relationship and, accordingly, a false positive is reported as a hit though the score occurred by pure chance. Similarly, a false negative is biological relationship which have been missed by the search program.

The task to better distinguish between these cases requires an assessment of the statistical significance of an observed score. The most frequently used approach for sequence alignment relies on classical test theory (see [PJ01] for a introduction). In this framework the statistical significance is quantified by the *p-value* for a given score. This means under a random sequence model, one wants to know the probability for the occurrence of at least one hit with a score greater than or equal to some given threshold value b, i.e. $\mathrm{Prob}(S \geq b)$. Note that this definition is different to many other statistical tests, where a large p-value means high significance. Here we have to interpret a small p-value as much evidence for a true positive. Often E-values are used instead. They describe the number of expected hits with a score greater than or equal to some threshold value.

One possible access to the statistical significance can be achieved under the null model of random sequences. In such a model, the pair of sequences is a random variable, implying that the optimal alignment score, as a function of a random variable, is also random. The probability of the occurrence of the score s under this model, $P(s) = \mathrm{Prob}(S = s)$, provides the basis for estimates of the p-values.

Analytic expressions for the probability distribution of S are only known asymptotically in the case of gapless alignments of long sequences, where an *extreme-value distribution* (also called *Gumbel distribution*) is predicted (see Sec. 4.1 below).

Because significant hits usually exhibit a high score, the rare-event tail, where the p-values are very small, are of particular interest. By viewing the alignment problem as a physical system, we are able to adopt Markov-chain Monte Carlo methods to obtain

the density of states from statistical physics (see Chapter 2) and address the problem numerically.

The results of the asymptotic theory including some recent results and approximations are outlined in Sec. 4.1. After that a few comments on the Monte Carlo methods are made in Sec. 4.2. In the result sections Sec. 4.3-Sec. 4.6, we will successively release the assumptions of the theory and study the score distribution for more realistic models by numerical simulations.

4.1 Karlin-Altschul-Dembo theory and beyond

In the early 1990s, Altschul, Dembo, Karlin and Zeitouni developed a theory (Karlin-Altschul-Dembo statistics [KA90, KD92, DKZ94]) that describes the infinite sequence limit of the relevant probability $\mathrm{Prob}(S \geq b)$ or the probability density function $P(s) = \mathrm{Prob}[\mathsf{S}_0^{\text{local}}(\mathbf{A}, \mathbf{B}) = s]$ [1], where $\mathbf{A}$ and $\mathbf{B}$ are random sequences.

In order to formulate this more precisely, we assume that the two random sequences are described by the distributions p^{query} and p^{subject}. With this notation we may define $P(s)$ as

$$\begin{aligned} P(s) &= \mathrm{Prob}[\mathsf{S}_0^{\text{local}}(\mathbf{A}, \mathbf{B}) = s] \\ &= \sum_{\mathbf{a} \in \Sigma^L} \sum_{\mathbf{b} \in \Sigma^M} \delta_{s, \mathsf{S}_0^{\text{local}}(\mathbf{a}, \mathbf{b})} \cdot p^{\text{query}}(\mathbf{a}) \cdot p^{\text{subject}}(\mathbf{b}). \end{aligned} \tag{4.1}$$

Since the optimal score can be seen as the ground state of a disordered system, the distribution $P(s)$ is equivalent to the ground-state-energy distribution. This kind of problem is an active research field in statistical physics, because from the shape of this distribution one can learn much about the microscopical interaction and vice versa [Pal03, KKL$^+$05, KKH06, MG06, MG08]. Of particular interest is the question whether the ground-state energy distribution is skewed like the extreme-value [Gum58] or the Tracy-Widom distribution [TW96] or symmetric like the Gaussian one. The latter one indicates an extensive character of the ground-state energy. Typically skewed ground-state-energy distribution are observed in disordered models with long-range interactions, such as the Sherrington-Kirkpatrick model [KKL$^+$05].

Instead of the state space of alignments, one may also see the state space of random sequences as a classical physical ensemble and interpret the optimal score as (negative) energy. The score distribution reads as the density of states of the system (up to normalization) and its logarithm $\log P(s)$ as the microcanonical entropy function, or rate function in the language of large deviation theory. One aim of this chapter is to discuss the shape of these functions for different scoring and sequence models.

Given the amino acid background frequencies (see Sec. 3.2.1) for the query $f^{\text{query}} : \Sigma \to [0, 1]$ and the subject $f^{\text{subject}} : \Sigma \to [0, 1]$ and the score matrix $\sigma : \Sigma \times \Sigma \to \mathbb{R}$ (see Sec. 3.1), the statistical theory requires that

(i) the letters in the random sequences are independent and identically distributed (i.i.d.), i.e.

$$p^{\text{query}}(\mathbf{a}) = \prod_{i=1}^{L} f_{a_i}^{\text{query}} \quad \text{and} \quad p^{\text{subject}}(\mathbf{b}) = \prod_{i=1}^{M} f_{b_i}^{\text{subject}}, \tag{4.2}$$

[1] In the asymptotic theory the score can be seen as a continuous variable.

(ii) $\mathsf{S}_0^{\text{local}}$ is the objective function of ungapped local alignment, where $\alpha = \beta = \infty$,

(iii) the expected score is negative, i.e. $\sum_{a,b} f_a^{\text{query}} f_b^{\text{subject}} \sigma(a,b) < 0$ and

(iv) a positive score must be possible, $f_a^{\text{query}} f_b^{\text{subject}} \sigma(a,b) > 0$ for at least one pair $a, b \in \Sigma$.

Condition (iii) states that the theory is only valid in the logarithmic regime (see Sec. 3.5 and Sec. 4.6 below) and is by construction automatically fullfilled for BLOSUM and PAM matrices, when gaps are not allowed. If condition (iv) was not fullfilled, the optimal local alignment would be the empty alignment for all input sequences.

Under conditions (i)-(iv) the probability $P(s)$ approaches a Gumbel distribution [Gum58]

$$P_{\text{Gumbel}}(s) = \text{Prob}(S = s) = \lambda K L_{\mathtt{Q}} L_{\mathtt{S}} \exp\left[-\lambda s - K L_{\mathtt{Q}} L_{\mathtt{S}} e^{-\lambda s}\right]. \tag{4.3}$$

as both sequence lengths $L_{\mathtt{Q}}$ and $L_{\mathtt{S}}$ tend to infinity with the same rate. The parameters K and λ are determined by the score matrix σ and the background frequencies f^{query} and f^{subject} by a transcendental equation[KA90]. Even though the proof of this theorem is non-trivial, some of the ideas can be understood with intuition. It is done in two steps. First one notices that the dynamic programming matrix of the Smith-Waterman algorithm $D_{i,j}$ that is filled via Eq. (3.8) exhibits many zero valued entries due to condition (iii). This yields independent so called high-scoring segments as already illustrated in Fig. 3.5(a). By renewal theory one can show that the local score (the maximum of those segments) is distributed according to the Poisson distribution. After filling the dynamic programming matrix $D_{i,j}$, the optimal score is given by the maximum over all matrix entries. Because the score of the segments are virtually independent, one may apply the extremal types theorem [Gum58]. This theorem states that the maximum of i.i.d. random numbers converges to one out of three universal distributions. The limiting distribution depends on the original distribution of the random numbers over which the maximum was taken. If this distribution decays faster than a power law, which is the case here, the limiting distribution is the Gumbel distribution given by Eq. (4.3).

The biological relevant case of gapped alignment and finite sequences is not governed by the analytical theory. Numerical studies [AG96] and a Poisson approximation [Wat94] suggest that, at least in the high probability region (see below), the Gumbel form is still a valid description of $P(s)$. However the parameters λ and K cannot be predicted directly. One practical approach to this problem is to use pre-computed parameters based on numerical simulations [AG96, ABOH01] for various widely used parameters.

According to Eq. (4.3), the form of the Gumbel distribution is independent of the sequence length in the limit $L_{\mathtt{Q}} = L_{\mathtt{S}} \to \infty$. In practice this is not the case due to edge effects [RO99, ABOH01] and database applications use adjusted λ's, but the distribution is still assumed to be of Gumbel form. Since this effect vanishes in the limit of infinite sequences, the tail of Eq. (4.3) can be understood as an upper bound for finite sequences.

Another consequence of finite system sizes becomes only visible in the rare-event tail of the score distribution. This region is characterized by high scores and long alignments. The length of these alignments are of the order of the sequence length and hence condition (iii) is not fullfilled any more. Hartmann studied this problem by parallell-tempering Monte Carlo simulations (see Sec. 2.4) and reweighting techniques [Har02]. He found strong deviations of the score distribution from the Gumbel form in

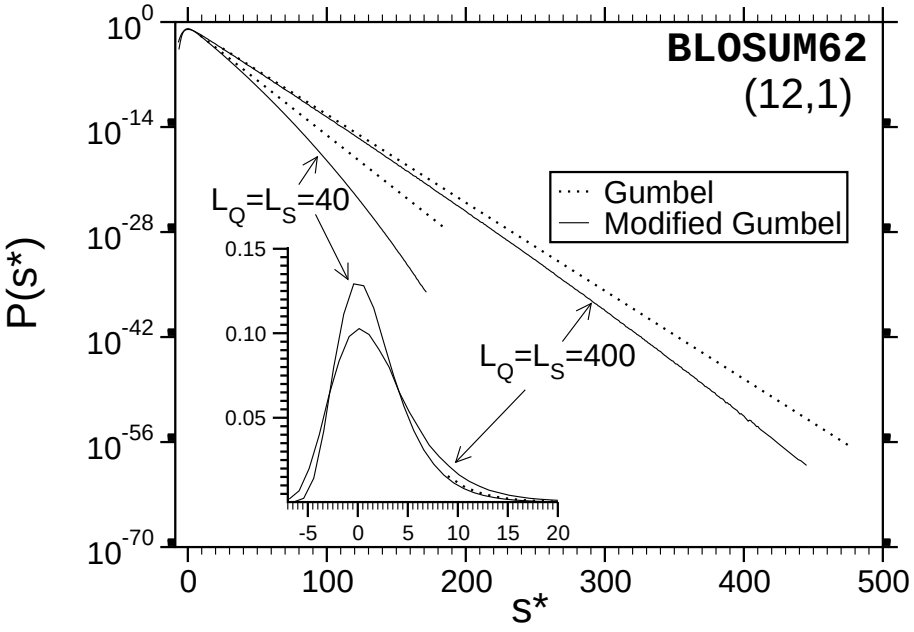


Figure 4.1: Probability distribution P(s) for gapped sequence alignment using BLOSUM62 matrices and affine gap costs with $\alpha = 12, \beta = 1$ for two sequences lengths $L_{\mathtt{Q}} = L_{\mathtt{S}} = 40$ and $L_{\mathtt{Q}} = L_{\mathtt{S}} = 400$ [Har02]. Strong deviations from the Gumbel distribution become visible in the tail. The dotted lines show the original Gumbel distribution, when fitted to the region of high probability. The inset shows the same data with linear ordinate.

the rare-event tail (see Fig. 4.1). Where the entropy function of the Gumbel distribution exhibit a straight line, the accurate entropy function is rather parabolic. This result was obtained heuristically by a least square fit of the empirical data to a *modified Gumbel distribution* in the form

$$\begin{aligned} P(s) &= P_{\text{Gumbel}}(s) \cdot \exp\left[-\lambda_2 \left(s - s_0\right)^2\right] \\ &= \lambda \exp\left[-\lambda \left(s - s_0\right) - \lambda_2 \left(s - s_0\right)^2 - e^{-\lambda(s - s_0)}\right], \end{aligned} \tag{4.4}$$

with $s_0 = \log(K L_{\mathtt{S}} L_{\mathtt{Q}})/\lambda$. We use normalized scores $s^* = s - s_0$ by subtracting the position of the maximum s_0 of the probability distribution throughout. Note that we would have to use a different normalization constant here, but since the correction dominates the tail of the distribution, the normalization constant is numerically indistinguishable from λ.

Results as in Ref. [Har02] are only useful if one obtains the distribution for a large range of parameter values which are commonly used in bioinformatics. It is one purpose of this thesis to study the score distribution for other relevant cases.

In the following section, the Monte Carlo approaches are made explicit. Their general formulation was made in Chapter 2, in particular the Metropolis algorithm (Sec. 2.2) in combination with importance sampling and reweighting (Sec. 2.6). For the first sub-project presented in Sec. 4.3, I used parallel tempering and the technique of

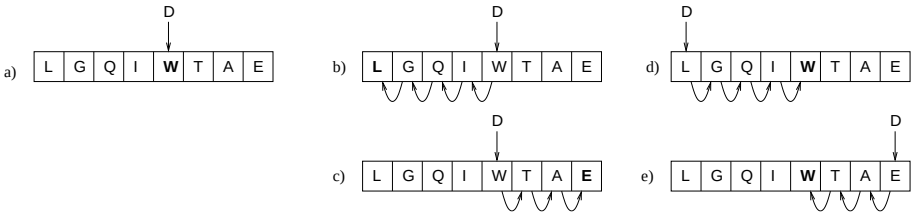


Figure 4.2: (i) substitution, (ii) insertion with left shift, (iii) insertion with right shift,(iv) deletion with right shift and (v) deletion with left shift.

reweighting mixtures (see Sec. 2.4 and Sec. 2.6.1). Because the reweighting procedure is quite complicated for practical terms, I used the generalized ensemble methods (see Sec. 2.7) in all further studies on this problem.

4.2 Sampling of rare events in the sequence space

In order to sample pairs of sequences from the distributions p^{query} and p^{subject} with Markov-chain Monte Carlo methods we need to define a local neighborhood $\mathcal{N}(\mathbf{a})$ of a sequence $\mathbf{a} \in \Sigma^L$. This construction is explained in the following.

First, one of the two sequences are chosen at random. Second, a random letter is drawn from the alphabet Σ according to the frequencies f_a and a random position k is chosen. The sequence is then modified by one of the following moves (see also Fig. 4.2) [WHRH]

(i) substitution at position k,

(ii) insertion at position k with left shift and removal of the first letter

(iii) insertion at position k with right shift and removal of the last letter,

(iv) deletion at position k with left shift and insertion at the last position,

(v) deletion at position k with right shift and insertion at the first position.

For the studies in the following section, I only used proposal (i). After finishing that work I realized that the performance can be enhanced with the additional moves (ii)-(v). Hence all other simulations in Sec. 4.4-Sec. 4.6, including those that are presented in Chapter 6, employed all five proposals.

It is easy to show that this choice guarantees detailed balance (see Sec. 2.2) when $f_a^{\text{unif}} = \frac{1}{|\Sigma|}$ for all $a \in \Sigma$. This proposal is accepted according to the Metropolis rule Eq. (2.4)

$$\alpha_{(\mathbf{a},\mathbf{b}),(\mathbf{a}^*,\mathbf{b}^*)} = \min\left\{1, \frac{w(\mathsf{S}_0^{\text{local}}(\mathbf{a}^*,\mathbf{b}^*)) \cdot p^{\text{query}}(\mathbf{a}^*) \cdot p^{\text{subject}}(\mathbf{b}^*)}{w(\mathsf{S}_0^{\text{local}}(\mathbf{a},\mathbf{b})) \cdot p^{\text{query}}(\mathbf{a}) \cdot p^{\text{subject}}(\mathbf{b})}\right\} \tag{4.5}$$

The implementation dependent weights w are specified below. For the case of i.i.d. sequences, one may use Hastings' generalization of the Metropolis algorithm to construct more efficient acceptance rates [Has70]. Instead of sampling the new letters from an uniform distribution $f_a^{\text{unif}} = \frac{1}{|\Sigma|}$, one may draw them directly from the desired frequencies f_a. Consequently the proposal densities

$$Q_{\mathbf{a},\mathbf{a}^*} \equiv \text{Prob}(\mathbf{a}^*|\mathbf{a})$$

are not symmetric any more. By the construction of $\mathcal{N}(\mathbf{a})$ we may write

$$\frac{Q_{\mathbf{a}^*,\mathbf{a}}}{Q_{\mathbf{a},\mathbf{a}^*}} = \frac{1}{Q_{\mathbf{a},\mathbf{a}^*}} \cdot \mathrm{Prob}(\mathbf{a}^*|\mathbf{a}) \cdot \frac{p(\mathbf{a})}{p(\mathbf{a}^*)} = \frac{Q_{\mathbf{a},\mathbf{a}^*}}{Q_{\mathbf{a},\mathbf{a}^*}} \cdot \frac{p(\mathbf{a})}{p(\mathbf{a}^*)} = \frac{p(\mathbf{a})}{p(\mathbf{a}^*)}$$

and thanks to Hastings generalization Eq. (2.2) and due to the factorization Eq. (4.2) one may compensate such asymmetric proposals by

$$\begin{aligned} \alpha_{(\mathbf{a},\mathbf{b}),(\mathbf{a}^*,\mathbf{b}^*)} &= \min\left\{1, \frac{w(\mathsf{S}_0^{\text{local}}(\mathbf{a}^*,\mathbf{b}^*)) \cdot p^{\text{query}}(\mathbf{a}^*) \cdot p^{\text{subject}}(\mathbf{b}^*) \cdot Q_{\mathbf{a}^*,\mathbf{a}} \cdot Q_{\mathbf{b}^*,\mathbf{b}}}{w(\mathsf{S}_0^{\text{local}}(\mathbf{a},\mathbf{b})) \cdot p^{\text{query}}(\mathbf{a}) \cdot p^{\text{subject}}(\mathbf{b}) \cdot Q_{\mathbf{a},\mathbf{a}^*} \cdot Q_{\mathbf{b},\mathbf{b}^*}}\right\} \\ &= \min\left\{1, \frac{w(\mathsf{S}_0^{\text{local}}(\mathbf{a}^*,\mathbf{b}^*))}{w(\mathsf{S}_0^{\text{local}}(\mathbf{a},\mathbf{b}))}\right\} \end{aligned} \tag{4.6}$$

In cases, where I considered i.i.d. sequences, I used the acceptance criterion Eq. (4.6) throughout. This also applies the the investigation that is discussed in Chapter 6.

At least approximately, the distribution of local alignment follows a Gumbel distribution, which exhibits an exponential behavior in the tail. Therefore an obvious choice for the biased weights w is an exponential distribution

$$w_\Theta(s) \propto \exp\left[s/\Theta\right]. \tag{4.7}$$

Since we may consider the sequence space as physical system this refers to sampling from the Gibbs-Boltzmann distribution

$$P_\Theta(\mathbf{a},\mathbf{b}) = \frac{1}{Z_\Theta} \exp\left[\mathsf{S}_0^{\text{local}}(\mathbf{a},\mathbf{b})/\Theta\right],$$

where Z_Θ is the (usually unknown) partition function. The parameter Θ corresponds to the temperature and the optimal score $\mathsf{S}_0^{\text{local}}$ to the energy function. Note that this perspective is different to the canonical ensemble of sub-optimal alignments introduced in Sec. 3.4, where the state space was defined as the set of alignments rather than sequences.

Having defined the local neighborhood $\mathcal{N}(\mathbf{a})$, it is easy to implement the Metropolis algorithm in the canonical ensemble. In order to accelerate equilibration, I used the parallel tempering algorithm (see Sec. 2.4).

Equilibration was detected by a criterion that checks whether distinct Monte Carlo chains converge to the same score independent from the starting configuration. This method is possible because we are able to generate pairs of sequences with high scores (low energies) directly by using the second sequence as a one-to-one copy of the first one. On the other side, we may also start with completely independent random sequences yielding a low score (high energy). The chain is considered to be in equilibrium when both runs converge to the same score value (within fluctuations). This is usually detected by averaging the chains over different independent runs starting from both extremes. Fig. 4.3 illustrates this test for a very simple 4 letter toy model. In order to determine correct error bars from correlated data, I only used a thinning interval that had been determined by the autocorrelation time (see Sec. 2.5.2).

Having obtained the chains $\{(\mathbf{a}_{ki},\mathbf{b}_{ki})\}$ for the m temperatures $(\Theta_1,\ldots,\Theta_m)$ with n_k samples each ($k = 1,\ldots,m$ and $i = 1,\ldots,n_k$), the score distributions are obtained

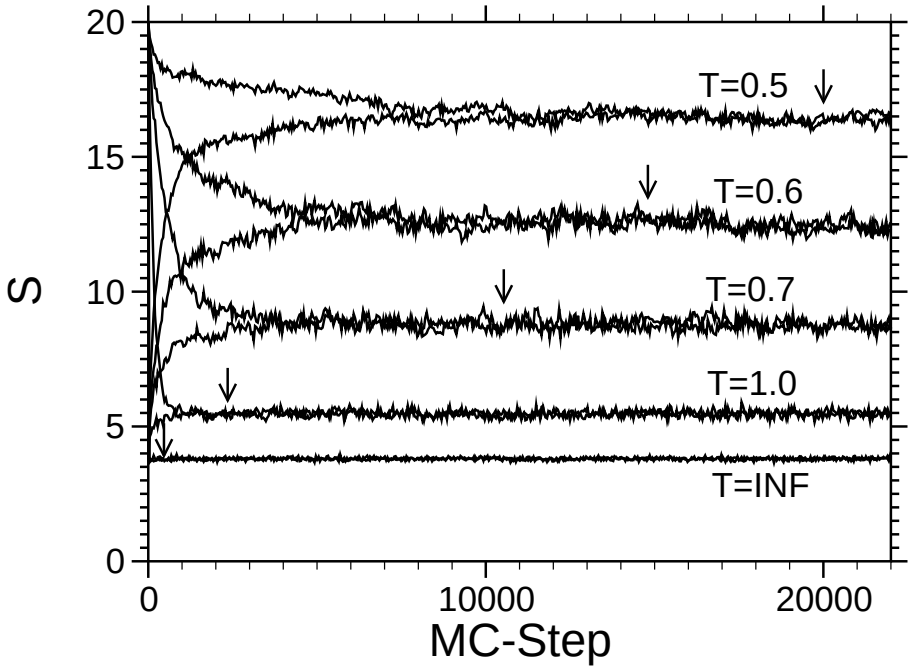


Figure 4.3: Illustration of the equilibration criterion for a 4-letter system ($L_{\mathsf{Q}} = L_{\mathsf{S}} = 20$) with temperatures $T = 0.5, 0.6, 0.7, 1.0, \infty$ Equilibrium is reached after $20000, 15000, 10000, 1000, 100$ steps (indicated by arrows) respectively. $S(t)$ is averaged over 250 independent runs in this example.

by the reweighting procedure described in Sec. 2.6.1, particularly Eq. (2.19)

$$\begin{aligned} P(s) = \mathrm{Prob}(S = s) &\approx \frac{1}{n} \sum_{k=1}^{m} \sum_{i=1}^{n_k} \delta_{\mathsf{S}_0^{\text{local}}(\mathbf{a}_{ki}, \mathbf{b}_{ki}), s} \cdot \frac{P_{(\Theta=\infty)}(\mathbf{a}_{ki}, \mathbf{b}_{ki})}{q_{\text{mix}}(\mathbf{a}_{ki}, \mathbf{b}_{ki})} \\ &= \frac{1}{n} \sum_{k=1}^{m} \sum_{i=1}^{n_k} \frac{\delta_{\mathsf{S}_0^{\text{local}}(\mathbf{a}_{ki}, \mathbf{b}_{ki}), s}}{q_{\text{mix}}(\mathbf{a}_{ki}, \mathbf{b}_{ki})}. \end{aligned}$$

The data analysis for the Wang-Landau / generalized ensemble method turns out to be much simpler because there is no need to determine q_{mix}. The reweighting for this method is performed by the usual importance reweighting formula Eq. (2.12)

$$P(s) = \mathrm{Prob}(S = s) \approx \frac{1}{z} \sum_{i=1}^{n} \frac{\delta_{\mathsf{S}_0^{\text{local}}(\mathbf{a}_i, \mathbf{b}_i), s}}{w(\mathsf{S}_0^{\text{local}}(\mathbf{a}_i, \mathbf{b}_i))},$$

with $z = \sum_{\mathsf{S}} w(\mathsf{S})$. The generalized ensemble weights w are obtained by the Wang-Landau iteration Algorithm 2.7.1. A second advantage over parallel tempering is that a generalization to bivariate distributions is straightforward. We shall use this property later on in Sec. 4.5.

4.3 Statistics of two i.i.d. sequences

This section deals with the results from the application of the parallel-tempering method to biological relevant systems: local sequence alignment of protein sequences using `BLOSUM62` [HH92] (see Sec. 3.2.2) and `PAM250` [DSO78, SD78] (see Sec. 3.2.1) matrices. In contrast to the results of the next section, I used amino acid background frequencies by Robinson and Robinson [RR91]. I considered different affine gap cost with $10 \leq \alpha \leq 16$, $\beta = 1$ for the `BLOSUM62` matrix and $11 \leq \alpha \leq 17$, $\beta = 3$ when using the `PAM250` matrix, as well as infinite gap costs. Furthermore different sequence lengths between $M = L = 40$ and $M = L = 400$, in detail $L = 40, 60, 80, 100, 150, 200, 250, 300, 350, 400$, were considered. For one case, also sequence lengths up to $L = 800$ were used.

Only temperatures where equilibration was guaranteed within a reasonable computation time were used for the calculation of $P(s)$. This means that we cannot resolve the score probability distribution over its full support, i.e. the ground state, the pair with the maximal possible score, cannot be reached for all sequence lengths. But the range of temperatures is large enough to evaluate the distributions down to values $P(s) \sim 10^{-60}$. The temperature sets I used in the parallel-tempering technique varied between $\{2.00, 2.25, 2.50, 3.00, 5.00, 7.00, \infty\}$ ($L = 40$) and $\{3.25, 3.50, 4.00, 5.00, 7.00, \infty\}$ ($L = 400$) for `BLOSUM62` matrices and between $\{2.75, 3.00, 3.25, 4.00, 5.00, 7.00, \infty\}$ and $\{4.00, 4.25, 4.50, 5.00, 8.00, \infty\}$ for the `PAM250` matrices. For each run, I performed 8×10^5 Monte Carlo steps. The resulting probabilities were obtained from 10 ($L = 400$) up to 100 ($L = 40$) independent runs. As emphasized in Sec. 2.6, it is required that all distribution overlap sufficiently. The typical overlap matrix that serves as a quantitative measure for this condition (defined in Eq. (2.14)) was

$$(w_{ij}) = \begin{pmatrix} 1 & 0.6850 & 0.5017 & 0.2717 & 0.0480 & 0.0015 \\ 0.6850 & 1 & 0.7857 & 0.4624 & 0.0984 & 0.0034 \\ 0.5017 & 0.7857 & 1 & 0.6409 & 0.1607 & 0.0117 \\ 0.2717 & 0.4624 & 0.6409 & 1 & 0.3587 & 0.0549 \\ 0.0480 & 0.0984 & 0.1607 & 0.3587 & 1 & 0.3777 \\ 0.0015 & 0.0034 & 0.0117 & 0.3777 & 0.3777 & 1 \end{pmatrix}.$$

for $L = 400$ and `BLOSUM62`. Thus the overlap graph is connected sufficiently. For $L = 40$ the relative errors of the normalization constants varied between 10^{-4} (highest temperature) and 0.4 (lowest temperature) and similarly for $L = 400$.

The main result is that most of the distributions deviate strongly from the Gumbel form, which is indicated in Fig. 4.1 and Fig. 4.4 by dotted lines. One observes that the discrepancy seems to be stronger for shorter sequences. Also, the case without gaps (Fig. 4.4) deviates, at least for $L_{\mathtt{S}} = L_{\mathtt{Q}} = 400$, only weakly from the Gumbel distribution. This might be expected due to the previous analytical work [KA90, DKZ94]. Qualitatively the behavior of the `PAM250`-matrices is the same and therefore the plots are not shown here. A quantitative analysis of all results will be given below. Empirically we find that the resulting distributions can be described by the modified Gumbel distribution given in Eq. (4.4) [Har01]. I modeled the data by a weighted least square fit using the program `gnuplot` [GNU]. The resulting fit parameters are shown in Tab. C.1 and Tab. C.2 in the appendix.

Note that only for not too small sequences χ^2_* is in the order of one. This means that Eq. (4.4) describes the data better for longer sequences. However biological relevant sequence lengths ($L > 200$) sit in the range were the fit works fine. Moreover, the

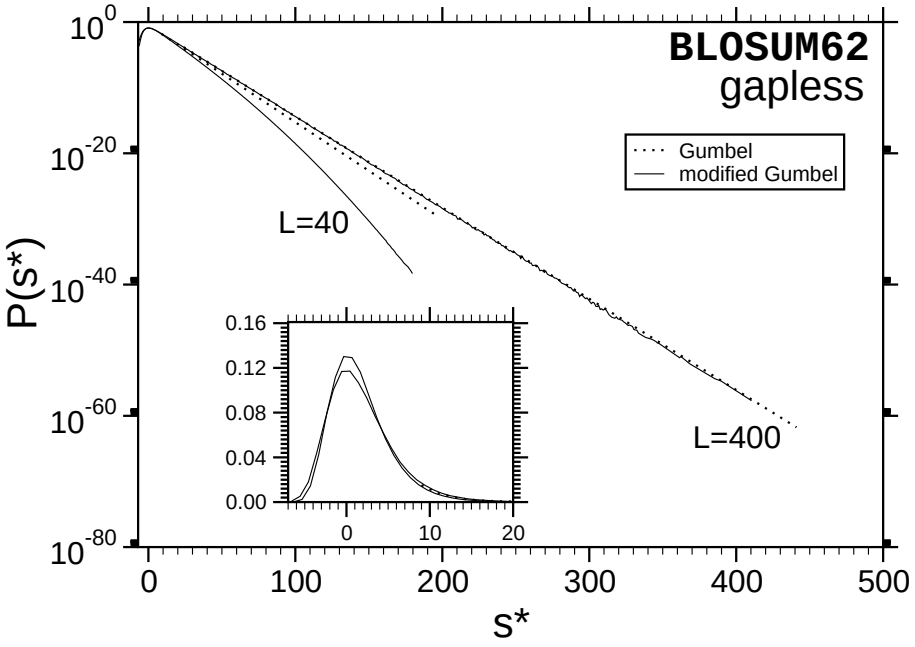


Figure 4.4: Probability distribution P(s) for ungapped sequence alignment using BLOSUM62-matrices. Deviations form the Gumbel-distribution can only be observed for short sequences ($L < 250$). The inset shows the same data with linear ordinate.

results for shorter sequences are still several orders of magnitude below the asymptotic Gumbel result, which yield a χ^2_* value of about 10^4 for the $L = 40$ system.

Next, we study the scaling behavior of the correction parameter λ_2, i.e. the curvature of the entropy function. Since the distributions seem to approach the Gumbel distribution with increasing sequence length, as can be seen in Fig. 4.1 and Fig. 4.4, we expect that λ_2 decreases for $L \to \infty$. Furthermore, when looking at Fig. 4.5, where $P(s)$ is shown for one sequence length $L_\mathtt{S} = L_\mathtt{Q} = 250$ but for different gap-opening costs α, we expect a weak dependence of λ_2 on α. In order to provide more quantitative evidence, we fitted all distributions to Eq. (4.4) and compared the resulting fit parameters.

Parameter	BLOSUM62 $\alpha = 10, \beta = 1$	BLOSUM62 $\alpha = 12, \beta = 1$
a	0.00928 ± 0.0001	0.0309 ± 0.01
b	0.643 ± 0.027	0.971 ± 0.08
$10^{-5}\,\lambda_2^*$	4.9 ± 1.2	3.2 ± 2.0
Parameter	PAM250 $\alpha = 11, \beta = 3$	PAM250 $\alpha = 13, \beta = 3$
a	0.0049 ± 0.0008	0.0053 ± 0.0005
b	0.575 ± 0.046	0.591 ± 0.023
$10^{-5}\,\lambda_2^*$	3.015 ± 2.0	6.1 ± 1.1

Table 4.1: . Fitting parameters of the scaling relation Eq. (4.8).

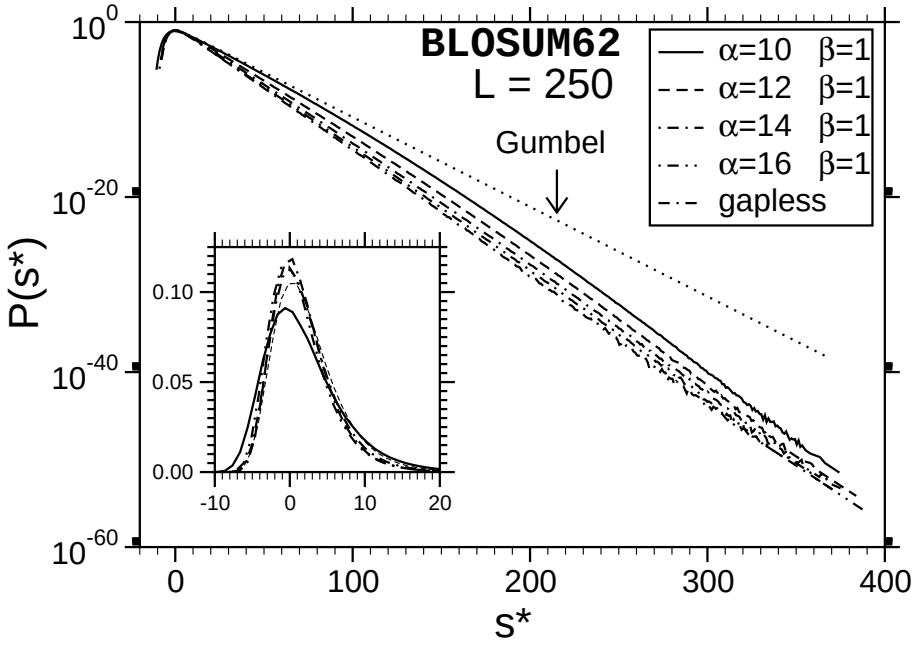


Figure 4.5: Probability distributions $P(s)$ comparing different gap costs. The dotted line denotes the distribution without Gaussian correction ($\lambda_2 = 0$). Deviations from the Gumbel distribution become stronger for small gap costs. The inset shows the same data with linear ordinate.

In the gapless case, no deviations from Gumbel could be detected for sequence lengths $L > 200$. For the other cases, the dependence of the scaling behavior λ_2 on the sequence length is plotted in Fig. 4.6(a) and Fig. 4.6(b). `BLOSUM62` and `PAM250` behaves qualitatively the same. λ_2 seems to decay with a power law

$$\lambda_2 (L) = a\, L^{-b} + \lambda_2^\star \tag{4.8}$$

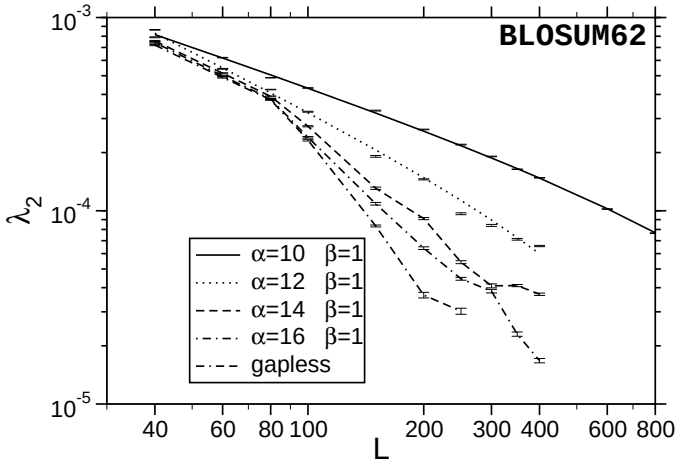
for the smallest gap costs and faster than a power law for larger gap costs.

By fitting the limiting cases (two smallest gap costs) to this function an upper bound of the decay could be estimated. The results are summarized in Tab. 4.1. Note that these arguments are purely heuristically attempts to look at the scaling behavior and its upper bound. It is hard to decide whether the extrapolation is valid for $L_\mathtt{S} = L_\mathtt{Q} \to \infty$. However, an important range of biological interesting sequence lengths are governed with this scaling analysis.

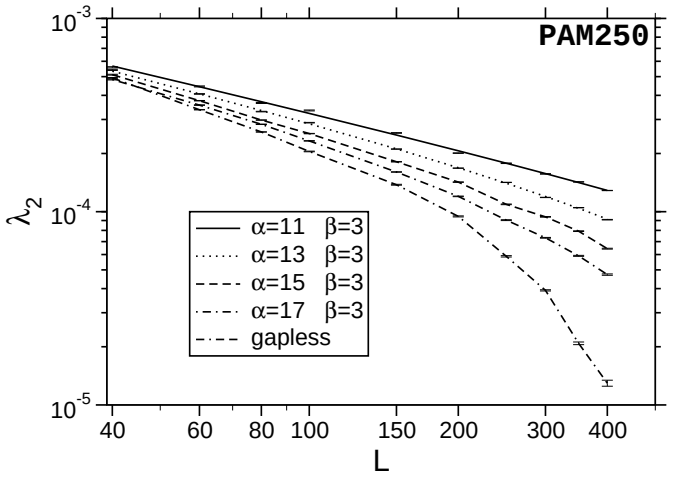
4.4 The biological example revisited

What do the results that we have seen above imply for the biological example that has been discussed in Sec. 3.6.1? Without any adjustment, the E-value that is reported by BLAST is determined by the Karlin-Altschul formula [BLA]

$$E = K L'_\mathtt{Q} N' \exp^{-\lambda s} \tag{4.9}$$



(a)



(b)

Figure 4.6: Scaling of the correction parameter λ_2 ((a) `BLOSUM62`, (b) (`PAM250`)). The decay of λ_2 with system size shows approximately a power law near the logarithm-linear transition (two smallest gap costs). For this cases the fit to Eq. (4.8) is shown by a line ($\alpha = 10$) and dots ($\alpha = 12$). The lines of the remaining cases are guides to the eye connecting the data points.

where $L'_{\mathtt{Q}}$ is the "edge-corrected" query length and N' the "effective database size" (sum of lengths of all sequences stored in the database). The corrections account for the abovementioned edge effects in the high probability region. The BLAST E-values are listed in Tab. 4.2 which completes the examples from Tab. 3.1. The exponential in

subjects				E-value	
acc.no.	organism	length	S^{local}	BLAST	accurate
P68873	Pan troglodytes (Chimpanzee)	147	780	8×10^{-82}	3×10^{-146}
P18989	Procyon lotor (Raccoon)	146	709	1×10^{-73}	1×10^{-125}
P02088	Mus musculus (Mouse)	147	638	2×10^{-65}	2×10^{-107}
P84792	Aythya fuligula (Tufted duck)	147	558	5×10^{-56}	6×10^{-88}
P10060	Sphenodon punctatus (Hatteria)	146	496	7×10^{-49}	2×10^{-73}
Q90486	Danio rerio (Zebrafish)	148	417	1×10^{-39}	7×10^{-56}
O13077	Gadus morhua (Atlantic cod)	147	326	4×10^{-29}	1×10^{-39}
P56692	Dasyatis akajei (Red stingray)	142	200	2×10^{-14}	7×10^{-18}
acc.no.	protein	length	S^{local}	BLAST	accurate
P02042	Hemoglobin subunit delta	147	727	1×10^{-75}	4×10^{-131}
P02100	Hemoglobin subunit epsilon	147	607	1×10^{-61}	1×10^{-99}
Q8WWM9	Cytoglobin (Histoglobin)	190	173	1×10^{-11}	4×10^{-14}
B4DUI1	cDNA FLJ55163	136	93	0.039	1×10^{-3}

Table 4.2: Completion of Tab. 4.2. The last two columns show the BLAST E-value and its correction according to the accurate distribution.

Eq. (4.9) is derived from the approximation of the cumulative Gumbel distribution for large s

$$\mathrm{Prob}(S > s) = 1 - \exp\left[-e^{-\lambda(s-s_0)}\right] \approx \exp\left[-\lambda(s-s_0)\right].$$

To address the question how the E-values change when one consideres the accurate distribution, I adjusted the original BLAST E-value by

$$E^{\mathrm{acc}} = E\frac{\sum_{t=s}^{\infty} P(t)}{\exp\left[-\lambda(s-s_0)\right]},$$

where $P(s)$ is the accurate distribution and the parameter λ and s_0 are obtained by a fit. Hence, the ratio $\frac{\sum_{t=s}^{\infty} P(t)}{\exp[-\lambda(s-s_0)]}$ gives a correcting factor. To determine $P(s)$ I repeated the simulations for the model of i.i.d. sequence. Here, I used the Wang-Landau / generalized ensemble approach instead of parallel-tempering. In detail, I

used exactly the same query length ($L_{\mathtt{Q}} = 147$) and the same subject sequence lengths as listed in Tab. 4.2. The distribution was obtained over a very broad range up to the scores that occur in the result set. For example, for $L_{\mathtt{S}} = L_{\mathtt{Q}} = 147$ we obtain $P(780) = 3 \times 10^{-156}$. The corrected E-values are listed in the last column in Tab. 4.2. The discrepancies between the original and the corrected E-value spread over several orders of magnitude. Hence the BLAST E-value underestimates the statistical significance, which would yield false negatives, when one is interested in strongly significant results.

4.5 Statistics of position dependent alignment for transmembrane protein models

Most of the existing statistical work for pairwise sequence comparison focuses on null models where both sequences are random and at each position a symbol $a \in \Sigma$ is chosen independently of the other positions. These models are governed by the Karlin-Altschul-Dembo theory (see Sec. 4.1). We shall refer to this model later as "random query - general-purpose scoring" (RQGS). The (RQGS) model is convenient, because the problem of computing significance values reduces to the estimation of only two parameters λ and K for each scoring scheme, which can be pre-computed. The results from the last section suggest that a third length dependent parameter λ_2 is required when one desires a better accuracy.

It is not always possible to extend the Karlin-Altschul-Dembo theory to more complex null models than the i.i.d. model, which is one of the reasons that they are not used in practice. Another striking consequence is the following one: The E-values reported by (the original) BLAST only depend on the raw score and query and subject length, and not on the individual query. This leads to large distortions when the query composition does not match the null model composition. For example, when we run a homology search for the Human transmembrane protein rhodopsin (UniProt accession `P08100`) with BLAST (BLOSUM 62, gap-init 12, gap-extend 1, no composition adjustment, no filtering), we find a possibly remote homolog `Q8NH42` with an E-value of $9 \cdot 10^{-8}$. However, using a recent "composition-based adjustment" option [YWA03, YA05] leads to a very different E-value of 0.001 for the same protein.

The statistics of position-dependent scoring and/or gap-cost schemes, like used in PSI-BLAST [AMS$^+$97] or in hidden Markov model (HMM) frameworks, are much less well explored. The central question here is, "given a query $\mathbf{a}$ and a position-specific scoring scheme, what is the score distribution when random null-model sequences of given length are scored against $\mathbf{a}$?". We refer to this model as "fixed query - position-dependent scoring" (FQPS). As a compromise between the general (RQGS) and the very specific (FQPS) models, one may release the i.i.d. assumption on the query of the (RQGS) model and draw query sequences according to probabilities given by an HMM.

In the following two subsections, we discuss the statistics for transmembrane proteins (see Sec. 3.2.3). Obviously this biologically important class of proteins is hardly described by an i.i.d. model, because the amino acid composition strongly depends on the position in sequence. As pointed out in Sec. 3.2.3 the helical membrane regions mainly consist of hydrophobic amino acids. Here, we discuss the statistics under the bipartite scoring model for transmembrane proteins that was discussed in Sec. 3.2.3

[MRR01]. Recall that the scoring function for this model is defined by Eq. (3.5),

$$\mathcal{S}(\mathcal{A},\ \mathbf{a}, \mathbf{b}) = \sum_{(i,j)\in\mathcal{A}} \begin{cases} \sigma^{\texttt{slim}}(a_i, b_j) & \text{if } i \text{ is a transmembrane position} \\ \sigma^{\texttt{blosum}}(a_i, b_j) & \text{otherwise} \end{cases} - \sum_{\Gamma} g(l_\Gamma).$$

In order to assign each position an indicator to decide whether it belongs to a transmembrane helix or a globular region, the scoring scheme requires a suitable method to predict the loci of the transmembrane regions. For this purpose various methods are available (see footnote on page 43). For practical reasons and because the benchmark results are quite convincing [CDS05], I used TMHMM (*Transmembrane Hidden Markov Model*) [SvHK98, KLvHS01].

Before describing the features of TMHMM, some important features of HMMs [Rab89, RDM98] are stated briefly. In this general probabilistic framework one assumes that a sequence of observed "output" symbols is generated through a sequence of "hidden" states. This state sequence, also called *path*, follows a simple Markov chain. The states are connected to the output symbols through emission probabilities; that is, a state can produce a symbol according to a distribution over all possible symbols. More formally, a HMM consists of

- a finite set Σ of symbols (in our case the amino acid alphabet),
- a finite set Q of (hidden) states,
- inital state probabilities π_μ for all $\mu \in Q$ with $\sum_{\mu\in Q} \pi_\mu = 1$,
- emission probabilities p^μ_σ in each state $\mu \in Q$ and for all $\sigma \in \Sigma$ with $\sum_{\sigma\in\Sigma} p^\mu_\sigma = 1$,
- a stochastic transition probability matrix $P_{\mu,\tau}$ $\mu, \tau \in Q$, i.e. $\sum_{\tau\in Q} P_{\mu,\tau} = 1$ for all $\mu \in Q$

The sequence of hidden symbols $Z_1 \dots Z_L$ and the sequence of output symbols $X_1 \dots X_L$ is a stochastic process, which characterized by the transition matrix $P_{\mu,\tau}$ together with the emission probabilities p^μ_σ. One can generate such sequences $\mathbf{x} = x_1 \dots x_L$ and $\mathbf{z} = z_1 \dots z_L$ via simple sampling. Given these model parameters and a fixed sequence $\mathbf{x} = x_1 \dots x_L$ of output symbols, the state sequence $Z_1 \dots Z_L$ is also a stochastic process.

For the Monte Carlo sampling as needed here, it is not possible to simulate a HMM directly to generate output sequences, since importance sampling changes the underlying sequence probabilities. Nevertheless, one still needs to compute the probabilities $f^{\text{HMM}}(\mathbf{x})$ for the Monte Carlo acceptance procedure, i.e. the probabilities that $\mathbf{x}$ is the observed sequence generated by the HMM. These probabilities can be computed in $\mathcal{O}(L \cdot |Q|^2)$ time using the well known *forward algorithm* [RDM98, Rab89] as described in the following. One introduces the auxiliary variables $f_\mu(i)$, which correspond to the probability that the subsequence $x_1 \dots x_i$ is generated by the model given that the last state variable Z_i has the value μ, i.e. $f_\mu(i) = \text{Prob}(X_1 \dots X_i = x_1 \dots x_i | Z_i = \mu)$. The overall probability is then $f^{\text{HMM}}(\mathbf{x}) = \sum_{\mu\in Q} f_\mu(L)$. The probabilities $f_\mu(i)$ can be determined by the recursion

$$f_\mu(i) = p^\mu_{x_i} \sum_{\tau\in Q} f_\tau(i-1)\, p_{\tau,\mu} \tag{4.10}$$

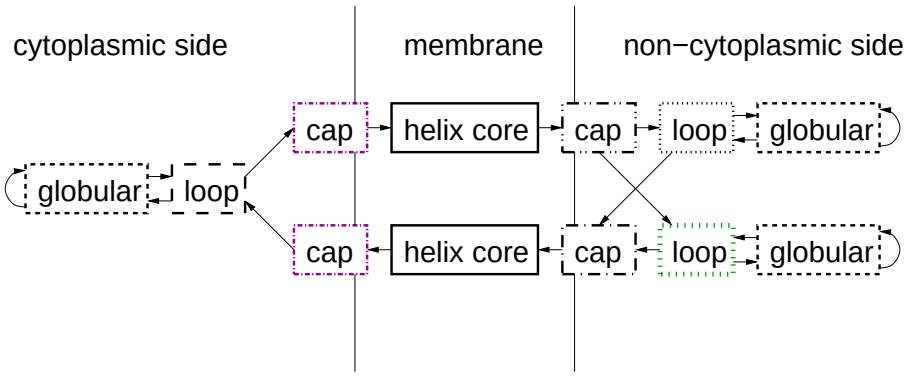


Figure 4.7: The layout of the HMM for transmembrane proteins according to Sonnhammer et.al. [SvHK98]. Each box corresponds to a group of states. For example, the helix-core block consists of 25 internal states. Each line type of the boxes represent different emission probabilities. For more details we refer the reader to the original publication.

with initial conditions $f_\mu(1) = \pi_\mu p^\mu_{x_1}$.

Within the same time complexity, the *Viterbi algorithm* $\mathcal{V}$ computes the most probable state path for a given sequence of observations, that is

$$z_1 \dots z_L = V(x_1 \dots x_L) = \text{argmax}_{\bar{z}_1 \dots \bar{z}_L \in Q^L} \operatorname{Prob}(Z_1 \dots Z_L = \bar{z}_1 \dots \bar{z}_L | x_1 \dots x_L).$$

Let $v_\mu(i)$ be the probability of the most probable path ending in state $\mu \in Q$ with observation x_i. These values can be computed recursively by

$$v_\mu(i) = p^\mu_{x_i} \max_{\tau \in Q} \{v_\tau(i-1)\, p_{\tau,\mu}\} \tag{4.11}$$

with boundary condition $v_\mu(1) = \pi(\mu) \cdot p^\mu_{x_1}$. Note that these probabilities are not normalized, in particular $\sum_{\mu \in Q} v_\mu(i) \leq 1$. The most probable path is reconstructed by backtracing [RDM98].

The HMM approach we use to sample transmembrane queries is the TMHMM developed by Sonnhammer et al. [SvHK98]. In this setting, the output symbols are (structural) domains, and hidden states are "tied" according to their emission probabilities. They are classified into seven groups (see Fig. 4.7):

- Helix core,
- two different groups of caps (a crossover region between an helix and a loop) on either side,
- loops on the cytoplasmic side,
- short and long loops on the non-cytoplasmic side,
- globular domains.

The internal structure of the helix core and loop module allows modeling different lengths of the corresponding protein domain by assigning jump probabilities. The globular domains have a self-looping structure and hence may also have various lengths.

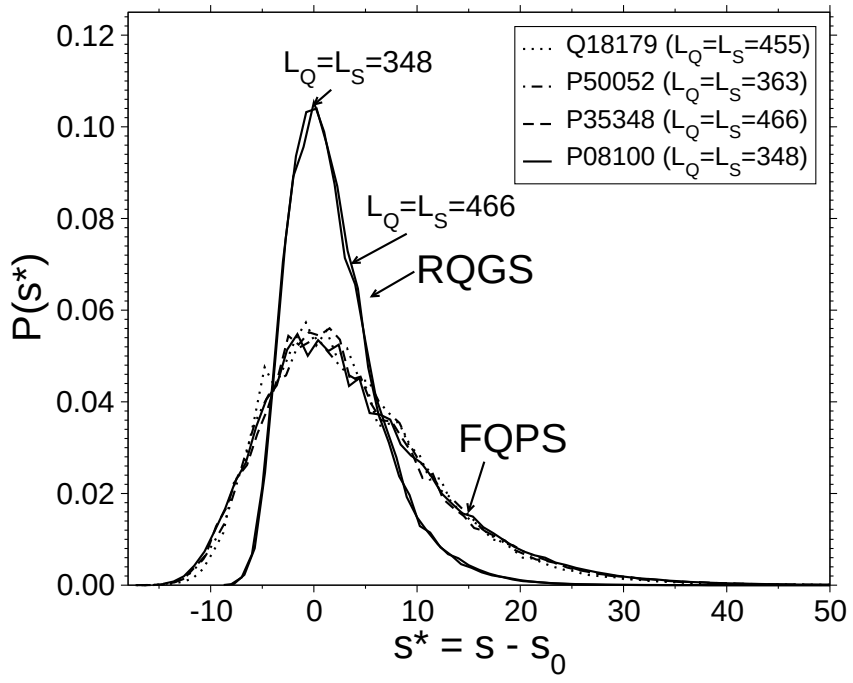


Figure 4.8: The score distributions for (RQGS) (classical) and (FQPS) models where the subject length equals the query length. In order to compare the shape, the distributions have been shifted by the center s_0. All distributions from the (RQGS) agree outside the tails (only two lengths are shown).

AC	Description	Organism	Length
P08100	Rhodopsin	H. sapiens	348
P50052	type-2 angiotension II receptor	H. sapiens	363
Q18179	putative neuropeptide Y receptor	C. elegans	455
P35348	Alpha-1A adrenergic receptor	H. sapiens	466

Table 4.3: A selection of transmembrane proteins.

The other modules have fixed length. The overall number of model parameters is 216. Fig. 4.7 shows the actual layout of TMHMM. Each box represents a group of "tied" states. The states corresponding to "helix core" represent the transmembrane helices that connect states of the cytoplasmic side and the non-cytoplasmic side of the membrane. The prediction of the positions of the "helix core" states determines the loci of the special purpose scoring matrix SLIM for position specific alignment.

4.5.1 Fixed queries versus random subjects

In the model of (FQPS), the query sequence $\mathbf{a}$ remains fixed whereas the subject that models the composition of the database is random. Recall that the `SLIM` matrix is especially designed for aligning transmembrane regions against general proteins, which

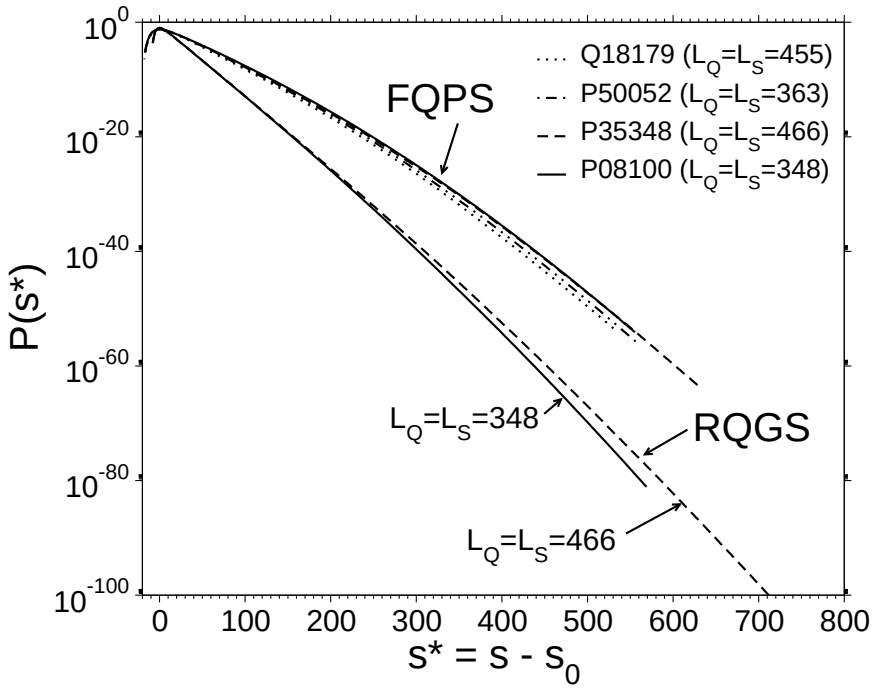


Figure 4.9: The rescaled score distributions for (RQGS) (classical) and (FQPS) models where the subject length equals the query length in a logarithmic view.

are well described by the i.i.d. model. Hence the score distribution Eq. (4.1) in this framework reads

$$P(s;\mathbf{a}) = \sum_{\mathbf{b}\in\Sigma^M} \delta_{s,\mathrm{S}_0^{\mathrm{local}}(\mathbf{a},\mathbf{b})} \cdot p^{\mathrm{subject}}(\mathbf{b}).$$

We discuss four different transmembrane proteins as queries (see Tab. 4.3) in the (FQGS) scheme.

First, the transmembrane helical regions had been predicted once for each protein. The score distribution is obtained by Wang-Landau sampling combined with a final Metropolis run in the generalized ensemble. Some results are shown in Fig. 4.8 and Fig. 4.9, where the distributions of (FQGS) and (RQGS) are compared against each other. The subject lengths equal the query lengths here. For the production run of one distribution that is show in Fig. 4.9 ($L_{\mathtt{Q}} = LS = 348$) 16,777,216 Metropolis-Hastings updates had been performed. This took about 16 hours on an Intel Pentium 4 with 3.4GHz.

We observe that the curvature is more pronounced in the (FQPS) model: Significant differences of shapes already show up in the high probability region, which is accessible by simple sampling (Fig. 4.8). All (RQGS) and (FQPS) distributions match almost perfectly (only two lengths for (RQGS) are shown here)

More pronounced differences are seen in the behavior of the tail (Fig. 4.9), which is accessible via our importance sampling approach. The difference between the probabilities spans several orders of magnitude; hence a wrong choice of the model would falsify the estimation of significance drastically. Most importantly, the entropy func-

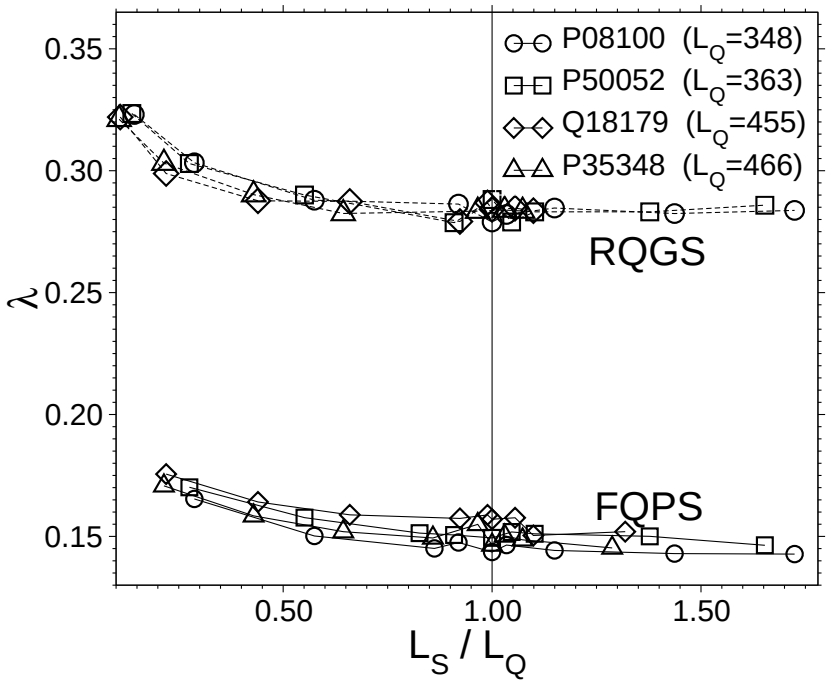


Figure 4.10: Dependence of the modified Gumbel parameter λ on the subject/query length ratio $L_{\mathtt{S}}/L_{\mathtt{Q}}$. The lines are guide to the eyes only. The vertical line corresponds to Fig. 4.8 and Fig. 4.9, where $L_{\mathtt{S}} = L_{\mathtt{Q}}$. For $L_{\mathtt{S}} > L_{\mathtt{Q}}$, λ varies only slightly in the subject length.

tion obtained using the position-specific scoring is considerably curved. Thus, using estimates from fits to data of the high-probability region is even more questionable here than in the (RQGS) model, where the entropy function is almost a straight line.

To investigate the impact of dissimilar query and subject lengths $L_{\mathtt{Q}}$ and $L_{\mathtt{S}}$ on the parameters of the modified Gumbel distribution, I varied $L_{\mathtt{S}}$ and consider the parameters λ and λ_2 as functions of the ratio $L_{\mathtt{S}}/L_{\mathtt{Q}}$ (see Fig. 4.10 and Fig. 4.11). All resulting fit parameters are summarized in Tab. C.3 in the appendix. The large gap between the values of λ for the two different models reflects the qualitative difference of the shape in the high probability regime. We see that in the models, λ is virtually independent of query and sequence length. However, in model (FQPS), λ varies with each individual query only slightly. For λ_2 one has to distinguish between $L_{\mathtt{S}} < L_{\mathtt{Q}}$ and $L_{\mathtt{S}} > L_{\mathtt{Q}}$. In the first case, λ_2 decreases, which is not surprising, since the correction term describes a finite-size effect and should vanish for increasing sequence lengths.

Once the target subject exceeds the query length, the search space is still growing, but the finite length of the query enforces subject length independent edge effects.

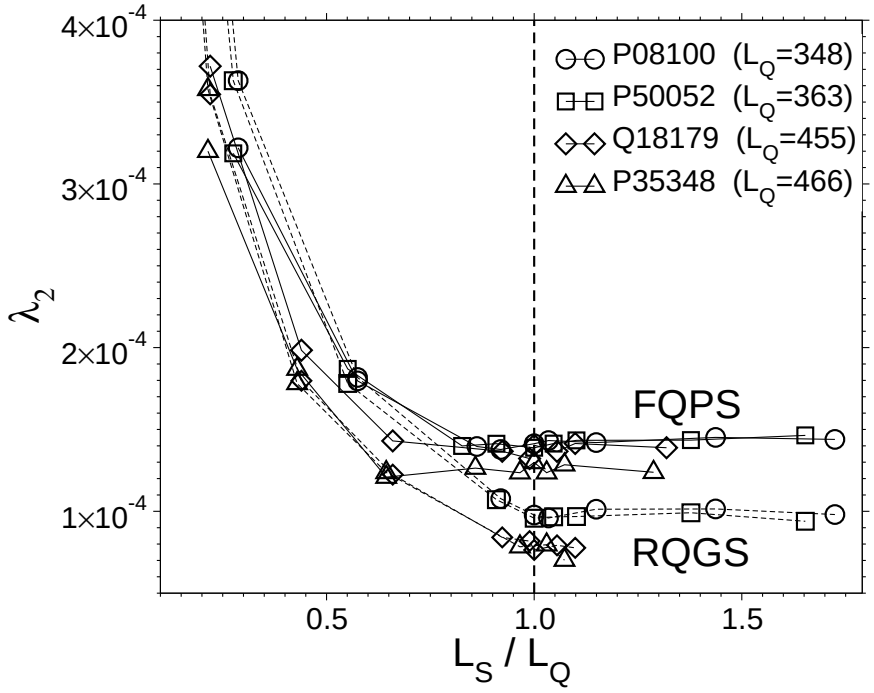


Figure 4.11: Dependence of the modified Gumbel parameter λ_2 on the subject/query length ratio $L_{\mathtt{S}}/L_{\mathtt{Q}}$. The vertical line corresponds to Fig. 4.8 and Fig. 4.9, where $L_{\mathtt{S}} = L_{\mathtt{Q}}$. The lines are guide to the eyes only. For $L_{\mathtt{S}} > L_{\mathtt{Q}}$, λ varies only slightly in the subject length. The parameter λ_2 characterizes the curvature of the entropy function in the tail (see Fig. 4.9). Large differences between (RQGS) and (FQPS) show up in the case where $L_{\mathtt{S}} > L_{\mathtt{Q}}$. λ_2 becomes subject-length independent for $L_{\mathtt{S}} > L_{\mathtt{Q}}$.

4.5.2 Random queries and position specific scoring

The statistics for (FQPS) is very accurate because it models a realistic search procedure, where a fixed query is searched against variable subjects. In practice such simulations are not feasible for each individual query that occur in typical large scale research projects. The TMHMM compromises between the model of (RQGS) and (FQPS), because we may

- draw sequences with Monte Carlo sampling and the probabilities $p^{\text{query}}(\mathbf{b})$ (via the forward algorithm Eq. (4.10)) and
- predict the transmembrane regions (the most likely path through the HMM via the Viterbi algorithm Eq. (4.11))

in polynomial time. The model contains more information than the distribution of S in the sense that each randomly drawn query is a member of a certain sub-class. These classes are characterized by the number of transmembrane regions "# of TM helices". Below we will denote this function as $N : \Sigma^{L_{\mathtt{Q}}} \to \mathbb{N}$. This observable is determined by a simple analysis of the output of the Viterbi algorithm.

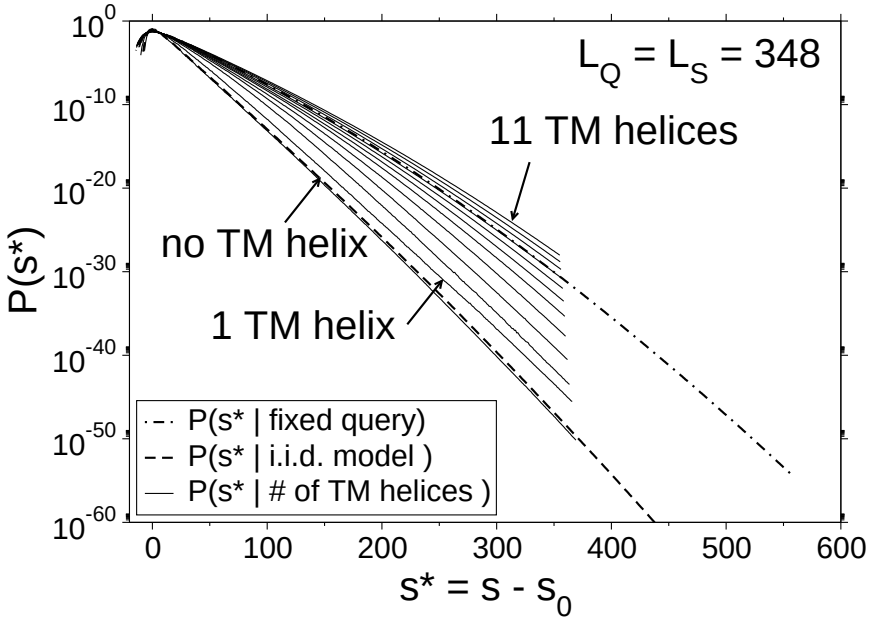


Figure 4.12: Score distributions for different alignment models (i.i.d., fixed query and TMHMM) with $L_{\mathtt{S}} = L_{\mathtt{Q}} = 348$. The distributions for the (HMM) have been obtained from the joint distribution.

In order to take this property into account we deal with the joint probability $\mathrm{Prob}(S = s, \#\text{ of TM helices} = n)$ and determine a score distribution for each class

$$P_n(s) = \mathrm{Prob}(S = s | \#\text{ of TM helices} = n).$$

In practice, when one wishes to query a transmembrane protein against a database, one first uses TMHMM or a related approach to setup the position specific scoring system and count the number n of transmembrane regions. To assess the statistical significance one may choose a query specific score distribution, $P_n(s)$. Below we shall see that these distributions $P_n(s)$ differ significantly for different n.

Because the subject is still i.i.d., I used a hybrid Metropolis-Hastings update rule, that combines the Metropolis-Hastings update for the subject Eq. (4.6) sequence and Eq. (4.5) for the query sequence,

$$\alpha_{(\mathbf{a},\mathbf{b}),(\mathbf{a}^*,\mathbf{b}^*)} = \min\left\{1, \frac{w(\mathsf{S}_0^{\text{local}}(\mathbf{a}^*,\mathbf{b}^*); N(\mathbf{a}^*)) \cdot p^{\text{query}}(\mathbf{a}^*)}{w(\mathsf{S}_0^{\text{local}}(\mathbf{a},\mathbf{b}); N(\mathbf{a})) \cdot p^{\text{query}}(\mathbf{a})}\right\} \quad .$$

For the subject the newly drawn letters are sampled from the `BLOSUM` letter frequencies, and those for the query from the uniform frequencies.

The performance of the Monte Carlo simulation of the HMM is weaker than for (FQPS) or (RQGS) for three reasons: Firstly, we are interested in a joint distribution for that we need more samples. Secondly, more proposals are rejected from the sampler

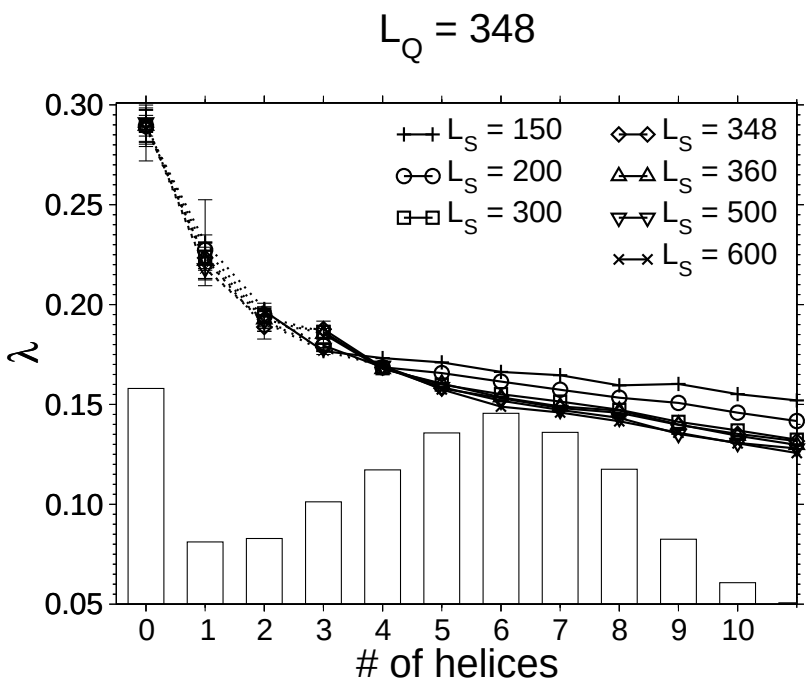


Figure 4.13: Fit parameter λ for the score distributions $P(S|\#\text{ of helices})$ for the (HMM) with a fixed query length $L_{\mathtt{Q}} = 348$ and various subject lengths $L_{\mathtt{S}}$. The lines are guide to the eyes only. The shape parameter λ decreases with increasing number of helices. The bars show the distribution of number of transmembrane helices obtained by direct simulations of the (HMM).

due to the HMM-weights and finally the computation of the forward-probabilities requires additional floating point operations. The computation of 16,777,216 Metropolis-Hastings updates for this model costs about 45 CPU hours. We use an 8 times larger sample size in order to account for the first drawback. Hence, we put an overall computational effort on this model, which is 23 times as large as for (FQGS) and (RQGS) (apart from the Wang-Landau iterations).

Next, we discuss the results for this model. I approximated the score distribution within each class (number of helices$= n$). The shape of the distributions clearly agrees with the curvature for (RQGS) and (FQPS) and the modified Gumbel distribution could be fitted (see Fig. 4.12) when the number of helices was not too small. This is indicated by a large reduced χ^2 value for distributions with a small number of helices. Also a visual inspection of the fit to the data supports this argument. All resulting fit parameters are summarized in Tab. C.4 and Tab. C.5 in the appendix.

The rare-event tail shows clear differences between the different sub-classes of the model over several orders of magnitude. Fig. 4.13 and Fig. 4.14 display the dependency of the fit parameters on the respective sub-class of the model. The effect of the ratio of sequence lengths $L_{\mathtt{S}}/L_{\mathtt{Q}}$ is shown in Fig. 4.15(a) and Fig. 4.15(b). Note that for distributions that are not well described via Eq. (4.4), I only fitted the data in the high probability region. Those data points are left out in the plot for λ_2 in Fig. 4.15(b) and

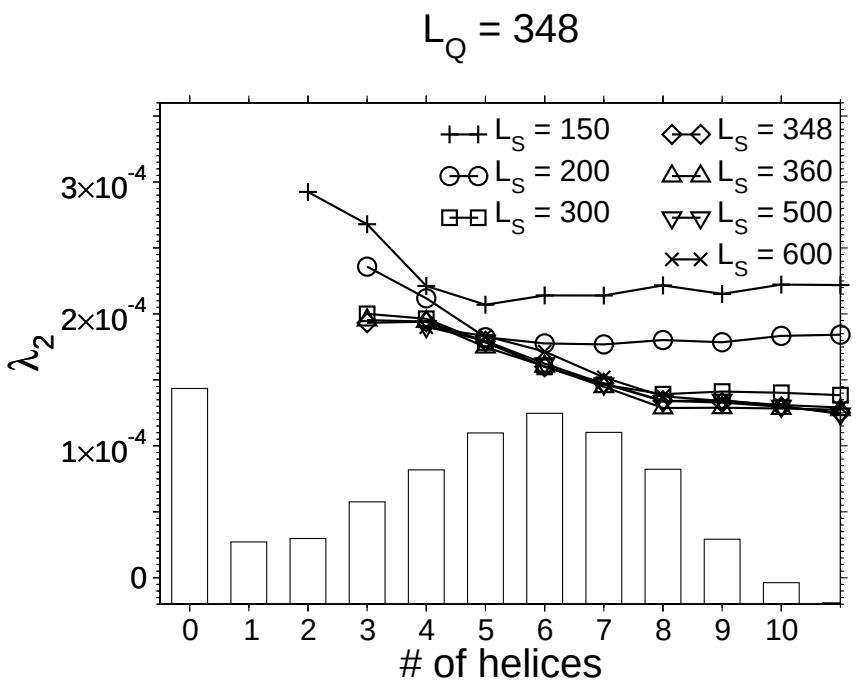
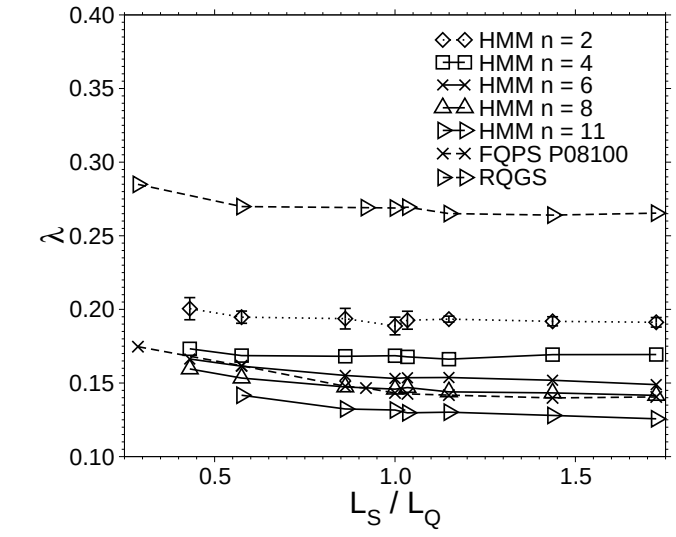


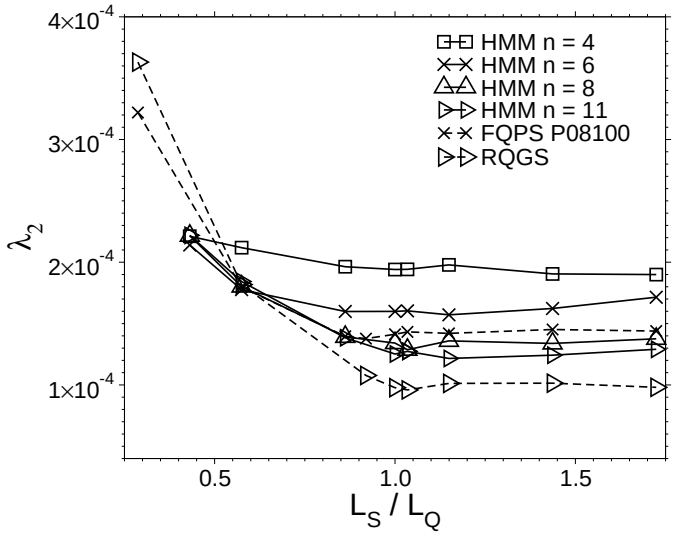
Figure 4.14: Fit parameter λ_2 for score distributions $P(S|\#\text{ of helices})$ for the (HMM) with a fixed query length $L_Q = 348$ and various subject lengths L_S. The lines are guide to the eyes only. Like λ, the shape parameter λ_2 decrease with increasing number of helices. The dependency on the subject length is stronger for λ_2 than for λ. For $L_S > L_Q$ the dependency of λ_2 on the subject length is only of marginal order. The bars show the distribution of number of transmembrane helices obtained by direct simulations of the (HMM).

are connected by dotted lines in Fig. 4.15(a).

In analogy to (RQGS) and (FQPS) the curvature remains constant when $L_S > L_Q$. Regarding the dependence on the number of helices the curvature decays with increasing number of transmembrane regions and then approaches an approximate constant value.



(a)



(b)

Figure 4.15: The L_S/L_Q dependency of λ (a) and λ_2 (b) extracted from the same data as in Fig. 4.13. The lines are guide to the eyes only. Dashed lines show the corresponding scaling behavior for the (FQRS) and (RQGS) models. The result for $n = 2$, that has been obtained from the high probability regions (see text), is indicated by dotted lines in (a)

4.6 Phase diagram and statistics of finite-temperature alignment

Sec. 3.4 treats the generalization of the optimal score (ground-state energy) to a canonical ensemble of sub-optimal alignments (finite-temperature alignment). The optimal score is replaced by a free energy F. In the same spirit as for the optimal score, we investigate the free-energy distribution over an ensemble over random i.i.d. sequences. The biological relevant 20 amino acid alphabet together with the `BLOSUM62` score matrix was used for this purpose.

In order to choose a temperature range, where one would expect an interesting crossover of the shape of the free-energy distribution, I first looked at the linear-logarithmic phase transition (see Sec. 3.5, [WGA87, AW94, BH00]) for finite temperature alignment [KL00].

At $T = 0$, the critical values were studied analytically by a self-consistent equation [BH00] or numerically by a finite-size scaling analysis [SAY05]. Both studies rely on a simple scoring model with a single mismatch parameter. In the latter approach the problem was approached by considering the linear-logarithmic phase transition as a percolation phenomenon [SA94]. In percolation problems one usually asks the question under which conditions a geometric object spans a volume of interest.

I adopted some ideas from the work of Sardiu et. al. [SAY05] for local alignment with the abovementioned scoring parameters [2]. As outlined in Sec. 3.5, the gap-costs are the crucial parameters that control whether alignments grow linearly (small gap costs) or logarithmically (large gap costs). Hence there is a critical gap cost parameter α_c (we consider affine gap-costs with $\beta = 1$ and only vary α throughout this section), at which the transition occurs.

I probed the critical line in the $\alpha - T$ plane that separates the linear phase from the logarithmic one. For that purpose we require a definition of a percolation criterion $h : \chi_{\mathbf{a},\mathbf{b}} \to \{0, 1\}$ that assigns each alignment a binary decision, "non-percolating" or "percolating". There are various possibilities to achieve this [SAY05]. Here, I regard an alignment $\mathcal{A}$ as percolating, $h(\mathcal{A}) = 1$, if the distances between the first aligned letter and the last aligned letter in both sequences are larger than $L/2$ at the same time. Otherwise $h(\mathcal{A})$ is set to 0. This choice is motivated by the fact that for large sequence lengths essentially all alignments in the logarithmic phase $\alpha > \alpha_c$ are reported as non-percolating, i.e. $h(\mathcal{A}) = 0$, whereas the opposite occurs in the linear phase $\alpha < \alpha_c$ where the alignment length grows like the sequence length.

The phase transition is investigated by the average percolation probability

$$P^{\text{perc}}(\alpha; T; L) := \left\langle \frac{1}{Z_{T;\mathbf{a},\mathbf{b}}} \sum_{\mathcal{A} \in \chi_{\mathbf{a},\mathbf{b}}} h(\mathcal{A}) \cdot e^{\mathcal{S}(\mathcal{A};a,b)/T} \right\rangle,$$

where $Z_{T;\mathbf{a},\mathbf{b}}$ denotes the partition function of the canonical alignment ensemble over a fixed realization $\mathbf{a}$ and $\mathbf{b}$, i.e. a pair of sequences. The average $\langle\cdot\rangle$ is taken over these realizations of the disorder of random i.i.d. sequences.

Thanks to finite size scaling theory [SA94], we may extrapolate data from finite sequence length to the thermodynamic limit $L \to \infty$. In this limit the percolation probability approaches a step function, which is 1 for $\alpha < \alpha_c$. In finite systems,

[2] Sardiu et. al. studied percolation of global alignment. They considered ground states alone and varied gap costs and a disorder parameter that models a simple scoring system.

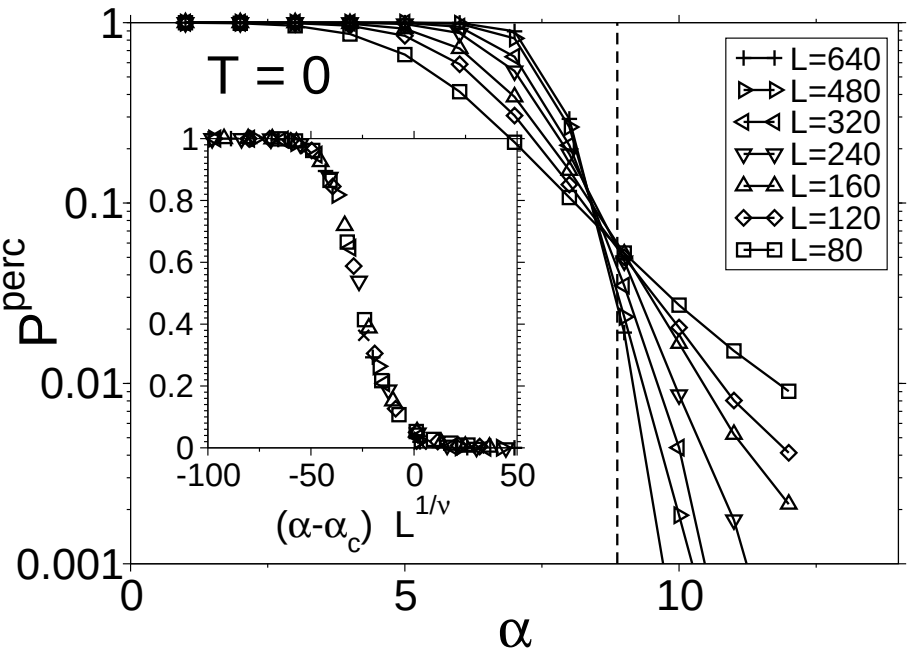


Figure 4.16: Percolation probability $P^{\text{perc}}(\alpha; T = 0; L)$. Dashed line indicate the critical paramater obtained by the finite-size scaling algorithm [HH04].
Inset: rescaled data according to Eq. (4.12).

$L < \infty$, the crossover is smeared out meaning that a finite percolation probability remains even above α_c.

Scaling theory states that the behavior of $P^{\text{perc}}(\alpha; T; L)$ close to criticality is described by

$$P^{\text{perc}}(\alpha; T; L) = \tilde{P}^{\text{perc}}\left[(\alpha - \alpha_c)L^{1/\nu}\right], \tag{4.12}$$

where $\tilde{P}^{\text{perc}}$ is an universal scaling function. We may use Eq. (4.12) to extract the critical exponents ν and the critical gap costs α_c as a function of the temperature simultaneously. The fit is performed by minimizing a weighted-χ^2-like objective function S [HH04], that measures the distance (measured in standard errors) of the data from the master curve.

To numerically determine the probabilities $P^{\text{perc}}(\alpha; T; L)$, I generated N random pairs of sequences and drew M (finite temperature) alignments for each realization. The method to sample alignments from the canonical distribution [MHS02] is explained in Appendix A.1. The temperature varied between $T = 0$ (optimal alignment) and $T = 4$. Fig. 4.16 displays the empirical percolation probabilities $P^{\text{perc}}(\alpha; T; L)$ for $T = 0$ and sequence lengths between $L = 80$ and $L = 640$. In fact I used lengths up to $L = 1920$, but the transition curves for larger sequences look quite similar to those of $L = 640$. I used $N = 1600$ realizations for the largest systems and $N = 12,800$ for the smallest one. For each realization $M = 100$ alignments were drawn from the canonical ensemble. The inset in Fig. 4.16 shows the rescaled data. Although the visual inspection suggests a quite nice collaps, the fit turned out to be not very accurate. I obtain a value $S = 60$ for $T = 0$, where a value of $S \approx 2$ is desirable for a strongly reliable result.

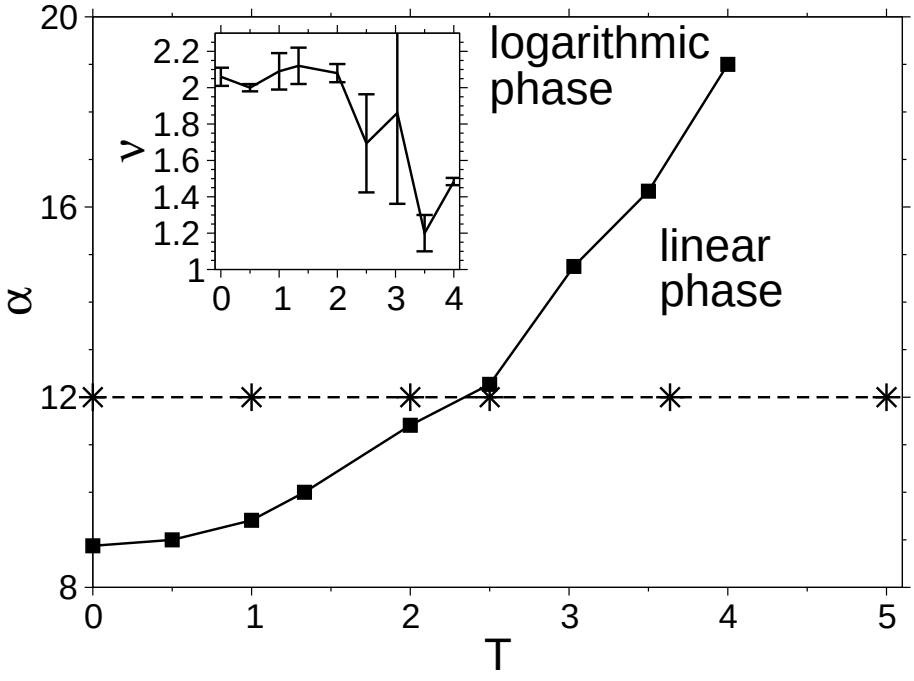


Figure 4.17: Phase diagram of finite-temperature alignment. The dashed line indicates the parameter-set of optimal alignment that is commonly used in bioinformatics $(\alpha = 12, \beta = 1)$. Stars show data points at which the free-energy distributions were obtained.
Inset: The critical exponent ν as a function of the temperature.

However, the data is good enough to approximately determine the critical gap costs for different temperature values. The phase diagram in the $\alpha - T$ plane is shown in Fig. 4.17. One observes that the critical gap costs increase with the temperature. At infinite temperature a logarithmic growth of the alignment length is expected because short alignments are entropically favorable. Eventually there is a critical point, where the critical line ends. This has not been probed so far.

In comparison with the illustrations in Sec. 3.6.2, one finds that fundamental crossovers of thermodynamic properties come along with the percolation transition. For example, the specific heat exhibits a peak close to the transition and the expected score (internal energy) changes from a positive value in the logarithmic phase to a negative value in the linear phase (see Fig. 3.12). Furthermore, the phase space exhibits a hierarchical structure close to the transition (see Fig. 3.13).

The critical exponents as a function of the temperature are shown in the inset. These values have to taken with care because they are usually more sensitive to the quality of the collaps. For the critical exponent we observe a crossover from $\nu = 2$ for $T = 0$ and a value between 1.2 and 1.5 for larger temperatures.

Sardiu et.al. obtained values between $\nu \approx 2$ and $\nu \approx 2.5$ [SAY05]. However their approach differs in several points (see above). For random (bond) percolation the critical exponent is known exactly, $\nu = 4/3$. The critical exponent for larger temperatures seems to be closer to this value than for $T = 0$. A detailed analysis of

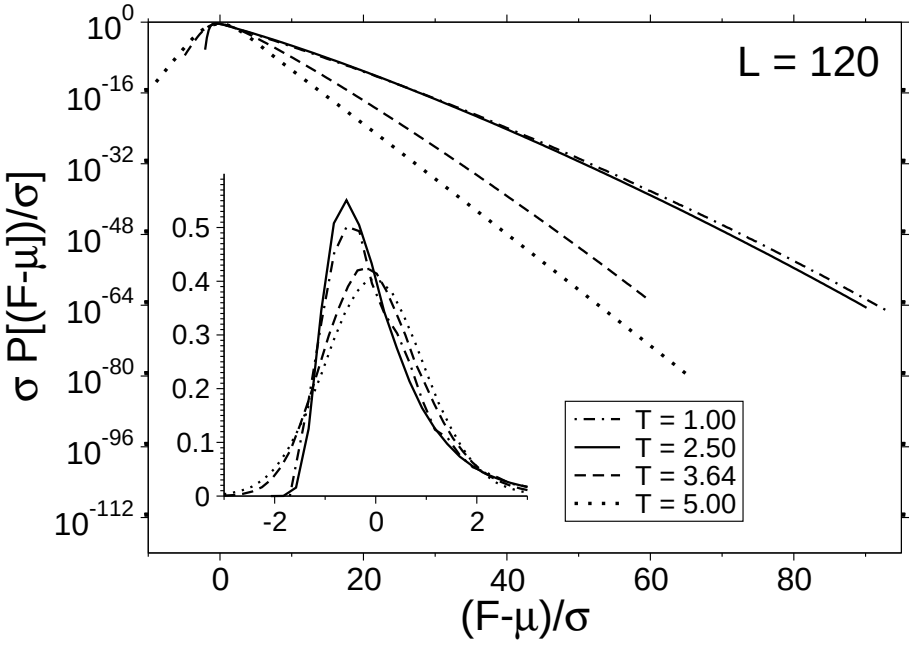


Figure 4.18: Rescaled free-energy distribution of finite-temperature alignments. At $T = 2.50$ and below the data is well described by a modified Gumbel distribution. For large temperature an exponential tail is observed.
Inset: The same data shown with a linear ordinate. In the high probability region the data for $T = 5.00$ is well described by a Gaussian distribution.

T	λ	$10^4\lambda_2$	s_0
0.00	0.2966(4)	3.182(1)	37.4(1)
1.00	0.2924(1)	2.900(5)	24.6(1)
2.00	0.2907(2)	3.122(7)	31.56(6)
2.50	0.2980(2)	3.16(1)	38.29(7)

Table 4.4: Fit parameters of least χ^2-fits of the free-energy distributions to the modified Gumbel distribution Eq. (4.4) for $L_{\mathtt{s}} = L_{\mathtt{q}} = L = 120$.

other critical exponents and scaling relations is beyond the scope here.

Instead, we use the phase diagram as a guide to study the free-energy distribution for various temperatures. I kept the gap-costs fixed ($\alpha = 12$, $\beta = 1$) fixed and only varied the temperature (between $T = 0$ and $T = 5$). The values are indicated by stars in the phase diagram in Fig. 4.17.

The simulations were performed in the generalized ensemble as above. The production run employed 4.8×10^7 Monte Carlo steps for each distribution. In the logarithmic regime ($T = 0, 1, 2, 2.5$) the free-energy distribution is well described by the modified Gumbel distribution Eq. (4.4) (see Fig. 4.18). Note that I have rescaled the distributions to unit variance and zero mean. The fit parameters only change slightly with the temperature (see Tab. 4.4).

The crossover from the logarithmic to the linear regime comes along with a change of the skewness, as can be seen in the inset of Fig. 4.18. In the high probability region and for $T = 5.00$ a Gaussian distribution describes the data well. This was confirmed by a Kolmogorov-Smirnov test that yields a p-value of 0.14. For $T = 1/0.275 \approx 3.64$ the evidence for a Gaussian distribution is much smaller (a p-value of 2×10^{-11}). I also checked that the change of the shape accompanies a change from logarithmic to linear growth of typical free energies (the position of the maximum) with the sequence length, i.e. the free energy becomes extensive (not shown here). This result is not surprising because the partition functions that occur in the transfer matrix calculation Eq. (3.9) become (more or less) independent and hence factorize. The total free energy decomposes into a sum of independent contributions and the central limit theorem applies.

When considering the rare-event tail at higher temperatures, the free-energy distribution is rather exponential than Gaussian, as can be seen in the main plot of Fig. 4.18. Hence, we observe a crossover from a Gaussian distribution in the high probability region to the characteristic exponential tail of the Gumbel distribution. With the same argumentation as for the optimal alignment, sequence pairs appearing in the tail feature high similarities. The overall free energy is dominated by the the ground state. This was confirmed by looking at the difference between the free energy and the ground-state energy for those sequences that occur in the tail of the distribution. The summation in the transfer matrix are virtually replaced by maximizations yielding an exponential tail. The finite-size effect that is responsible for the curvature of the optimal alignment statistics seems to be of marginal order in this case.

4.7 Concluding discussion

In this chapter, I have presented a simple universal method to accurately sample the far right tails of the score distribution of various sequence comparison algorithms. The most widely used search program, BLAST [BLA], reports E-values that are based on the assumption that the Gumbel distribution is the accurate distribution for finite sequences. We observe clear deviations from the Gumbel distribution in the biologically relevant rare-event tail, which is out of reach of simple sampling methods used so far. In almost all cases, a modified Gumbel distribution turns out to be a suitable description of the data.

The method has a disadvantage: Because of the high number of samples required for estimation of the distribution, it can presently not be used in on-line database search web services. For example, to generate the 16,777,216 samples for Fig. 4.8($L_{\mathtt{Q}} = L_{\mathtt{S}} = 348$) took approximately 16 hours on an Intel Pentium 4 with 3.4GHz. Very recently (during the preparation of this dissertation), a promising alternative method has been published [New08]. This allows one to draw sequences from an importance sampling distribution in a direct way, i.e. with zero autocorrelation. This means the Markov-Chain Monte Carlo approach to the alignment statistics problem seems to become obsolete in near future.

Chapter 5

RNA secondary structure prediction

Biopolymers such as DNA, RNA or proteins are heteropolymers. That means they consist of different types of linearly connected monomers. This linear connection is called “backbone”. In the case of RNA, which is considered here and in the following two chapters, the monomers are called “nucleotides”. They consist of one out of four nitrogenous bases (adenine (A), cytosine (C), guanine (G) or uracil (U)), a ribose sugar and a phosphate connected through phosphodiester bonds. The sequences of bases are referred as “primary structures”.

In the last two decades fundamental knowledge about RNA has been achieved, in particular the fact that the transport of genetic information (via messenger RNA, or mRNA), where the relevant description is the primary structure, is only one out of many functions of RNA.

Nowadays, it is established that RNA also work as catalyst [CZG81, GTGM$^+$83] and regulator [MG90]. In particular in biochemical processes in the ribosome, so called ribosomal RNA (rRNA) plays a leading role in the translation process [Nol91]. Together with the change of the viewpoint of RNA playing an active biochemical role instead of a passive information carrier, the spatial conformation of the molecule has become of particular interest, because, in analogy to proteins, the three-dimensional structure, or *tertiary structure*, determines the molecule’s function. However, the prediction of higher order structures from primary sequences is conceptional simpler than protein-folding, because the formation of *secondary structures* (i.e. the topology of the folded molecule in terms of paired bases) is energetically separated from the full three-dimensional structure [TB99]. This implies that the tertiary structure can be seen as a perturbation to the secondary structure, in contrast to the protein folding problem. For this reason the RNA secondary structure that is determined by the primary sequence is already a meaningful description of the molecule.

Biochemically, the bases in the primary sequence interact with other ones in the same chain by forming hydrogene bonds. The base pairs adenine – uracil (A–U) are formed by two and cytosine – guanine (C–G) are formed by three hydrogen bonds. Pairs of bases that may form bonds are said to be complementary (A–U and C–G), or Watson-Crick pairs. In RNA in particular in tRNA some modified non-standard bases, such as Inosine (I), occur [Kni06]. Also non-Watson-Crick base pairings are possible, for example the ”Wobble“ pairs G–U, or I–C [Cri66]. Their occurrencies depend on

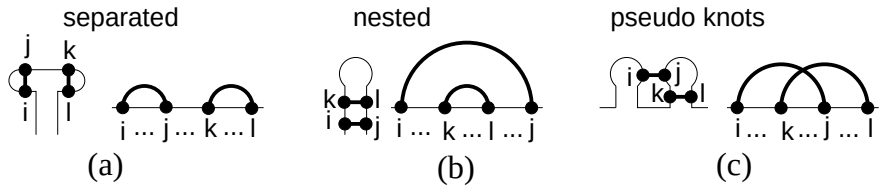


Figure 5.1: Three different cases of orders of the pairs (i, j) and (k, l) illustrated in the fold (left) and diagrammatic (right) representation of the RNA secondary structure: (a) separated pairs, (b) nested pairs and (c) pseudo knots.

the context of the nearest neighborhood in the primary sequence and are considered in modern models [MSZT99, XSB$^+$98], such as the free-energy model that is introduced in Sec. 5.3.

In order to formalize the secondary structure prediction algorithms in Sec. 5.2 and Sec. 5.3, the basic notation is fixed in the following section.

5.1 Notation of RNA secondary structures

The presentation here is restricted to the standard RNA alphabet. For realistic free-energy models, that are outlined in Sec. 5.3, non-standard letters and non-standard pairings are also considered.

Let Σ be the alphabet of bases ($\Sigma = \{A, U, C, G\}$) and $\mathbf{a} = a_1 \dots a_L \in \Sigma^L$ be an RNA-sequence over Σ. A base pair between the bases a_i and a_j is denoted by (i, j). Within this notation, we always assume that $i < j$.

Each base can be paired with another base at most once and for any two base pairs (i, j) and (k, l) there are in principle three possible cases, namely

(a) $i < j < k < l$ (separated pairs)

(b) $i < k < l < j$ (nested pairs) and

(c) $i < k < j < l$ (pseudo knots) .

These cases are illustrated in Fig. 5.1 in two different representations, the ”fold“ and ”diagrammatic“ representation. In both representations the backbone is shown as a thin line and paired bases are indicated as dots. Bonds between bases are indicated by bold lines connecting the dots. The fold representation resembles more the true structure, i.e. the backbone is flexible and the hydrogene bond representations are of equal length. In the diagrammatic representation the backbone is a straight horizontal line and bonds are drawn as arcs, whose radii measures the distance in the primary sequence. This kind of picture is interesting for computational and theoretic aspects.

When disallowing the case of pseudo knots, efficient algorithms to determine the minimum-free-energy structure based on free-energy models are available.

The precise notation of secondary structures is fixed by

Definition 5.1.1

(i) *A (pseudo-knot-free) secondary structure $\mathcal{C}$ on the sequence $\mathbf{a} = a_1 \dots a_L \in \Sigma^L$ is a set of pairings $\mathcal{C} = \{(i_1, j_1), \dots, (i_N, j_N)\}$ with $i_k < j_k$ for $k = 1, \dots, N$, such that each two bonds $(i, j), (k, l) \in \mathcal{C}$ with $i < k$ are either nested ($i < k < l < j$) or separated ($i < j < k < l$).*

(ii) *The state space of all secondary structures on $\mathbf{a}$ is denoted as $\chi_{\mathbf{a}}$.*

(iii) *The base a_i is called unpaired, if there is no $(i, j) \in \mathcal{C}$ or $(j, i) \in \mathcal{C}$.*

Note that this definition has remarkable similarities to the definition of the state space of sequence alignments, that was fixed in Def. 3.1.1 in Sec. 3.1. Firstly, both spaces are sets of pairings of letters either between two distinct sequences in the case of alignment or a self-interaction here. Secondly, disallowing crossings or pseudo knots allows for algorithms to find ground states and partition functions in polynomial time (see Sec. 5.2, Sec. 5.3 and [dG68, NJ80, ZS81, ZS84, McC90, HFS$^+$94] for the RNA secondary structure prediction) For sequence alignment these algorithms were described in detail in Sec. 3.3 and in Sec. 3.4. Thirdly, direct sampling from the Gibbs-Boltzmann distribution is possible in both cases (see [MHS02, Hig96], Appendix A.1 and Appendix A.2).

Each secondary structure can uniquely be decomposed in the so called *secondary structure elements*. That are different types of *loops*,

- *hairpin loops*,
- *stacked pairs*,
- *bulges*,
- *internal loops* and
- *multi-loops*,

and dangling ends at the begin and end of the sequence. Note that bases may belong to different loops. The loops are shown as grey areas in Fig. 5.2. The *topological order of a loop* is given by number of base pairs that close these areas. Hence an unpaired base and hairpin loops are of order $\mathcal{O}(0)$ and $\mathcal{O}(1)$ respectively. stacked pairs, bulges and internal loops are of second order,$\mathcal{O}(2)$, and, accordingly, multi-loops have higher orders than two. A formal definition can be found in Ref. [San85, HFS$^+$94, CB05].

So called *stacks* are an important feature of the secondary structure, because they stabilize the molecule. These objects are defined as

Definition 5.1.2 *A stack of size n is a set of consecutive base pairs $(i, j), (i + 1, j - 1), \dots, (i + n, j - n) \in \mathcal{C}$.*

Single bases and stacked pairs are special cases, i.e. stacks of size 0 and 1.

In the following the algorithms for a simple model of RNA secondary structures is introduced. After that, in Sec. 5.3 a more realistic free-energy model is discussed without going into the details.

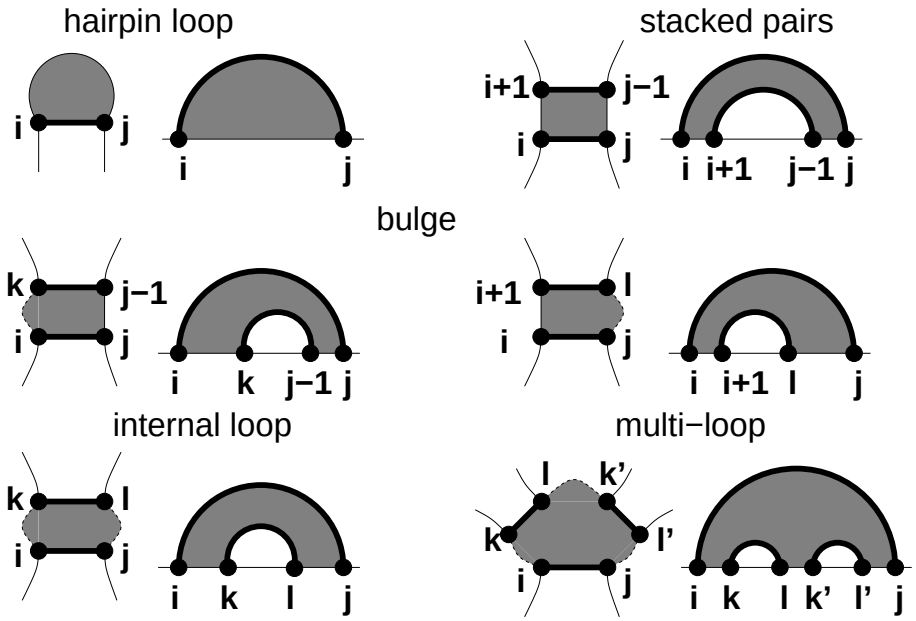


Figure 5.2: Classification of the RNA secondary structure elements hairpin loops, stems, bulges, internal loops and multi-loops. Each of them is shown in the fold and the diagrammatic representation. Thin lines represent the covalent backbone, thick lines the hydrogen bonds and dots the paired bases with their respective positions i, j, k or l in the sequence. For all diagrams it assumed that $i < l < k < j$ and additionally for multi-loops $k < l' < k' < j$. Multi-loops are only shown up to third order.

5.2 The pair-energy model

RNA folding algorithms rely on particular *energy models*. That is a function that assigns each structure of a fixed sequence $\mathbf{a}$ an energy $E : \chi_{\mathbf{a}} \to \mathbb{R}$.

A very simple model, the so called *pair-energy model* or *pair-matching model* [NPGK78, NJ80], involves contributions due to hydrogene bonds. The negative energy is given by the number of base-pair and only states that fullfil the following constraints are allowed:

(i) Only Watson-Crick pairs can be built.

(ii) Due to the bending rigidity of the RNA molecule it is impossible that two bases a_i and a_j close to each other in the primary structure can be paired, therefore we require a minimum distance, i.e. $j - i \geq h_{\text{min}}$, here I use $h_{\text{min}} = 2$ throughout.

This two conditions yield the energy function

$$E(\mathcal{C}; \mathbf{a}) = \sum_{(i,j)\in\mathcal{C}} \epsilon_{i,j},$$

with

$$\epsilon_{i,j} = \begin{cases} -e_{\text{pair}} & \text{if } a_i \text{ and } a_j \text{ are complementary and } j_k - i_k \geq h_{\min} \\ \infty & \text{otherwise} \end{cases}.$$

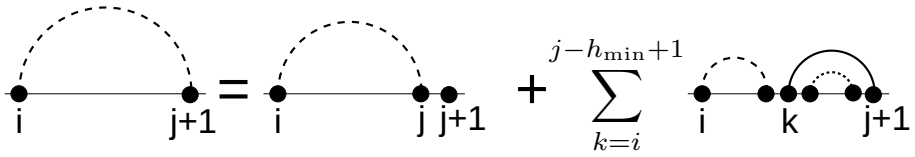


Figure 5.3: Diagrammatic representation of recursion relation Eq. (5.1). Dashed arcs represent partition functions of subsystems regardless whether connected bases are paired or not. The solid line represents a base pair.

The parameter $e_{\text{pair}} > 0$ defines the energy scale and is set to $e_{\text{pair}} = 1$ for the simple base-pair counting here[1].

Now, let us consider the canonical ensemble of all secondary structures for a fixed sequence at the temperature T. The partition function of this system is given by

$$Z = \sum_{\mathcal{C} \in \chi_{\mathbf{a}}} e^{-E(\mathcal{C};\mathbf{a})/T}$$

It is possible to compute Z using dynamic programming (i.e. transfer matrix) techniques [McC90]. This method is discussed now.

There are $L^2/2 - L/2$ possible subsequences $a_i \cdots a_j$ $(i < j)$ with corresponding partition functions $Z_{i,j}$. Since pseudo knots are also excluded, the hypothetically inserted pair $(k, j+1)$ induces two independent subsystems $a_i \cdots a_{k-1}$ and $a_{k+1} \cdots a_j$. Therefore the partition function $Z_{i,j+1}$ depends on all $Z_{k,l}$ $(i \leq k < l \leq j)$ and the subsequence $a_i \cdots a_j\, a_{j+1}$ only. One has to sum over the different cases of bond formation of the last position $j+1$. There are at most $j - i - h_{\min} + 2$ candidate pairs that connect the base at the position $j+1$ with any other base at position k in the subsequence. Due to the definition of the energy model positions with $j - k + 1 < h_{\min}$ and non-complementary bases are excluded.

Hence the partition function $Z_{i,j+1}$ can be written recursively

$$Z_{i,j+1} = Z_{i,j} + \sum_{k=i}^{j-h_{\min}+1} Z_{i,k-1} \cdot \mathrm{e}^{-\epsilon_{k,j+1}/T} \cdot Z_{k+1,j}. \tag{5.1}$$

The diagrammatic representation of Eq. (5.1) is shown in Fig. 5.3.

Starting with the boundary conditions $Z_{i,i} = 1$ and $Z_{i,i-1} = 1$, one can calculate $Z_{i,j}$ for increasing values of $j - i$, finally arriving at $i = 1$ and $j = L - 1$ which yields the full partition function $Z = Z_{1,L}$. Since the number of possible subsequences grows quadratic in the sequence length and the sum in Eq. (5.1) can be computed in linear time, the overall time complexity is of order L^3 and the required memory grows like L^2.

The partition functions $Z_{i,j}$ can be used to sample states from the canonical ensemble directly [Hig96]. Also direct sampling of ground-states with equal weights is possible [Har01] without using a temperature variable. The idea is quite similar to the stochastic backtracing method for sequence alignment. An algorithm is provided in Appendix A.2.

[1] In fact e_{pair} is an effective free energy and one would have to account for different bond energies of different bases pairs. On the other side, even with a more distinctive pair energy contribution this model is still too simple to predict realistic secondary structures. It is rather a powerful vehicle to study fundamental physical properties of secondary structures either analytically or numerically.

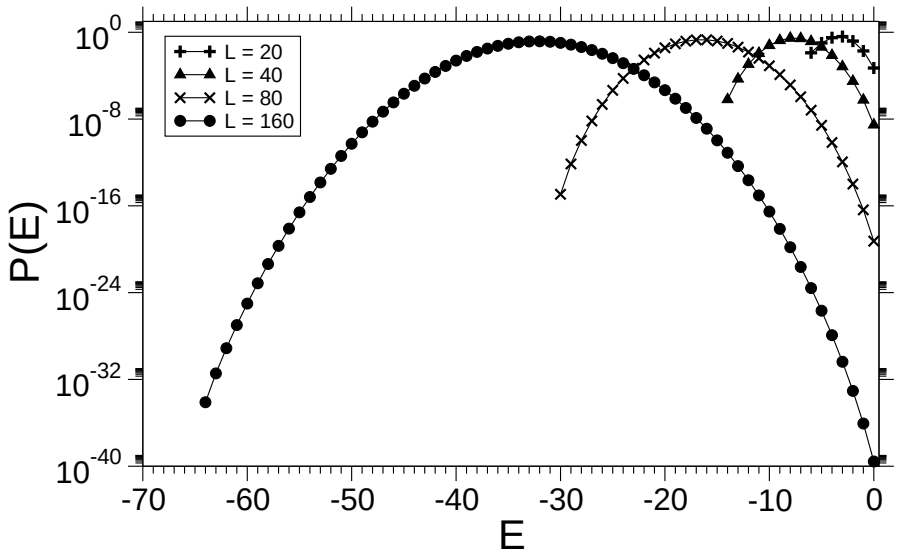


Figure 5.4: Normalized DOS of different randomly generated RNA sequences. Lines are guides to the eyes only.

With the same arguments as in Sec. 3.4 the dynamic programming algorithm for the partition function can easily be generalized to exact DOS calculations, here in $\mathcal{O}(L^5)$ time complexity. This is possible because the energy occurs in multiples of an energy "quantum" (the energy of a pair e_{pair}) and the DOS is obtained by an high temperature expansion in the parameter $z = \mathrm{e}^{-1/T}$. The numbers in Eq. (5.1) are replaced by polynomials in z, i.e. $\hat{Z}_{i,j}(z)$ instead of $Z_{i,j}$ and the full partition function is also a polynomial in z,

$$\hat{Z}(z) \equiv \hat{Z}_{1,L}(z) = \sum_n c_n z^n$$

with some coefficients c_n. The DOS can be obtained by re-substituting $z = \mathrm{e}^{-1/T}$ in $\hat{Z}(z)$ and noting that the energy can only occur as a multiple of $e_{\text{pair}} \equiv 1$. This implies that $E = -n$ and $g(-n) = c_n$.

In Fig. 5.4 the normalized DOS $\hat{g}(E)$ for four different randomly generated RNA sequences of lengths between 20 and 160 are illustrated. If one is interested in low lying excitations alone, for example in quantities such as

- the ground state energy E_0,
- the ground state degeneracy $g(E_0)$ or
- the microcanonical entropy difference $S = \log g(E_0)/g(E_0 + 1)$,

one may employ a truncated polynomial, where only the terms with the two largest degrees are considered. This allows one to compute these quantities in $\mathcal{O}(L^3)$ time complexity instead of $\mathcal{O}(L^5)$ for the full DOS.

It is also straightforward to modify the partition-function calculation given in Eq. (5.1) to determine various other thermodynamic quantities, such as the expectation value of the internal energy $\langle E \rangle_T$ or the specific heat in $\mathcal{O}(L^3)$ time complexity without computing the full DOS [McC90].

5.3 The free-energy model

Although the simple pair-energy model describes the folding process qualitatively it lacks in the description of natural RNA, because it involves only a single energy scale. Also the temperature, which enters in the partition function calculation discussed in the last section, can hardly be associated with a realistic temperature (for example the physiological temperature $37^\circ C$).

Hence, more sophisticated energy models have been introduced and much effort have been made to adjust the parameters in order to increase accuracy in secondary-structure prediction. Fortunately, efficient algorithms for RNA secondary structure prediction are not only available for the simple pair-energy model [dG68, NPGK78, NJ80], but also for more realistic models [ZS81], which, equipped with empirical free-energy parameters, are able to predict structures to an accuracy of $60-90\%$ in terms of correctly predicted base pairs [MSZT99]. Surprisingly these algorithms work without increasing the characteristic computational complexity of $\mathcal{O}(L^3)$, when one considers some biologically reasonable approximations.

Each structure is assigned a Gibbs free-energy[2] (or free enthalpy) $\Delta G : \chi_{\mathbf{a}} \to \mathbb{R}$, where each loop contribute a certain amount that depends on the type, size and composition of loops, in particular the terminal bases. The free-energy parameters had been determined experimentally (mainly via absorbance versus temperature melting curves [FKJ$^+$86, WTK$^+$94, MSZT99]) at the standard physiological temperature 37° and a given salt concentration. Next, they have been improved by comparison of predicted structures with those known from phylogenetic analysis [JTZ89]. The locality of the loop contributions are described by the so called nearest-neighbor model [XSB$^+$98]. Within that model the dependence of the free-energy contribution is assumed to depend only on few bases close to the boundaries of the loop. Stacked pairs consist of an enthalpic and entropic term, whereas other loops contribute entropically

$$\Delta G^{\text{pair}} = \Delta H - T\Delta S \quad \text{and} \quad \Delta G^{\text{loop}} = -T\Delta S.$$

Since these contributions depend on the position we write, in analogy of $\epsilon_{i,j}$ in Eq. (5.1), $\Delta G^{(1)}_{i,j}$ for the free-energy contribution of first order loops (hairpins) and $\Delta G^{(2)}_{i,j;k,l}$ for the one of second order loops.

The pairs (i,j) and (k,l) denote terminal pairs. Higher order loops are treated effectively, details are not presented here. The functions $\Delta G^{(1)}_{i,j}$ and $\Delta G^{(2)}_{i,j;k,l}$ contain nearly all essential free-energy parameters; there are hundreds of those [MSZT99].

The partition function of the full system is given by

$$Z = \sum_{\mathcal{C}\in\chi_{\mathbf{a}}} \exp\left[-\Delta G(\mathcal{C})/RT\right],$$

where R denotes the gas constant. The calculation of Z requires auxiliary partition functions similar as for the affine gap-cost sequence alignment algorithm Eq. (3.9). That are the partition functions of structures on the subsequences $a_i \dots a_j$ given that (i,j) are paired, denoted as $Z^{\text{pair}}_{i,j}$, and, as usual, the partition function of all structures on $a_i \dots a_j$ regardless whether (i,j) is paired or not, $Z_{i,j}$. Hence $Z_{1,L}$ equals the full partition function.

[2] $G = U + pV - TS$

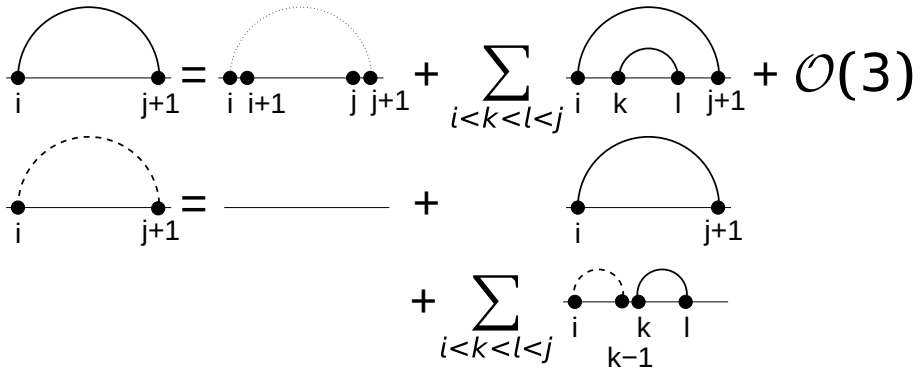


Figure 5.5: Diagrammatic representation of Eq. (5.2). Dashed arcs represent the unconstrained partition functions $Z_{i,j}$, and solid arcs the partition functions $Z^{\text{pair}}_{i,j}$. The dotted corresponds to the Boltzmann factor of the first order loop $e^{-\Delta G^{(1)}_{i,j+1}/RT}$. The partition function of the empty structure translates to an horizontal line. The symbol "$\mathcal{O}(3)$" represents the effective treatment of multiloops

Without restrictions the transfermatrix calculation of $Z_{1,L}$ can be done in $\mathcal{O}(L^4)$ time complexity via iterating the equation

$$\begin{aligned} Z^{\text{pair}}_{i,j+1} &= e^{-\Delta G^{(1)}_{i,j+1}/RT} + \sum_{i<k<l<j+1} e^{-\Delta G^{(2)}_{i,j;k,l}/RT} \cdot Z^{\text{pair}}_{k,l} + \mathcal{O}(3) \\ Z_{i,j+1} &= 1 + Z^{\text{pair}}_{i,j+1} + \sum_{i<k<l<j+1} Z_{i,k-1} \cdot Z^{\text{pair}}_{k,l}, \end{aligned} \tag{5.2}$$

whose diagrammatic representation is shown in Fig. 5.5. The symbol "$\mathcal{O}(3)$" denotes the effective treatment of multiloops.

The unconstrained partition function $Z_{i,j+1}$ involve contributions from the empty structure, from the possibility to build the pair $(i, j+1)$ and a sum over all possible pairings of (k, l) on the subsequence $a_{i+1} \dots a_j$. In order to achieve a time complexity of $\mathcal{O}(L^3)$, loops are usually restricted in size and hence the double sums in Eq. (5.2) are computable in linear time.

The prediction of the "optimal" secondary structure is based on minimizing the free-energy $\Delta G = -T \log Z$. Corresponding minimization algorithms can easily be obtained from Eq. (5.2) by replacing summations by minimizations and multiplications by additions. The optimal structure is then obtained by a backtracing procedure[CB05, RDM98].

Different implementations of this model have been published, two popular alternatives are the program `mfold`, maintained by Michael Zuker [Zuk03] and the `vienna` package [HFS+94], maintained by Ivo Hofacker. Here, I have used the `vienna` package, because it offers a well documented C interface. Since both programs are based on the same algorithms and parameters, I would not expect any difference in the main results in Chapter 6. The package contains a bundle of software for different purposes, including

- the prediction of the minimum free-energy structure and base pair probabilities
- the computation of partition functions and the specific heat curve,

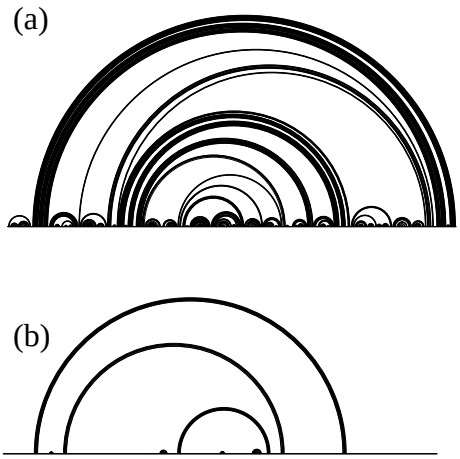


Figure 5.6: RNA secondary structures of the same molecule at different temperatures. The sequence was taken from the rRNA database "SivlaDB" [PQK$^+$07] (16S ribosomal RNA of Escherichia coli [BDSN81], accession number: `V00348`). The structure was predicted with the `vienna` package [HFS$^+$94] by minimizing the free energy at the physiological temperature $T = 37\,^\circ\mathrm{C}$ (a) and $T = 100\,^\circ\mathrm{C}(b)$.

- suboptimal folding,
- inverse folding, i.e. RNA design and
- diverse analysis tools,

only to mention a few.

Let us consider biological examples of secondary structures on natural rRNA sequences. Similar as in the case of local sequence alignment (Sec. 3.6), each sequence features its own thermodynamic properties. The temperature and salt concentration are control parameters that determine the structure. It is assumed that natural conformation of most RNA molecules are the one that have a minimum free-energy and, hence, when the temperature is decreased slowly enough the folding process is described by equilibrium thermodynamics [TB99].

With decreasing temperature, the enthalpy dominates the entropy more and more, which means that more hydrogene bonds are built. The formation of these bonds also decreases the entropy for further loop formation. Predicted minimum free-energy structures at two temperatures (at the physiological temperature $T \approx 37\,^\circ\mathrm{C}$ and above) are shown in Fig. 5.6. The high temperature behavior is characterized by large loops and only a few hydrogene bonds. Chapter 6 describes an analysis of destabilizing / stabilizing effects due to extremely rare sequences.

Of importance is, of course, the specific heat as a function of temperature, as shown in Fig. 5.7 for different rRNA molecules. These "melting curves" depend strongly on the sequence and may exhibit several peaks corresponding to the formation of certain loops.

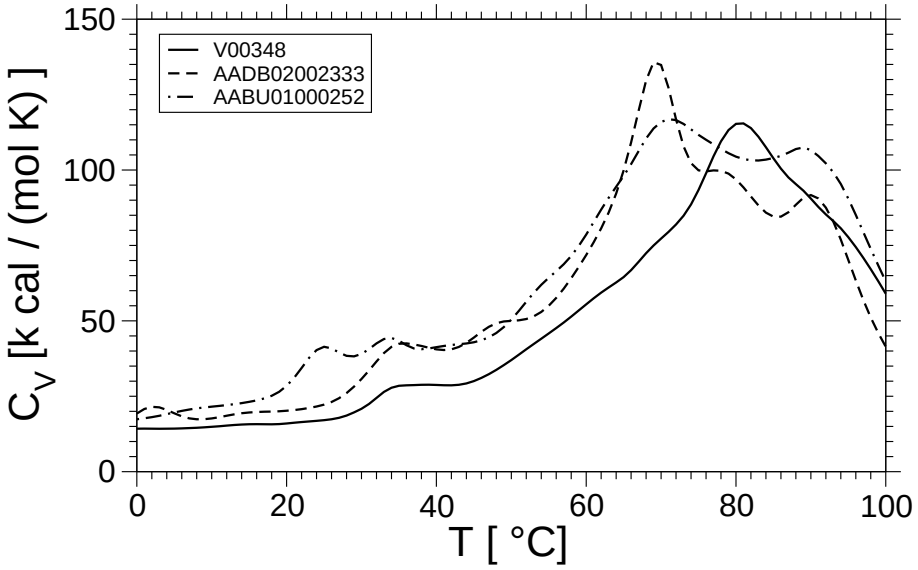


Figure 5.7: The melting curve (specific heat vs. temperature) of the RNA secondary structure of rRNA molecules of the organisms Escherichia coli [BDSN81] (accession number: `V00348`), Drosophila melanogaster (fruit fly, accession number: `AABU01000252`) and Homo Sapiens (accession number: `AADB02002333`) taken from the data base "SivlaDB" [PQK+07]. The specific heat was computed with the `vienna` package.

5.4 The molten-glass transition

Physically, RNA secondary structures can be seen as a disordered system with a rugged free-energy landscape [Hig96]. In this context the sequence is considered as a an random object and each particular realization induces a Gibbs-ensemble of possible structures.

The low-temperature properties of the simple pair-energy model, which was introduced in Sec. 5.2, is suitable to understand the low-temperature properties of RNA qualitatively. The model exhibits a static phase transition at a finite temperature [PPRT00, BH02b, BH02a, FKM02, LW06, HT06] between a "molten" high-temperature phase and a "glassy" low-temperature phase. In the molten phase the disorder does not play a role, i.e., in the thermodynamic limit, the structure of the phase space does not depend on the realization of the disorder.

The low-temperature phase is characterized by large sample-to-sample fluctuations that do not vanish as the sequence length tends to infinity (i.e. it lacks self averaging). This phenomenon also occurs in other disordered systems such as spin glasses [You98].

One approach to determine the critical temperature is based on the Parisi order paramater, i.e. the overlap between different structures $q(\mathcal{C}_1, \mathcal{C}_2)$. Let $\sum_{k \notin \mathcal{C}}$ denote the sum over unpaired bases in the structure $\mathcal{C}$. The Parisi order parameter for RNA secondary structures has been defined [Hig96, PPRT00] as

$$q(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{L} \left[2 \sum_{(i,j) \in \mathcal{C}_1} \sum_{(k,l) \in \mathcal{C}_2} \delta_{i,k} \delta_{j,l} + \sum_{i \notin \mathcal{C}_1} \sum_{k \notin \mathcal{C}_2} \delta_{i,k} \right], \tag{5.3}$$

where the first double sums is taken over all pairs and the second one over all unpaired bases in both structures. Note that $q(\mathcal{C}_1, \mathcal{C}_1) = 1$, but $\mathcal{C}_1 \cap \mathcal{C}_2 = \{\}$ does not imply $q(\mathcal{C}_1, \mathcal{C}_2) = 0$ in general.

In the low temperature phase the distribution of q for a given sequence is broad and it fluctuates from sample to sample, which indicates a complex behaviour [PPRT00].

The relationship between these aspects of complex statics and the dynamics of Monte Carlo algorithms is to be discussed in Chapter 7.

Chapter 6

Minimum-free-energy distribution of RNA secondary structures

Beside the simplification of the energy model as pair counting, also the consideration of RNA sequences as purely random objects is not valid for natural biological sequences.

It is established that, in most cases, natural sequences have a lower minimum free energy than random sequences drawn from ensembles with similar statistical properties as the natural one (for example the same composition) [Hig93, WK99, SD99, CB05]. Higgs has illustrated that natural tRNA sequences have a lower minimum free energy than purely random ones with the same composition [Hig93] and also that the probability to find the minimum free energy (among all states) is larger for realistic sequences at realistic physiological temperatures. In the case of mRNA this issue has been discussed controversially. Where Seffens and Digby found evidence that natural mRNA are more stable than random ones [SD99], Workman and Krogh found contrary results [WK99]. This could be explained by the dependency of free-energy contributions in the local neighborhood of a stacked base pair [WK99, CB05]. Workman and Krogh included this kind of local dependency explicitly in their sequence model by a description of the sequence as a first order Markov process. In other words, it turned out that the results depend on different definitions of the random sequence ensembles, i.e. different shuffling procedures.

Another important observation [Hig93, SD99, MSZT99] is that the minimum free energy is strongly correlated with the $C + G$ content of the sequence.

The evidence that a natural RNA sequence has a lower free energy than random ones is measured by the so called z-score of the minimum free energy of the natural sequence against the random ensemble. This quantity measures the distance of the observed free energy value $G_{\min}$ from the mean μ of the free-energy distribution over an ensemble in terms of standard deviations,

$$\text{z-score} := \frac{G_{\min} - \mu}{\sigma}.$$

The free-energy distribution is determined by a randomization of the natural sequence [Hig93, SD99, WK99, CB05], according to a random sequence model.

Here, we approach the problem from a different direction. Instead of comparing

natural RNA against a reference ensemble characterized by the statistical properties (e.g. the composition) we keep the (normalized) free energy fixed and compare entropic properties of natural RNA sequences against those of microcanonical sequence ensembles. For example, one may ask how likely natural sequences are modeled by a i.i.d. sequences with uniform composition (each letter occurs with equal probability) constrained that the random and the natural sequences have the same minimum free-energy. Since each sequence in a microcanonical ensembles occurs equally likely, one may check how likely a natural sequence is compatible with a maximum entropy principle. To address this problem I adopted the methodology that has been applied to the score statistics of local sequence alignment (Chapter 4,[Har02, WBH07, WHRH]). Having access to the tail of the free-energy distributions allows one to probe properties of large deviations (in the sequence space) and relate those to the corresponding minimum free energy. These properties are then compared with properties of natural rRNA sequences taken from a current database.

This chapter is organized as follows. In Sec. 6.1 the i.i.d. sequence model and a comparison method are introduced. The generalized ensemble methods that are used here were discussed in detail in Chapter 2 particularly in Sec. 2.7. Sec. 6.2 treats some special issues that are important here. The main results including the comparison between random and natural sequences are presented in Sec. 6.3. A general discussion in Sec. 6.4 completes this chapter.

6.1 Sequence models

The sequence space of RNA is the set of all possible sequences of length L over the alphabet $\Sigma = \{A, C, G, U\}$. This space will be denoted as Σ^L.

For random sequences I have chosen a simple model of i.i.d. (identically and independent distributed) sequences. That means each letter $a \in \Sigma$ occurs with a fixed probability f_a ($f_a = 1/|\Sigma| = 1/4 \ \ \forall a \in \Sigma$ here) independent of the other letters and of the position in the sequence. Hence the sequence $\mathbf{a}$ occurs with probability

$$p(\mathbf{a}) = p(a_1, \dots a_L) = \prod_{i=1}^{L} f_{a_i} = \frac{1}{|\Sigma|^L}.$$

Later on, we shall compare composition of natural RNA sequences against microcanonical averaged compositions or uniform compositions. For this purpose I used the Bhattacharyya distance measure (BDM) [Bha43], which is defined as

$$B(p||q) = \sum_i \sqrt{p(i)} \cdot \sqrt{q(i)}. \tag{6.1}$$

The BDM, measures the "distance" between the distributions p and q and fulfills the properties

- $0 \leq B(p||q) \leq 1$,
- $B(p||q) = 1$, if and only if $p = q$, and
- $B(p||q) = B(q||p)$.

This allows one to measure the distance of an observed normalized composition $\hat{f}(a) = \frac{1}{L}\sum_{j=1}^{L}\delta_{a,a_j}$ of a given sequence $\mathbf{a} = a_1 \dots a_L$ to a "null" distribution $f_0(a)$

$$\begin{aligned}\hat{B} &= B(\hat{f}||f_0) \\ &= \sum_{a=1}^{|\Sigma|}\sqrt{\hat{f}(a)}\sqrt{f_0(a)}.\end{aligned}$$

The BDM alone does not provide a statistical interpretation in the spirit of test theory, because it depends on the sample size (here the sequence length) and the number of bins (here the alphabet size). Under the assumption that $\hat{f}$ is described by f_0, the BDM deviates from 1 more likely for short sequences than for longer ones. A statistical interpretation becomes possible, if one assess a p-value to an observed BDM $\hat{B}$. This issue together with an algorithm to compute the p-value is discussed in Appendix A.4.

6.2 Simulation method

The structure of the problem is very similar to that under consideration in Chapter 4. Firstly, the space of the realizations of the disorder are sequences over finite alphabets (the 20 letter amino acid alphabet in Chapter 4). Secondly, the minimization of the free energy (or maximization of the similarity score, respectively) is of polynomial running time and based on transfer-matrix calculations in both cases (the Smith-Waterman algorithm [SW81] in Chapter 4).

Instead of the optimal-score distribution, the quantities of interest in this chapter is the distribution of the minimum-free-energy distribution for the biological relevant model that was described in Sec. 5.3,

$$P(G_{\min}) = \sum_{\mathbf{a}\in\Sigma^L} p(\mathbf{a})\,\delta_{G_{\min},G_{\min}(\mathbf{a})}.$$

The construction of the Markov chain for the i.i.d. letter composition can be directly adopted from Sec. 4.2, in particular, the five moves that have been described there. Throughout this chapter I have used the generalized ensemble Metropolis algorithm in combination with the Wang-Landau scheme. This methodology is discussed in Sec. 2.7.

Since the weights depend on floating point numbers, I made use of discretized weights. The bin size was chosen as $1\mathrm{kcal}/\mathrm{mol}$ (the standard physical unit that is used in the `vienna` package).

Given n sampled sequences $\{\mathbf{a}_1, \dots \mathbf{a}_n\}$ and corresponding free-energy values $G^i_{\min} \equiv G_{\min}(\mathbf{a}_i)$, expectation values are approximated by

$$\langle A\rangle_p \approx \frac{1}{z}\sum_{i=1}^{n}\frac{A(\mathbf{a}_i)}{w([G^i_{\min}])},$$

where $[G_{\min}]$ denotes rounding to the closest integer and z is the normalization constant $z = \sum_{i=1}^{n}\frac{1}{w([G^i_{\min}])}$.

I used data sampled from the generalized ensemble to approximate microcanonical averages

$$\langle A\rangle_{G_{\min}} \approx \frac{1}{z'}\sum_{i=1}^{n}\frac{A(\mathbf{a}_i)}{w([G^i_{\min}])}\begin{cases}1 & \text{if } \; G_{\min}-\Delta \le G^i_{\min} < G_{\min}+\Delta \\ 0 & \text{otherwise}\end{cases}$$

with some bin-size Δ and a normalization constant z'. This was used to determine the compositions as a function of the free energy.

Alternatively, in order to avoid binning effects and to obtain better statistics, a made use of canonical-like ensembles to determine thermodynamic quantities as a function of the minimum free energy. For this purpose I introduced an inverse "temperature" Θ. By choosing different values of Θ one may probe the entire free-energy range that has been sampled ($\Theta < 0$ probes the right tail above the mean and $\Theta > 0$ the left one). Expectation values in this ensemble are defined as

$$\langle A \rangle_\Theta \approx \frac{1}{z_\Theta} \sum_{i=1}^{n} \frac{A(\mathbf{a}_i)}{w([G^i_{\min}])} \cdot e^{-\Theta G^i_{\min}}, \tag{6.2}$$

with $z_\Theta = \sum_{i=1}^{n} e^{-\Theta G^i_{\min}}/w([G^i_{\min}])$. As a first step, the temperature is tuned such that the expectation value of the free energy equals a desired value $G_{\min} = \langle G^i_{\min} \rangle_\Theta$ and then the "canonical" average of the quantity of interest $\langle A \rangle_\Theta$ is computed and related to $\langle G_{\min} \rangle_\Theta$ via Θ, for the sake of simplicity denoted as $A(G_{\min})$ below.

I sampled the minimum-free-energy distributions for different sequence lengths between $L = 40$ and $L = 160$ and different temperatures ($T = -100, 0, 37\,^\circ\mathrm{C}$). For the largest system 4.5×10^7 Monte Carlo steps for the production run in the generalized ensemble were performed, yielding to $24,000$ "uncorrelated" sequences. The correlation time was determined through the autocorrelation function as described in Sec. 2.5.2.

6.3 The minimum-free-energy distributions

In this section, the resulting distributions are discussed. Before presenting the data of the rare event simulation, first the scaling properties of the mean, standard deviation and the skewness of the distributions are discussed. For this purpose I used simple sampling (see Sec. 2.1) for considerable larger system sizes (up to $L = 1280$).

Informal spoken, the skewness measures how much probability mass is located at either side of the mean. A positive (negative) value indicates the distribution to have more mass on the right (left) tail. It is defined as

$$\text{skewness} := \frac{\mu_3}{\sigma^3},$$

where $\mu_3 = \left\langle (\langle X - \langle X \rangle)^3 \right\rangle$ is the third moment about the mean and $\sigma = \sqrt{\langle (X - \langle X \rangle)^2 \rangle}$ the width of the distribution. The sample size varied between $10,000$ for the smallest ($L = 40$) and 1300 for the largest system. The result is shown in Fig. 6.1. The first moments and the widths scale in analogy to previous studies [SD99] as

$$\langle G_{\min} \rangle_L = c_1 \cdot L + c_0 \quad \text{and} \quad \sigma[G_{\min}]_L = d \cdot L^\nu \tag{6.3}$$

The resulting fit-parameters of a least-χ^2 fit are summarized in Tab. 6.1.

The small skewness differs from other models with quenched disorder and long range interaction. For example, the long-range spin-glass exhibits ground state energy distribution that can be described by a modified Gumbel distribution [KKH06], i.e. a skewed distribution. Also for the ground-state-energy distribution of the pair-energy model introduced in Sec. 5.2, I found a different behavior (results not shown here). For this model I found positive skewed distributions. For small sequences, the skewness

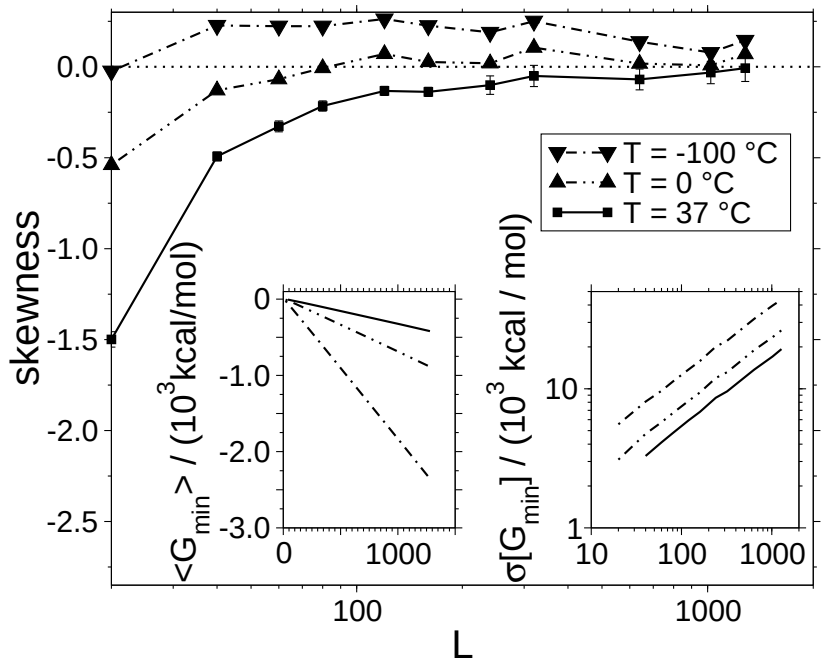


Figure 6.1: Skewness of the minimum-free-energy distribution over an random ensemble of i.i.d. sequences for different temperatures as a function of sequence length. Insets: scaling of the first moments and widths of these distributions with sequence length.

is much smaller, even negative, for human-body temperature. In all cases the skewness approaches 0 for large system sizes, which means that the distributions are essential symmetric in the high probability region. This can also be seen in the inset of Fig. 6.2, where the unscaled free-energy distributions for different temperatures are shown.

The main plot of Fig. 6.2 displays the distributions obtained by the generalized ensemble simulation in a logarithmic scale. The shape of the distributions at different temperatures differ slightly. Interestingly the one for lower temperature seems to be more symmetric, which is again in contrast to other models like the distribution of finite-temperature alignment that is discussed in Sec. 4.6.

In order to better understand the finite-size effects, the rescaled distributions for different system sizes and $T = 37^\circ$ are shown in Fig. 6.3. For large probabilities and

	c_0	c_1	d	ν
T=37 °C, $B = 1$	6.7(4)	0.324(2)	0.581(1)	0.382(4)
T=37 °C	8.9(4)	0.331(2)	0.51(1)	0.511(5)
T=0 °C	10.6(5)	0.691(4)	0.75(1)	0.498(3)
T=−100 °C	17.8(7)	1.842(5)	1.29(2)	0.494(3)

Table 6.1: Fit parameters of a least square fit of the mean and standard deviation of the minimum free-energy distributions to the functional form Eq. (6.3).

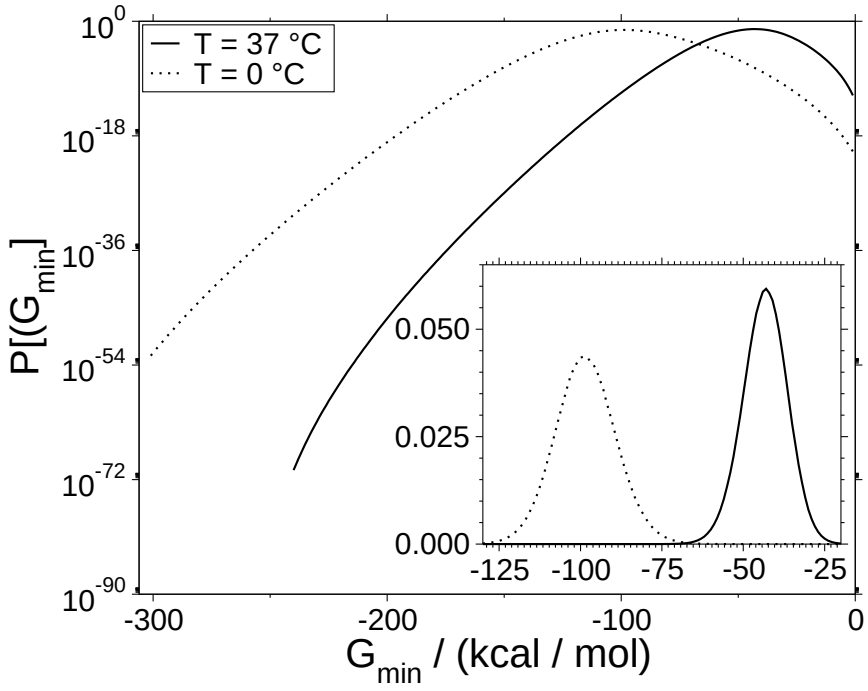


Figure 6.2: Raw minimum-free-energy distributions at different temperatures for the largest system $L = 160$.

for the short tail the distributions collapse quite well. In the long tail some effects show up. Long sequences seem to have a given rescaled free energy less likely than short sequences (for intermediate values of the rescaled free energy $(G_{\min} - \mu)/\sigma$).

6.3.1 Entropy and thermodynamics of large deviations

For the pair-energy model one observes (not shown here), that the sequence composition is uniform in the left tail and highly non-uniform in the far right tail. This can be understood by entropic arguments: In order to achieve a low energy the sequence requires to have many complementary bases. Ideally the second half of the sequence consists of complementary partners of the first one in the same linear order. In this case the ground-state is just a single stack of size $L/2$ (neglecting the condition that only bases with a larger distance than $h_{\min}$ can be paired). Such sequences exhibit an uniform composition, because one may choose the letters of the first half freely. In contrast, for a large ground-state energy, the sequence composition requires a huge amount of non-complementary bases, because the presence of a certain letter requires its complementary partner to occur rarely in the sequence.

In the same spirit, I analyzed the sequence ensembles that are biased towards very rare events of the free-energy distribution. Here, in contrast to the simplified pair-energy model, the observed letter distributions were non-uniform in both tails, which is shown in the bottom of Fig. 6.4. Also in Fig. 6.4 the functional dependence of $B(\hat{f}||f_0)$ with $f_0(a) = 1/|\Sigma| \;\; \forall a \in \Sigma$ on $G_{\min}$ is shown. That means for each sample $\mathbf{a}_i$ the empirical composition $\hat{f}^i$ and the corresponding value of the BDM $B^i \equiv B(\hat{f}^i||f_0)$

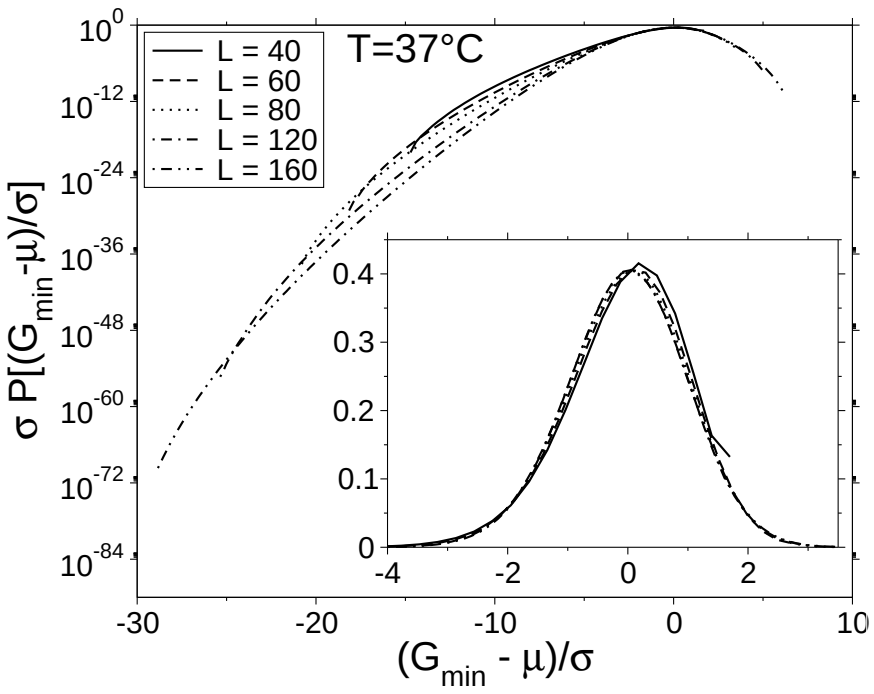


Figure 6.3: Rescaled minimum-free-energy distributions at $T = 37°C$ for different system sizes.

was estimated. Then the canonical averages for different Θ's were determined using Eq. (6.2) and identified with $G_{\min}$, as explained in Sec. 6.2.

Close to the mean of the distribution the value B is also close to 1 as would be also expected from simple sampling. Far in the left tail the value shrinks, what is also supported by the form of the histograms that are shown in the bottom of the figure. In the right tail also non-uniform compositions are observed, implying B to deviate from 1.

The plots in Fig. 6.4 are labeled with the medians of the p-values of a BDM test of the observed microcanonical sequence ensembles (depending on $G_{\min}$) against an uniform letter composition (see Appendix A.4).

Note, that to determine the histograms and the p-values I used binned free-energy intervals instead of the reweighting procedure. For that purpose the free energy range was divided into 50 bins for the largest system $L = 160$.

Sequences at the left end of the distribution essentially only consist of the bases C and G, which form three hydrogen bonds. The resulting structures are very stable [Hig93, SD99, MSZT99].

The composition in the right tail seems to be unexpected at the first glance, in particular as it not only describes the average composition, but it also turned out that individual sequences in this region have a similar empirical letter frequency. Even though there are many $A - U$ Watson-Crick pairs available, the minimum free energy is relatively large. This is so because a loop needs to be closed by a stable pair, ideally by $C - G$.

Additionally, the presence of Cs without the complementary partner G seems to

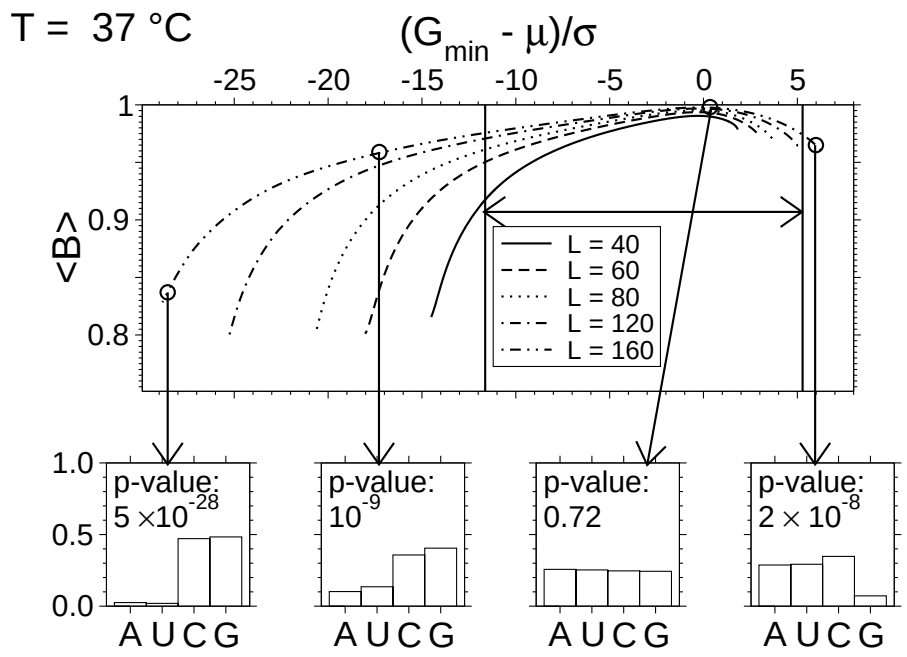


Figure 6.4: top: Observed BDM as a function of the rescaled minimum free energy. Non-uniform compositions are found in both tails. Vertical lines indicate the rescaled minimum-free-energy range of the selection of natural rRNA sequences (see Sec. 6.3.2) bottom: Histograms of observed compositions in different bins, very far and far from the mean on the left side, close to the mean and far in the right tail. The medians of the corresponding p-values for the BDM-test (against a perfectly uniform composition) are written in the plots of the histograms.

destabilize the structure, which can be supported by the following simple computer experiment on a sequence of length $L = 160$. First the sequence is initialized as $A^{L/2}U^{L/2}$, yielding a low minimum free-energy structure ($G_{\min} = -63.50\mathrm{kcal/mol}$) consisting of a single large stack. Then the sequence is modified by randomly replacing letters with Cs. The minimum free energy increases rapidly with the concentration of C's and reaches $G_{\min} = 0$, when approximately every third letter is modified. On the other side, when repeating the experiment by replacing the letters with G instead of C a much higher fraction of replacements (approximately 70%) is neccessary in order to achieve $G_{\min} = 0$.

By looking in the standard free-energy reference material, which was summarized by Mathews et.al. [MSZT99], this effect can be explained by penalty terms to the overall free energy for certain unstable secondary structure motives. Noticeable are so called "olgio-C loops" and "tandem mismatches" (see Table 6. and Table 11. in ref. [MSZT99]). Olgio-C loops are hairpin-loops, in which all unpaired bases are C. Tandem mismatches are internal loops with two unpaired bases on each strand. Free-energy contributions of loops of this kind have different values depending on the types of the mismatches (unpaired letters) and on closing base pairs. Some combinations have negative contributions others have positive penalties. Cases, where tandem mis-

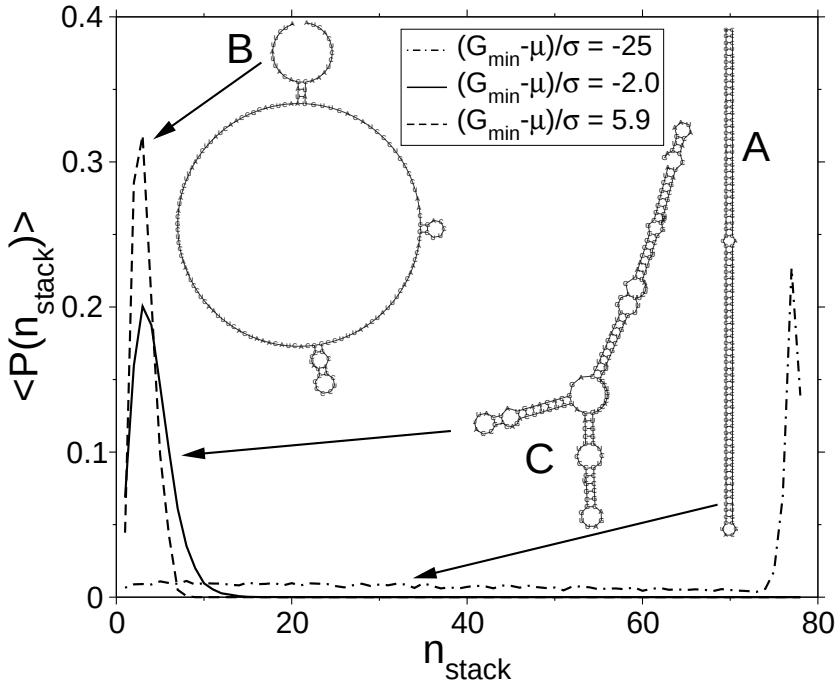


Figure 6.5: Reweighted stack-size distribution as a function of $G_{\min}$ for $L = 160$ and typical structures in the generalized ensemble.

matches are closed by $A-U$ pairs and that contain $C-A$, $C-U$ or $C-C$ mismatches are penalized most. $A+U$ rich sequences that are "dotted" with C are entropically more favorable than sequences that contain only few complementary letters, which is the condition to achieve a large ground-state energy in the pair-energy model.

The thermodynamics of large deviations can be studied by looking not only at the sequences and values of the free energy but also at properties of the minimum-free-energy structures, which are also reported by the program `RNAfold`. Fontana et. al. [FKSS93] studied various of such quantities using simple sampling of random RNA sequences and compared the statistics of this ensembles with natural RNA sequences. One quantity that was considered in [FKSS93] is the distribution of stack sizes over the ensemble of minimum free-energy structures, which is adopted here for the biased ensembles.

Three typical structures that occur in the biased sequence ensemble are shown in Fig. 6.5. The underlying sequence of structure A has a typical $C+G$ rich composition, which occurs in the left tail of the minimum free-energy distribution. Large stabilizing stacks are characteristic for those sequences. Although these structure are most stable, from the biological point of view, they are not very interesting for lack of important structural elements. The sequence with B as minimum-free-energy structure was drawn from the rare event tail on the right side and consists of large loops, that are usually very unstable. More attractive is structure C, which has a free-energy of 2.0 standard deviations below the mean of the minimum-free-energy distribution.

Reweighted stack-size distributions (based on the method described in Sec. 2.6) for

three values of the minimum free-energy is also shown in Fig. 6.5. In the ensemble of large minimum free energies only short stacks occur. For those sequences that have an extremely low minimum free energy, stack sizes on all length scales occur equally likely. Additionally a strong peak for stack sizes that are of the order of the half of the sequence length is observed. This reflects the observation of structure A, where a large stack is interrupted by a small internal loop. Interestingly, the difference between the biological interesting free-energy range (slightly below the mean) and the extreme unstable region is not significant. However deviations up to $n_{\text{stack}} = 15$ become not as unlikely as for those sequences from the far right tail. The loop-size distribution (note shown here) seems to be a better description in order to characterize differences between the right tail and sequences from the left tail in an intermediate probability range, whereas the stack-size distribution distinguishes better very rare events from the left tail and typical sequences.

The mean stack size and the width of the stack-size distribution as function of the $G_{\min}$ is shown in the upper row of Fig. 6.7. The left plots indicate that only a small fraction of sequences have minimum free-energy structures that consist of a single stack in the order of the sequence length. Fontana et.al. [FKSS93] observed that the mean stack size converges to a length independent value of approximately 3 base pairs. By studying the width of the stack size distribution one also learns that the greatest variety of stack lengths occurs in very rare sequences.

Both, the composition of the sequences and the stack-size distribution is discussed under the viewpoint of natural biological sequences in the following.

6.3.2 Comparison between random and natural RNA sequences

The distribution of random RNA sequences allows one to gain more insight in the question in which sense natural RNA sequences differ from random i.i.d. sequences. Under the viewpoint of rare events in the sequence space, we want to study thermodynamic and entropic aspects for natural ribosomal RNA sequences. For that purpose I randomly selected 2078 large subunit rRNA sequences from different species up to lengths $L = 1000$ from a current database [PQK$^+$07]. This kind of selection seems reasonable to me, because we are not interested in the biological details in this study. First of all, the minimum free energies of all sequences were obtained. In order to make the values of sequences of different lengths more comparable the free-energy values have been rescaled by subtracting the average value and then dividing by the width which are given by the scaling relations Eq. (6.3), using the fit parameters that are listed in Tab. 6.1. This rescaled free energy is the z-score with respect to the i.i.d. sequence ensemble for each sequence.

In a similar way as for the random sequence ensemble, I performed Bhattacharyya test against an uniform letter distribution $f_0(a) = 1/|\Sigma|$ (see Appendix A.4) for each individual sequence and I found the relationship between p-values of the test and rescaled free energy energy that is shown in Fig. 6.6(a).

Natural sequences, which have a minimum free energy below the mean down to about 5 standard deviations (a z-score of -5), exhibit intermediate and large p-values (dots in Fig. 6.6(a)). This indicates that there is some evidence that all letters of those sequences occur (more or less) equally frequently. However in this region there are also realizations with relatively small p-values (down to $\sim 10^{-9}$), but these values are large, in comparison to sequences that are more than 5 standard deviations below the mean, where p-values down to $\sim 10^{-26}$ occur. Since the distribution of p-values is broad, I included their medians as a function of the rescaled free energy (dashed line).

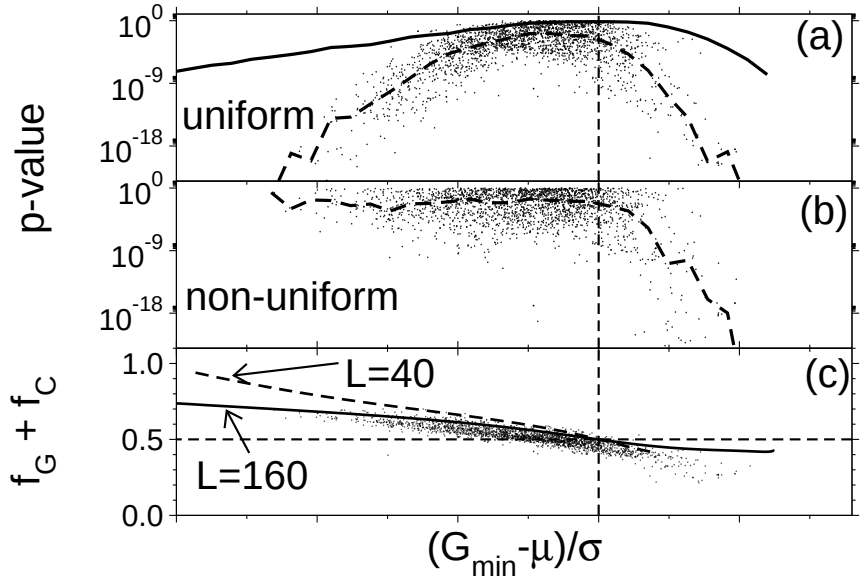


Figure 6.6: (a) Dependence of the p-values of Bhattacharyya tests against an uniform letter composition on the rescaled minimum free energy using natural rRNA sequences (dots). The dashed line marks the median of the p-value of natural rRNA sequences. The solid Line indicates the median of the p-value of the random sequence model in the generalized ensemble (L=160).
(b) The p-values of a Bhattacharyya test of the composition of the natural sequences against compositions that occur at the same rescaled minimum free-energies in the random sequence model. The dashed line indicates the median. The observed p-values are much smaller for large deviations towards small free-energies.
(c) The observed frequencies of $G + C$ as a function of minimum free energy.

Sequences above the mean are also very unlikely modeled by an uniform i.i.d. letter distribution, also indicated by very small p-values of the natural sequences. I compared this with the random sequence model by calculating the dependence of the median of the p-values as function of deviation of the free energy from the mean, which is shown as solid lines in Fig. 6.6(a). The qualitative behavior resembles those of natural sequences. Numerical deviations are probably due the fact that the largest system for the random-sequence model was $L = 160$, whereas the natural sequences are explicitly longer. Additionally, in agreement with previous observations [Hig93, SD99, WK99, MSZT99], one observes that most of the sequences are located below the mean.

For the free-energy model the stabilizing effect of $C - G$ pairs shows up in the clear correlation between free-energies and $C + G$ content, as shown in Fig. 6.6(c). In addition, the mean of the $C + G$ content of the random ensemble, shown by lines, tells us that the random model is suitable to explain low free energies due stabilizing $C - G$ pairs over a broad free energy range , as it is also observed in previous studies of natural RNA [Hig93, SD99, MSZT99]. In order to support this argument, the statistical

test of the sequence composition of the collection of natural sequences was repeated under the assumption of a different null hypothesis. That is the assumption, that the composition of a natural biological sequence is given by the mean composition of the random sequence model given the same rescaled minimum free energy (the histograms in Fig. 6.4 are 4 out of 50 different reference compositions).

At this point a few statements about the approximation of this test should be made. It is assumed that the composition is determined by the rescaled free energy alone and not on the sequence lengths (expect the scaling of the mean and the width). The sequence lengths are much larger for the collection of natural sequences. This assumption becomes reasonable, when comparing Fig. 6.4 with the scatter plot in Fig. 6.6. The rescaled free energies of the natural sequences (z-scores) range from -10 to 5. At least in the left tail, the finite size effects of the BDM are relatively small for lengths $L > 120$ in the biological relevant range of the rescaled free energies. The test was performed by using frequency tables, obtained by binning the minimum-free-energy range for $L = 160$ into 50 bins. These the empirical frequencies of the natural sequences were tested against those distributions. The corresponding p-values, see Fig. 6.6 (b), show a significant increasing of the values for low free energies in comparison to the original test against a perfectly uniform composition. On the other side, for large free energies no such observation could be made. Hence the assertion, that low free energies are strongly related to the $C + G$ content is further confirmed. Note that the free energy parameters rely on the nearest neighbor model [XSB$^+$98] (see Sec. 5.3). This means the $C + G$ content alone is only the leading effect to obtain a low free energy. This issue is discussed at the end of this chapter in Sec. 6.4.

Obviously, natural sequences with relatively large minimum free energies do not have compositions that are comparable with the random sequence model, where $A + U + C$ rich sequences are entropically favorable.

Regarding the stack sizes we find, in agreement with [FKSS93], no correlation between the value of the minimum free energy and mean and width of the stack-size distribution, as shown in the bottom in Fig. 6.7. The biological relevant free-energy region is above the sequence length dependent threshold value, where stacks sizes are of the order of the sequence length. Also the maximum of the width, where the greatest variety of stack sizes is expected, sits below this region.

In analogy, I also checked for a possible correlation between the minimum free energy and other thermodynamic quantities, for example a measure for the non-extensive character of the free energy [BH02b, BH02a, HT06]. That is the difference between free energy of the entire sequence and the sum of the free energies of the first and the second half of the sequence, when it is broken exactly in the middle, $\Delta G_{\min} = G_{\min}(r_1, \ldots r_L) - G_{\min}(r_1 \ldots, r_{L/2}) - G_{\min}(r_{L/2+1} \ldots, r_L)$. Again, $\Delta G_{\min}$ is largest for very low free energies, but in the biological relevant region it remains small and is not correlated to the free energy of natural sequences. Also the mean loop size of structures of natural sequences does not correlate with the minimum free energy (not shown).

6.4 Discussion and outlook

To my knowledge, I have presented the first Monte-Carlo study of the rare-event tail of the minimum-free-energy distribution of RNA secondary structures down to very small probabilities ($\approx 10^{-70}$). Large-deviation properties of random RNA sequences are discussed. I have illustrated how they can provide an additional classification of

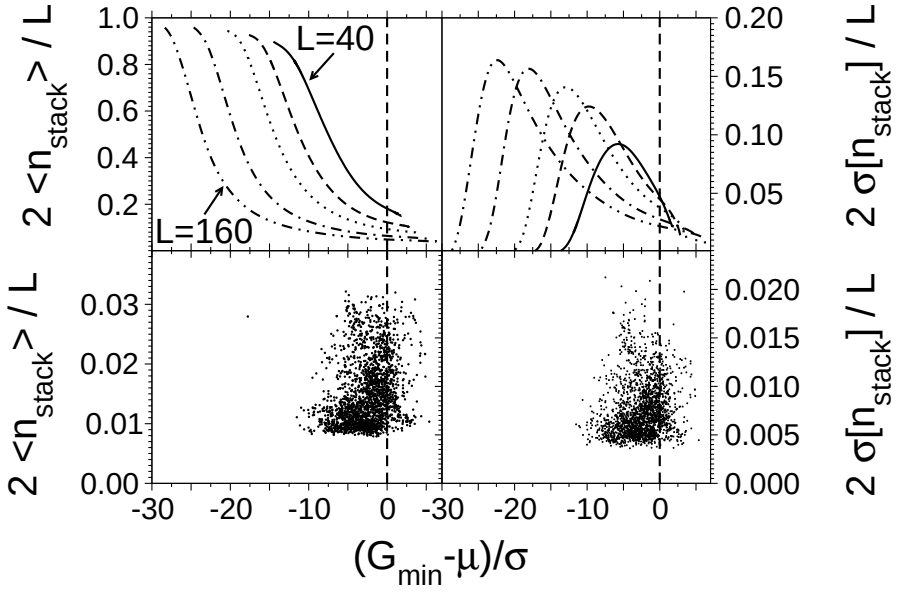


Figure 6.7: Top: mean and width of the stack-size distribution normalized to sequence length as a function of the rescaled minimum free energy.
Bottom: Scatter plot of the mean and width of the stack-size distributions of natural biological rRNA sequences (see Sec. 6.3.2)

"randomness" of natural RNA sequences.

Properties of large deviations can be explained by entropic and thermodynamic arguments (Sec. 6.3.1). As an entropic measure on the sequence level, the Bhattacharyya distance measure was used in order to discriminate observed sequences against the null-model with perfectly uniform composition, which is expected in the high probability region close to the mean. For the pair-energy model the composition is flat, even in the far left tail (low energies), whereas the composition deviates significantly from an uniform distribution in the right tail.

For the free-energy model non-uniform compositions occur in both tails. The leading effect for stable structures in the left tail (low free energies) is due to $G + C$ rich sequences. The destabilizing effect of $A + U + C$ rich sequences are responsible for very large free energies. These sequences are entropically favorable over such sequences that have many non-complementary bases, which would be the reason for a large ground-state energy within the pair-energy model.

In comparison to natural biological sequences, $G + C$ rich sequences also have the lowest minimum free energies, whereas many $A + U + C$ rich sequences are not found. One expects that all sequences in a microcanonical-like ensemble occur equally likely, due to the maximum entropy principle. From the statistical tests of the natural sequences against those in the microcanonical ensemble one may infer that natural sequences, constrained on low minimum free energies, are (more or less) compatible with entropy maximization. For large free energies this assumption seems not to be the

case.

There is a plenty of room for further studies of the z-score statistics from this microcanonical perspective. Even though, at least in the left tail, the p-value have increased significantly when going from the uniform null distribution to the one obtained from microcanonical ensemble, they are still relatively small. For example, the median of p-values changes from 10^{-20} to 0.02 for the free-energy bin $(G_{\min} - \mu)/\sigma \approx -10$. One may change the sequence model from i.i.d. to a first order Markov model, like in Ref. [WK99] or even more complicated shuffling procedures [CFKK05]. Possibly one would observe even larger p-values in the left tail. Eventually these models allow one to better describe the microcanonical sequences from the right tail as well. Similarly one may also modify the test statistics from the BDM to more complicated descriptions like Markov sequences instead of an i.i.d. model.

Chapter 7

Complex state spaces and glassy Monte Carlo dynamics

Beside its usefulness to model problems in molecular biophysics, RNA secondary structures are of fundamental interest to understand the relation between static and dynamic properties of disordered systems. The model exhibits quenched disorder, as described in Sec. 5.4, and has a complex low-energy landscape.

The static behavior of the model can be analyzed exactly using partition-function calculations for each single realization of the disorder. As shown in Chapter 5, the computation time grows only polynomially with the system size. This approach also allows one to generate secondary-structure configurations in equilibrium without rejection and exhibiting zero correlations between different configurations (see Appendix A.2).

There are only a few models that combine complex static properties and a feasible computational complexity. For example two-dimensional $\pm J$ Ising spin glasses and fully frustrated models can be solved exactly by transfer matrix methods [MB80] or by the program of Saul and Kardar in polynomial time [SK94]. On the other hand, no rejection-free equilibrium sampling method is known. Furthermore, two-dimensional spin glasses only have a phase transition at zero temperature [Vil77]. Better comparable to the RNA secondary structures is a model of directed polymers in random media [Mez90, Kar94], where direct sampling using transfer matrices of the partition function could be used and a non-trivial phase transition was detected. This model is related to the sequence-alignment problem, which was pointed out in Sec. 3.3.1.

Such complex energy landscapes usually feature also slow dynamical properties that can be seen in Monte Carlo or molecular dynamics simulations. The question, which static properties causes slow dynamics is often not easy to answer. The aim of the study in this chapter is to gain more insight into the relationship between static and dynamic complexity. Furthermore a study of this kind allows one to benchmark new Monte Carlo approaches.

In Chapter 2, different Monte Carlo approaches were described. These algorithms allow one to obtain the full DOS. This chapter treads two of them, namely the generalized ensemble method (see Sec. 2.7) in two variants and the ParQ algorithm (see Sec. 2.8.2). The variants of the generalized ensemble methods include a perfectly flat histogram ensemble and the optimized ensemble. In the first one the weights are chosen as $w(E) \propto 1/g(E)$ and the weights of the latter one uses weights that minimize the round trip time.

As in the Monte Carlo studies of the local alignment score statistics in Chapter 4 and the analysis of the minimum-free-energy distribution in Chapter 6, the disorder plays a key role in this study as well. A major difference is that in those investigations, I focused on rare events in the space of realizations of the disorder. Here, the realizations are quenched and taken from the typical regime that is accessible by simple sampling and the attention is drawn to rare events that occur in the state space of RNA secondary structures over fixed realizations. In this setup the disorder plays an important role when comparing rare-event properties of the Monte Carlo algorithms to static properties among different realizations.

In the following section some details on the Monte Carlo algorithms that are considered here are explained. After that, in Sec. 7.2, the convergence properties of a "hard" realization are examined. Sample-to-sample fluctuations and the relationship between structural and Monte Carlo complexity are to be discussed in Sec. 7.3. In Sec. 7.4 a possible performance enhancement by extended state spaces are considered and a final discussion in Sec. 7.5 closes this chapter.

7.1 Markov chain Monte Carlo sampling of secondary structures

The state space of a pseudo-knot-free secondary structures $\chi_{\mathbf{a}}$ on the sequence $\mathbf{a} \in \Sigma^L$ was defined in Def. 5.1.1. All algorithms here are based on a Markov chain on this space. In this chapter, only the pair-energy model is considered. In order to formulate this more precisely, one has to specify the update routine of the Metropolis algorithm, see Algorithm 2.2.1. In particular the neighborhood relationship $\mathcal{N}(\mathcal{C})$ of a structure $\mathcal{C} \in \chi_{\mathbf{a}}$ has to be made explicit. Because the major difference between the generalized ensemble methods and the ParQ algorithm is that the weights $w(E)$ that occur in the Metropolis algorithm are time-dependent in the latter approach, the following statements apply to both Monte Carlo approaches.

Formally, we define the neighborhood $\mathcal{N}(\mathcal{C})$ of the structure $\mathcal{C}$ as

Definition 7.1.1 *Let $\mathcal{C} \in \chi_{\mathbf{a}}$ a pseudo-knot-free secondary structure on $\mathbf{a}$. The local neighborhood $\mathcal{N}(\mathcal{C})$ is set of structures for which each $\mathcal{C}^* \in \mathcal{N}(\mathcal{C})$ fullfills the following properties:*

(i) it is pseudo-knot free, i.e. $\mathcal{C}^* \in \chi_{\mathbf{a}}$,

(ii) it is valid according to the pair-energy model, i.e. $E(\mathcal{C}^*; \mathbf{a}) < \infty$ and

(iii) the structures $\mathcal{C}^*$ and $\mathcal{C}$ differ in at most one pair, i.e. $|(\mathcal{C}^* \setminus \mathcal{C}) \cup (\mathcal{C} \setminus \mathcal{C}^*)| = 1$ or $\mathcal{C} = \mathcal{C}^*$.

The first two conditions define constraints of the model, whereas the third one defines the locality of the neighborhood. In the simplest implementation of a Monte Carlo move, one may draw one out of the $L(L-1)$ random pairs (i, j) with $i < j$. If the current structure $\mathcal{C}$ contains (i, j), i.e. $(i, j) \in \mathcal{C}$, the pair (i, j) is proposed to be removed from $\mathcal{C}$, $\mathcal{C}^* = \mathcal{C} \setminus (i, j)$. According to the pair-energy model the energy of the new structure $\mathcal{C}^*$ is given by

$$E(\mathcal{C}^*; \mathbf{a}) = E(\mathcal{C}; \mathbf{a}) + e_{\text{pair}} \equiv E(\mathcal{C}; \mathbf{a}) + 1.$$

Otherwise, if $(i,j) \notin \mathcal{C}$, one attempts to insert the pair into $\mathcal{C}$. If all three conditions above are valid[1], the energy of the proposed state is given by

$$E(\mathcal{C}^*;\mathbf{a}) = E(\mathcal{C};\mathbf{a}) - e_{\text{pair}} \equiv E(\mathcal{C};\mathbf{a}) - 1.$$

In the case that one condition is violated, the energy $E(\mathcal{C}^*;\mathbf{a})$ is set to ∞.

The moves are accepted with the usual Metropolis acceptance rate Eq. (2.4), i.e.

$$\alpha_{\mathcal{C},\mathcal{C}^*} = \min\left\{1, \frac{w(E(\mathcal{C}^*;\mathbf{a}))}{w(E(\mathcal{C};\mathbf{a})}\right\}, \tag{7.1}$$

where $w(\infty) = 0$. This means "forbidden" structures never occur in the simulation.

Since the average number of pairs increases linearly and the number of proposed pairs quadratic with the sequence length, most of the proposals will be rejected, especially close to the ground state. In order to avoid this, I have implemented a variant of the N-fold way [BKL75] (see Sec. 2.3) where only allowed structures are proposed. At the beginning of the simulation a list of all N_{possible} *possible pairs* $\{(i,j)\}$ is created. These pairs are compatible to the energy model $e(a_i,a_j) < \infty$, i.e. all (a_i, a_j) are Watson-Crick pairs and have sufficient distance h_{min} along the sequence. There are still $\mathcal{O}(L^2)$ of possible pairs.

At each stage of the simulation the set of allowed pairs is divided into three classes. The first class consists of the set of *active pairs*, i.e. that pairs that are currently contained in the secondary structure. The class of inactive pairs can be divided into two sub-classes. The first one contains all *allowed pairs*. That are those that can be inserted into the current structure without violating condition (i) in Def. 7.1.1. Those that would violate (i), but fullfill (ii) and (iii) belong to the class of *currently forbidden pairs*. Active pairs are associated with an energy change of $\Delta E = 1$, allowed pairs with $\Delta E = -1$ and forbidden pairs with $\Delta E = \infty$. The current number of members in each class given the structure $\mathcal{C}$ is denoted by $N(\mathcal{C},+1)$, $N(\mathcal{C},-1)$ and $N(\mathcal{C},0)$ for active, possible and forbidden pairs respectively.

A secondary structure is represented as a list of links to the static array of possible pairs. Then the simulation requires some bookkeeping of the lists for all three classes. For this purpose it makes sense to setup a list of cross-links between all pairs indicating incompatibility, i.e. for each pair a list of references to other pairs that lead to pseudo knots, when both are inserted at the same time.

The "forbidden attempts" are taken into account, by advancing the simulation-time clock sufficiently. This kind of dynamics combines a "rejection-free dynamics", as implemented in the n-fold way [BKL75] (see Sec. 2.3), with standard acceptance probabilities.

When performing the simulation, one has to account for the *waiting times* τ due to forbidden transitions in the local environment. This waiting times are determined with the concepts of the N-fold way described in Sec. 2.3: Let p be the probability that a forbidden pair is selected, given that the random walk sits in the state $\mathcal{C}$, i.e. $p = N(\mathcal{C},0)/N_{\text{possible}}$. Consequently the probability that the random walk selects a non-forbidden pair in the current state after m trials is given by Eq. (2.6),

$$p(m) = p^m(1-p)$$

and a random waiting time can be drawn from that distribution via Eq. (2.3).

[1] The condition (iii) always holds by construction

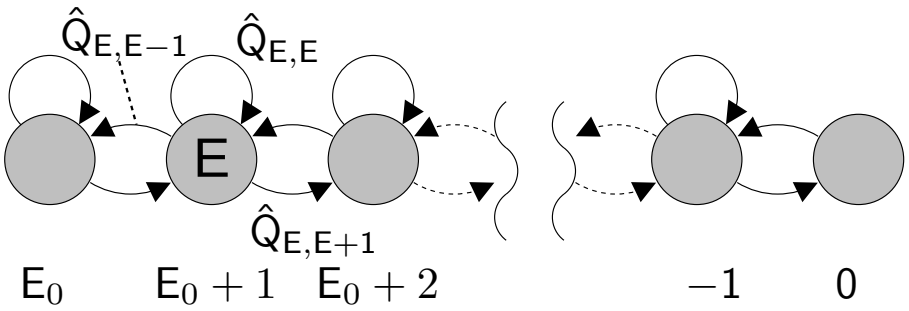


Figure 7.1: Macro states and possible transitions.

After the simulation-time clock has advanced by the random waiting time, a pair is selected from the set of active and allowed pairs with uniform probability, and the pair is flipped with a probability given by Eq. (7.1). Hence, if the flip is rejected, then the current structure persists. This reject-accept part of the algorithm completes one MC step.

For each Monte Carlo method M independent runs $j = 1, \ldots, M$ had been performed. During each simulation three quantities of interest were sampled. That are the *energy*, a random *energy change* $\Delta E = 0, \pm 1$ associated with each attempt (regardless if the step is accepted or not) and a random *waiting time*. This yields independent chains

$$(\hat{E}_1^{(j)}, \Delta\hat{E}_1^{(j)}, \hat{\tau}_1^{(j)}), \ldots, (\hat{E}_n^{(j)}, \Delta\hat{E}_n^{(j)}, \hat{\tau}_n^{(j)}).$$

The transitions proposed in all steps are counted in the tridiagonal matrix $\hat{W}_{E,E+\Delta E}$. The waiting times are allways added to the diagonal of this matrix. From $\hat{W}$ a stochastic transition matrix is determined by

$$\hat{Q}_{E,E+\Delta E} = \frac{\hat{W}_{E,E+\Delta E}}{\sum_{\Delta E'=-1}^{1} \hat{W}_{E,E+\Delta E'}}.$$

Fig. 7.1 illustrates all macro states and all possible transitions. Note that jumps from the empty structure allways occur with zero waiting time. The method described in Sec. 2.8.1 can be used to obtain an estimate of the DOS $g(E)$ from $\hat{Q}$. This is done by iterating the master equation

$$g(E_i; t+1) = \sum_k \hat{Q}_{(E_k),(E_i)} \cdot g(E_k; t) \tag{7.2}$$

with some initial guess $g(E; 0)$. The iteration is stopped, when the relative change of $g(E)$ between the t*th* and the $(t+1)$*th* iteration is sufficient small for all energies.

As discussed in Sec. 2.8.1, if the microcanonical property is fullfilled, g converges towards the true DOS as the number of simulations, M, tends to infinity. These convergence properties are to be discussed in Sec. 7.2 for the two algorithms under consideration here. Although the general concepts of the Monte Carlo algorithms were introduced in Chapter 2, a few remarks which are relevant for the specific application of the RNA secondary structure are made in the following.

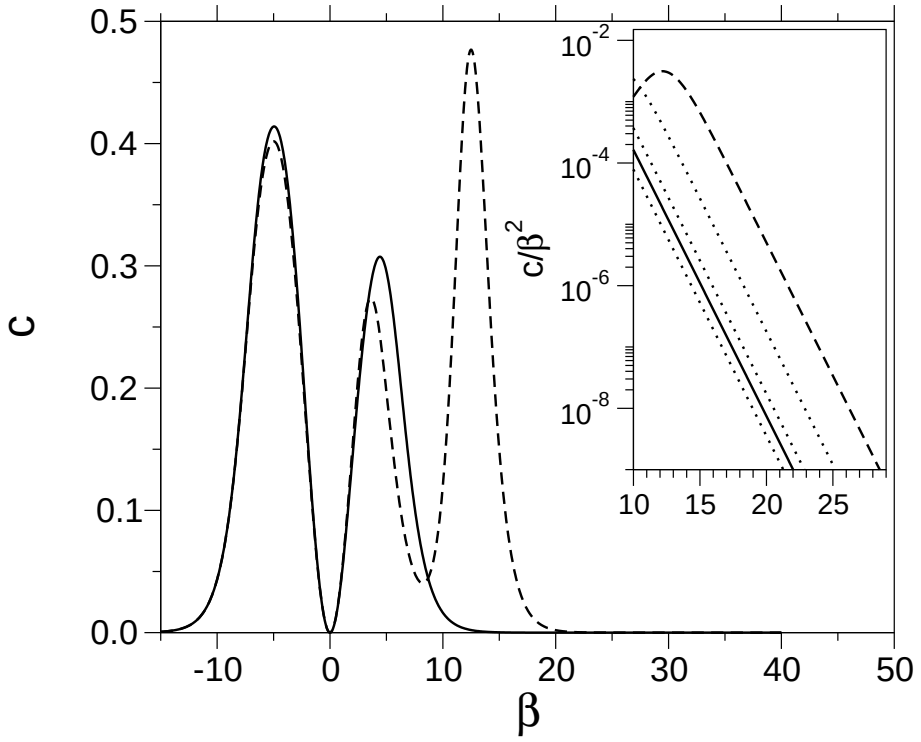


Figure 7.2: Specific heat as a function of the temperature, shown for two realization of length $L = 80$ with a typical (solid line) and a large (dashed) ratio $g(E_1)/g(E_0)$. Inset: low temperature decay rate of the reduced specific heat $c/\beta^2 \sim \mathrm{e}^{-\beta}$ is the same for all realizations. Dotted lines show some other realizations.

7.1.1 The ParQ simulation

The ParQ algorithm [AHM$^+$88, HH05] combines ideas from simulated annealing [KGV83, JJS06] and Transition Matrix Monte Carlo. Instead of estimating the transition matrix from an equilibrium simulation, the temperature is lowered according to a certain protocol. The acceptance rule is the usual Metropolis one

$$\alpha = \min\left(1, \exp[-\beta \Delta E]\right),$$

where $\beta = 1/T$.

The advantage of the method is that no assumption about the DOS is required at the beginning of the simulation. Secondly, in contrast to the Wang Landau method, ParQ is easy to parallelize because many independent runs can be performed simultaneously.

It is required that all regions of interest are visited by the random walk. Therefore, the annealing schedule has to be adjusted. Basically there are two ingredients: the functional form of the (inverse) temperature protocol $\beta(t)$ and the start and end value of the temperature β_1 and β_2. At infinite temperature, the random walk is located at the maximum of the DOS (see Fig. 5.4 in Chapter 5), which corresponds to simple sampling, where all allowed steps are accepted. In order to go beyond the maximum towards the unfolded RNA, i.e. increasing the energy, one has to chose a *negative*

temperature. For the opposite direction, towards the ground state, the temperature has to be positive and finite.

The simplest annealing schedule is a linear increase of the inverse temperature β from $\beta_1 < 0$ to $\beta_2 > 0$. This kind of protocol will be denoted as *inverse schedule* (INV).

However, this kind of cooling schedules might not be optimal. Therefore I also checked two other forms, where the inverse temperature is first increased from β_1 to a certain positive value above the critical temperature [PPRT00], say $\beta = 1$ in a linear fashion. Then the system is cooled down either linearly or exponentially in the temperature T. We will denote these schedules as *linear* (LIN) or *exponential* (EXP) cooling respectively and compare the performance of the three methods later on.

The temperature range $[\beta_1, \beta_2]$ should be chosen, such that the energy fluctuations vanish sufficiently. This can be assessed by considering the specific heat capacity $c = \beta^2(\langle e^2 \rangle - \langle e \rangle^2)$ [JJS06] obtained from exact calculations, where e is the energy per base, i.e. $e = E/L$. For the usual case of other systems, where the DOS is not a priori known, $c(\beta)$ has to be estimated from a few primary simulations or the temperature range has be estimated in other heuristic ways.

The specific heat capacity for two different realizations of length $L = 80$ is shown in Fig. 7.2. For these two examples, I used inverse temperature ranges $[-10, 10]$ and $[-10, 15]$, respectively. Note that the decay of c in the low temperature limit $\beta \to \infty$ (see inset of Fig. 7.2) can be understood very well [WS88] via the ratio $g(E_1)/g(E_0)$ of the number of first excitations with energy $E_1 = E_0 + 1$ and the degeneracy of ground states. At low temperatures the partition function is dominated by the ground state and first excitations only and hence

$$\frac{c}{\beta^2} = \langle e^2 \rangle - \langle e \rangle^2 \sim (E_0 - E_1)^2 \cdot \frac{g(E_1)}{g(E_0)} \cdot \mathrm{e}^{-\beta(E_1 - E_0)}$$

Since $E_1 - E_0 = 1$ for all realization in our simple model the specific heat capacity decays as $C/\beta^2 \sim \exp(-\beta)$ and only the prefactor is dominated by large sample-to-sample fluctuations of $g(E_1)/g(E_0)$ (see Sec. 7.3.1). A large value of this ratio implies a narrowed peak of the specific heat capacity and hence increasingly slow relaxation times. In more complex systems, such as RNA secondary structure with hybrid energy models [BH05], even the exponent may vary because of variable energy difference between ground states and first excitations.

7.1.2 Flat-histogram and optimized ensembles

The generalized ensemble was introduced in Sec. 2.7. The basic idea is that each macro state is sampled with equal probability, instead of sampling configurations according to the Boltzmann weight $w(E) \propto \exp(-\beta E)$. A *perfectly flat histogram* ensemble, where $w(E) \propto 1/g(E)$, requires the knowledge of the DOS $g(E)$.

In Monte Carlo simulations it is usually desired to reduce the autocorrelation times in order to obtain more independent samples within fewer Monte Carlo steps (see Sec. 2.5.2). For this reason, the perfectly flat histogram ensemble might not be the best choice. Especially near phase transitions, where the specific heat diverges, a huge amount of computation time is required. Therefore I also considered the optimized ensemble method [THT04] , where the weights are optimized by minimizing the round-trip time (see Sec. 2.7.2). The optimal weights $w^{\mathrm{opt}}(E)$ are determined iteratively via the recursion relation Eq. (2.21). Note that the pair-energy model of RNA secondary structures was also used to illustrate the convergence of the method in Fig. 2.4.

Since the events for going from a first excited state E_1 to the ground state E_0 occur very rarely, the statistics and the iteration scheme Eq. (2.21) converges slowly, if the complete energy spectrum is considered for optimizing the ensemble. For this reason I employed two energy intervals, the complete one $[E_0, 0]$ and a restricted one $[E_-, E_+] \equiv [E_0 + 1, 0]$. All states of the complete energy range are allowed to be visited by the random walk. For the optimization of the weights the restricted energy interval was used. The link to the remaining weight $w^{i+1}(E_0)$ can be made by requiring the next iteration, $(i+1)$, to visit either the ground state or the first excitations with equal probability (any other finite fraction will work as well), i.e. we set

$$w^{i+1}(E_0) \quad = \quad w^{i+1}(E_1) \cdot \frac{g(E_1)}{g(E_0)}.$$

During each iteration, the round-trip time of the random walk over the full spectrum from E_0 to the null structure was used as a quantity which describes the performance. For a small system $L = 40$, I compared the performance of the optimization over the full spectrum and the restricted spectrum and found no significant difference in round-trip times. In both cases the round-trip time decreases by a factor of about 2 already in the second iteration of updating the weights. For $L = 40$ this iteration scheme was already illustrated in the general introduction to Monte Carlo methods in Fig. 2.4.

7.2 Convergence properties of the Monte Carlo algorithms

In order to assess the performance of different MC algorithms, I conducted simulations using the different approaches described above. I compared the performance using a fixed realization of length $L = 80$ and small ground-state degeneracy, i.e. a large ratio $g(E_1)/g(E_0)$. This ratio is somehow a measure for the amount of meta-stable states. It is a purely local property and does not depend on large structures of the energy landscape. Those instances with a large value of this ratio are the expected to be "hardest" instances by comparison with spin glasses [ATHT04, DTW$^+$04], as indeed confirmed by our results, see Sec. 7.3.

For all simulations techniques, 5×10^{10} MC steps (Metropolis updates) were used totally. Measured in real time, one run of 5×10^9 steps costs approximately one hour on a modern CPU. Simulated annealing with exponential cooling was only slightly slower than the optimized ensemble.

I performed various independent simulations (25 for ParQ and the optimized ensemble and 16 for the flat histogram approach) with different seed values but fixed realization. The convergence properties which are discussed in the following are averaged over these simulations. The DOS is estimated from empirical transition matrices obtained up to certain numbers of Monte Carlo steps.

The ParQ result was obtained by a linear and inverse temperature schedule with a temperature range from $\beta_1 = -10$ to $\beta_2 = 15$ (data for the inverse schedule is not shown in Fig. 7.3) and the overall 5×10^{10} MC steps were separated in 10 independent runs of length 5×10^9. For inverse schedule also 100 independent runs á 0.5×10^9 steps were tested.

The so approximated DOS obtained from ParQ was used as an input for the flat-histogram method, as well as for the first iteration of the optimized flat-histogram method. This might be a realistic procedure, because the DOS is not known in general. For the standard flat-histogram approach, no further adjustment had to be made,

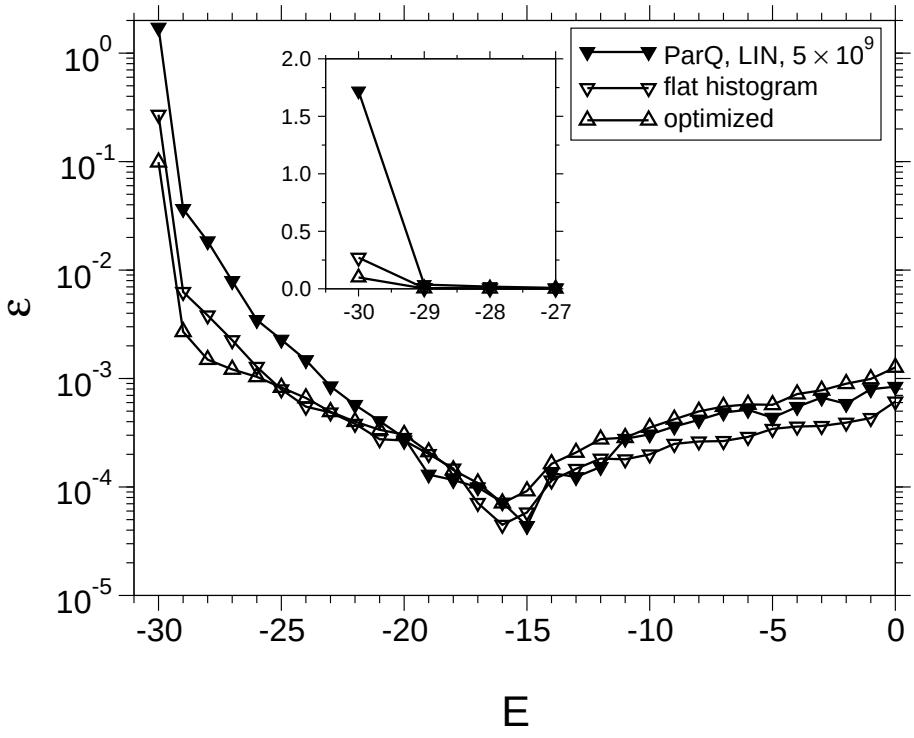


Figure 7.3: Relative error of the DOS of a low degeneracy of ground states using flat histogram ensemble, optimized ensemble and ParQ with fixed cooling rate (5×10^{10} steps per run). The inset shows the same data with a linear ordinate. The ratio $g(E_1)/g(E_0)$ for the realization was $3120649/16$, which is larger than typical values. Lines are guides to the eyes only.

hence the histograms could be sampled using all available MC steps. For the optimized ensemble, to optimize the function $f(E)$, describing the history of the walk with respect to the labels $+$ and $-$, I applied 10^9 MC steps for the first iteration and then doubled this number always for each following iteration. Similar to the $L = 40$ system (Fig. 2.4), the estimate of $f(E)$ converged after only 5 iterations, i.e. totally $(1 + 2 + 4 + 8 + 16) \times 10^9 = 3.1 \times 10^{10}$ steps.

Hence, the optimal weights were found quickly. Via this optimization, the round-trip time decreased by a factor of about 4.

For the remaining 1.9×10^{10} steps where the weights were kept fixed the transition matrices from all iterations had been used to obtain the convergence of the DOS.

To compare the power of the different algorithms, I considered the relative error of the MC approximation with respect to the exact solution

$$\epsilon(E) = \left| g(E) - g^{\text{exact}}(E) \right| / g^{\text{exact}}(E),$$

where g is the sample estimate obtained by the iteration of the master equation, Eq. (7.2). The averaged $\epsilon(E)$ is shown in Fig. 7.3. A second quantity, which gives a relevant measure of performance is the sample error of the ratio of the number of first

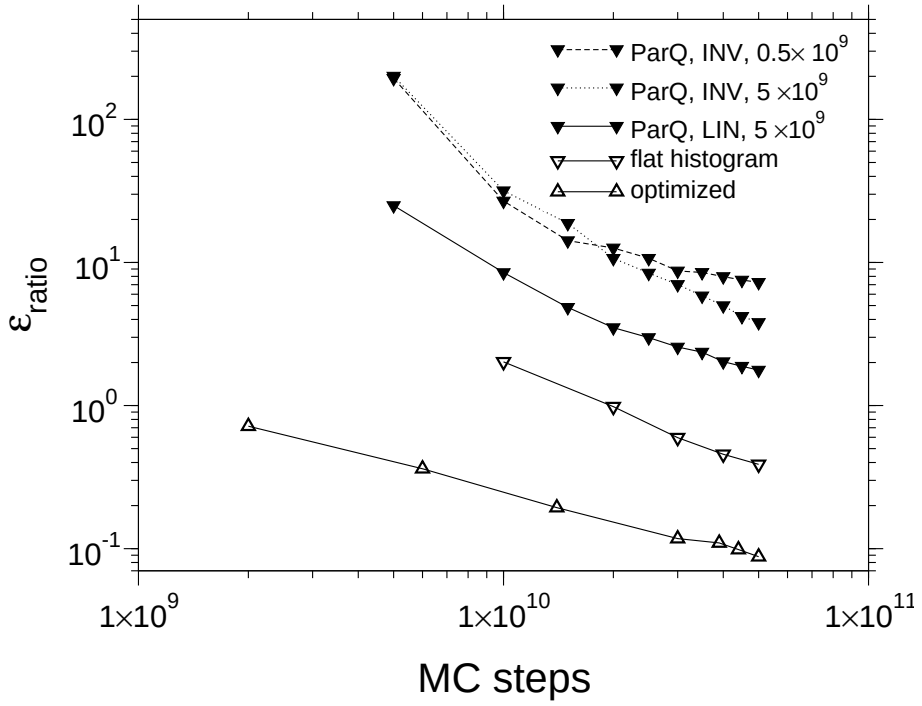


Figure 7.4: Rate of convergence of the relative error of the ratio $g(E_1)/g(E_0)$ of one instance for different simulation methods: ParQ for inverse and linear cooling schedule using 500×10^6 and 5000×10^6 steps per run. For the ParQ run with 500×10^6 steps, only accumulated data up to certain numbers of MC steps are shown.
For 5000×10^6 steps no significant difference between inverse and linear cooling is visible. Flat histogram and the optimized ensemble sampling perform much better than ParQ. The ratio $g(E_1)/g(E_0)$ for the realization was $3120649/16$, which is larger than for typical instances.

excitations and ground states

$$\epsilon_{\text{ratio}} = \frac{|g(E_1)/g(E_0) - g^{\text{exact}}(E_1)/g^{\text{exact}}(E_0)|}{g^{\text{exact}}(E_1)/g^{\text{exact}}(E_0)}. \tag{7.3}$$

This quantity as a function of MC steps is shown in Fig. 7.4.

From Fig. 7.3 and Fig. 7.4 one can learn that in the high-energy region were only a few sites are connected by bonds, the flat histogram method clearly outperforms the other methods, whereas in the relevant low-energy region the optimized random walk seems to be best. The most significant difference between the methods is located at the ground state of the system, where the ParQ method is not very accurate. Also the rate and form of the annealing schedule affects the performance: The linear schedule seems to outperform the inverse schedule and, as expected, few long runs beat many short ones.

Note that Fig. 7.3 and Fig. 7.4 are worst case scenarios, because I picked out a sample, where the ratio $g(E_1)/g(E_0)$ is very large, i.e. there are many meta-stable states that might be separated by large barriers from ground states. I also performed the same

kind of simulations for a typical realization of the same length, where $g(E_1)/g(E_0)$ is relatively small. The errors of the ratio decrease by a factor of 9.5, 35 and 39 for the ParQ, flat histogram and optimized ensemble method respectively, but the generalized ensemble methods still outperform the ParQ method.

In order to check, if the qualitative ranking of the methods, i.e. $\epsilon_{\text{ratio}}^{\text{optimized}} < \epsilon_{\text{ratio}}^{\text{flat}} < \epsilon_{\text{ratio}}^{\text{ParQ,LIN}}$, is a general feature of the system I generated an ensemble of 2000 realizations of length $L = 40$ and performed the same kind of simulations as before with 5×10^7 steps for all simulations. In the majority of the cases (59%) I find the same kind of ranking and second most frequently (33%) a ranking of $\epsilon_{\text{ratio}}^{\text{flat}} < \epsilon_{\text{ratio}}^{\text{optimized}} < \epsilon_{\text{ratio}}^{\text{ParQ,LIN}}$. Only in 2% percent of the cases ParQ outperforms one of the generalized ensemble methods. Sample averages of $\epsilon_{\text{ratio}}^{\text{optimized}}$, $\epsilon_{\text{ratio}}^{\text{flat}}$ and $\epsilon_{\text{ratio}}^{\text{ParQ,LIN}}$ were 0.030, 0.055 and 0.551 respectively. Probably these differences increase for larger systems.

I also checked that linear cooling is better than the other two alternatives in 53% of the cases (exponential and inverse cooling only 15% and 31% respectively).

7.3 Correlation between algorithmic and structural complexity

As already mentioned, the performance strongly depends on the ratio $g(E_1)/g(E_0)$, which was also obtained for the $\pm J$ spin glasses [ATHT04, DTW$^+$04]. In this section we want to study the distribution of this ratio and its relationship to the performance of MC algorithms in the case of RNA secondary structures. I also check if there is a correlation between the degree of ultrametricity at finite temperature (see Sec. 7.3.2) and performance.

7.3.1 Ratio of number of first excitations and ground states

For the usual 2d $L \times L$ Ising ferro magnet without disorder it is obvious that the ratio $g(E_1)/g(E_0)$ scales as L^2, because there are exactly $L \times L$ possibilities to excite the ground state by one single spin flip. In our model, RNA secondary structures, the scaling behavior can not be obtained with such simple arguments.

Hence, I generated ensembles of up to $40,000$ realizations for sequence lengths between $L = 20$ and $L = 1021$ and obtained the distribution of $g(E_1)/g(E_0)$. Even though the transfer matrix algorithm is polynomial, the computations of systems larger than $L = 320$ become very time consuming. Therefore, I only computed the number of ground states and first excitations instead of the complete energy spectrum for larger systems. This can be achieved by truncations of the polynomials in the transfer matrix after the term of the second largest power.

Empirically one can find a *generalized extreme-value distribution* (see Fig. 7.5), whose cumulative distribution function is given by

$$\text{Prob}\left(\frac{g(E_1)}{g(E_0)} \geq x\right) = \exp\left[-\left(1 + \xi\,\frac{x-\mu}{\sigma}\right)^{-1/\xi}\right], \tag{7.4}$$

similar as in [DTW$^+$04].

The parameters of the distribution μ (*location*), ξ (*shape*) and σ (*scale*), were obtained through a maximum likelihood fit using the `FORTRAN` program by Hosking [Hos85, Mac89]. The resulting probability density functions (pdf) and the scaling behavior of the fit parameters are shown in Fig. 7.5.

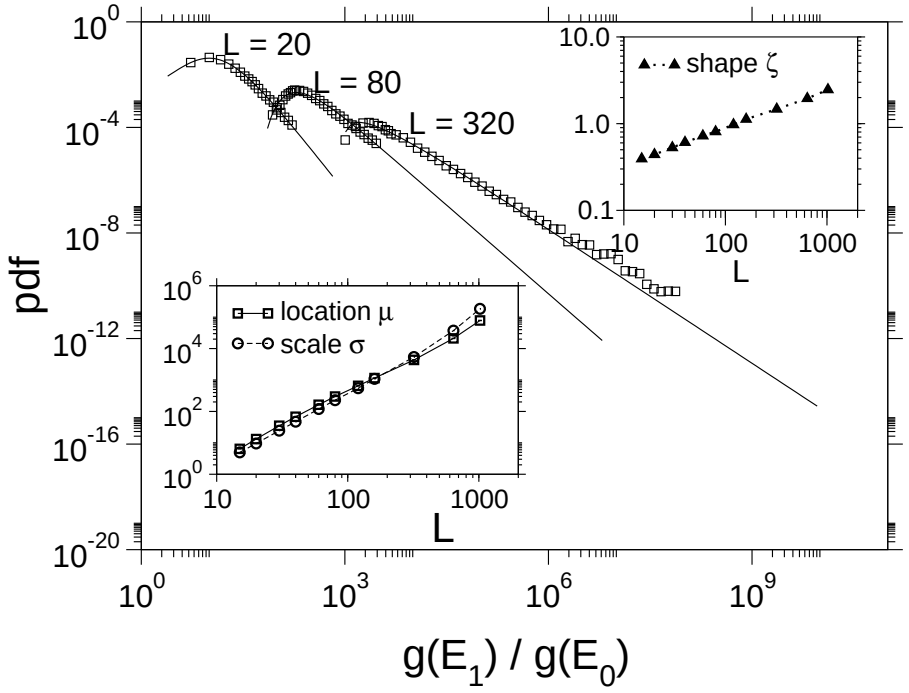


Figure 7.5: Probability density of the ratio $g(E_1)/g(E_0)$ for different sequence lengths. Squares indicate binned data. The largest errorbar is as large as the symbols. Insets: scaling of the location, scale and shape parameter on a double logarithmic scale.

The qualities of the maximum likelihood fits were not good enough to be convinced that the data indeed follows Eq. (7.4), especially for large sequences. This is also supported by small p-values of Kolmogorov-Smirnov tests [Har09]. But the data at least allows one to distinguish between an exponential and algebraic growth of the location and scale parameter: Similar as for the $\pm J$ model [SK94] we find an algebraic behavior of location and shape parameter $\mu(L) = A \cdot L^{z_\mu}$ and $\xi(L) = B \cdot L^{z_\xi}$ with exponents of $z_\mu = 2.1(1)$ and $z_\xi = 2.4(9)$. Although the quality of the fit is not very high (as can be seen already in the lower left inset of Fig. 7.5 where the empirical data do not follow a straight line in the log-log plot), an exponential scaling can be safely excluded by the data.

7.3.2 Ultrametricity of the phase space

The study of ultrametric spaces dates back many decades and has entered the physical literature in the context of spin-glass theory (see [RTV86] and references therein). An ultrametric space M is defined by following axioms:

(i) $0 \leq d(\mathcal{A}, \mathcal{B})$ and $d(\mathcal{A}, \mathcal{B}) = 0 \iff \mathcal{A} = \mathcal{B}$

(ii) $d(\mathcal{A}, \mathcal{B}) = d(\mathcal{B}, \mathcal{A})$

(iii) $d(\mathcal{A}, \mathcal{C}) \leq \max\{d(\mathcal{A}, \mathcal{B}), d(\mathcal{B}, \mathcal{C})\}$,

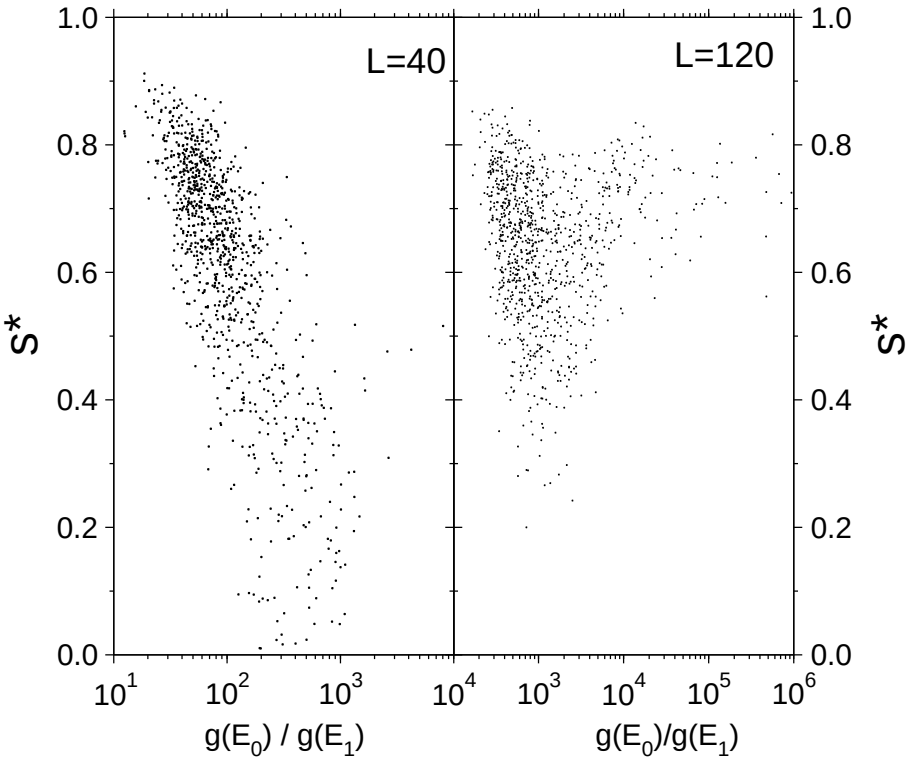


Figure 7.6: Scatter plot between $g(E_1)/g(E_0)$ and s^* at $T = 0.125$.

for all $\mathcal{A}, \mathcal{B}, \mathcal{C} \in M$ and the metric $d(\cdot, \cdot)$. Note that every ultrametric space is a metric space.

Higgs found evidence that RNA secondary structures exhibit an ultrametric structure [Hig96] at low temperatures. The existence of a phase transition was then confirmed numerically by Pagnani et al. [PPRT00] by considering the width of the overlap distribution (see Sec. 5.4) and then examined by other authors using droplet theory [BH02b], the ϵ-coupling method [FKM02] and renormalized field theory [LW06].

Ultrametricity can be detected by considering the "distance" between two structures drawn from a canonical ensemble at a given temperature. Using the transfer matrix $Z_{i,j}$ it is possible to draw states directly without performing Markov chain MC [Hig96] (see Appendix A.2).

The overlap of two structures $\mathcal{C}_1$ and $\mathcal{C}_2$) was defined by Eq. (5.3),

$$q(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{L}\left[2 \sum_{(i,j)\in\mathcal{C}_1} \sum_{(k,l)\in\mathcal{C}_2} \delta_{i,k}\delta_{j,l} + \sum_{i\notin\mathcal{C}_1} \sum_{k\notin\mathcal{C}_2} \delta_{i,k}\right],$$

With this definition we may define a normalized distance between $\mathcal{C}_1$ and $\mathcal{C}_2$, by

$$d(\mathcal{C}_1, \mathcal{C}_2) = 1 - q(\mathcal{C}_1, \mathcal{C}_2).$$

In perfectly ultrametric spaces each triangle is isosceles, i.e. the two largest sides of a triangle are of equal length. This property provides a numerical criterion for the

detection of ultrametricity [Hig96]: The degree of ultrametricity can be estimated by the difference of the two largest distances of a set of candidate triangles. This quantity, which is denoted as s, vanishes in perfectly ultrametric spaces and become small in approximately ultrametric spaces.

There might be different reasons, why s may vanish in the thermodynamic limit and therefore one has to filter real ultrametricity against trivial one. E.g. in the high temperature phase, s also scales as $N^{-1/2}$ and the maximum size of s is limited by the triangle equation. To distinguish from this trivial ultrametricity, Higgs sampled [Hig96] sets of "uncorrelated" triangles by computation of three independent distances $d(\mathcal{C}_1, \mathcal{C}_2), d(\mathcal{C}_3, \mathcal{C}_4), d(\mathcal{C}_5, \mathcal{C}_6)$. Each $\mathcal{C}_i$ was drawn from the canonical distribution

$$P(\mathcal{C}) = \frac{1}{Z} \exp[-\beta E(\mathcal{C})].$$

If these distances fulfill the triangle inequalities, i.e.

$$d(\mathcal{C}_1, \mathcal{C}_2) \leq d(\mathcal{C}_3, \mathcal{C}_4) + d(\mathcal{C}_5, \mathcal{C}_6)$$

and for all other combinations of the distances, the difference between the two largest distances is computed. Finally the average of the differences taken over all valid uncorrelated triangles s_{uncor} is computed. Non-trivial ultrametricity should emerge faster than the trivial ultrametricity obtained from uncorrelated distances. Hence $s^* = s/s_{\text{uncor}}$ should vanish in the presence of an ultrametric structure in the thermodynamic limit.

In principle one should distinguish the finite temperature and zero temperature behavior in complex phase, as already pointed out in [Har01]. Using direct sampling of ground states a "non-trivial" overlap distribution at zero temperature could be ruled out by numerical extrapolation. This implies that grounds states alone are *not ultrametric*. For this reason I considered only finite temperature states, where the overlap distribution is non-trivial [PPRT00] and evidence for an ultrametric phase space still remains.

In small systems the correlations between the ratio $g(E_1)/g(E_0)$ and s^* (see Fig. 7.6) are stronger. We assume that this is a finite-size effect, because this effect is weaker for larger systems.

A widely used technique to visualize hierarchical spaces is the use of dendrograms and distance matrices, which had already been shown in Sec. 3.6.2 in the context of finite-temperature sequence alignment. Here, I used the distance measure $d(\cdot, \cdot)$ introduced above and the clustering method by Ward [JD88] (see Appendix A.3) to illustrate the structure of the static state space.

In Fig. 7.7 four different distance matrices for different realizations and temperatures are illustrated. As one can see, a clear cluster structure emerges only at low temperatures. Note that one can apply a clustering algorithm to any set of data, hence also to non-ultrametric ones. There are quantitative methods, which test how much the tree imposed by the clustering algorithm correlates with the distances in the data. Here, we have just used the visual impressions obtained by looking at the matrices. Furthermore, in Sec. 7.3.4 we will use the so detected clusters to check whether all ground states are visited with equal probability.

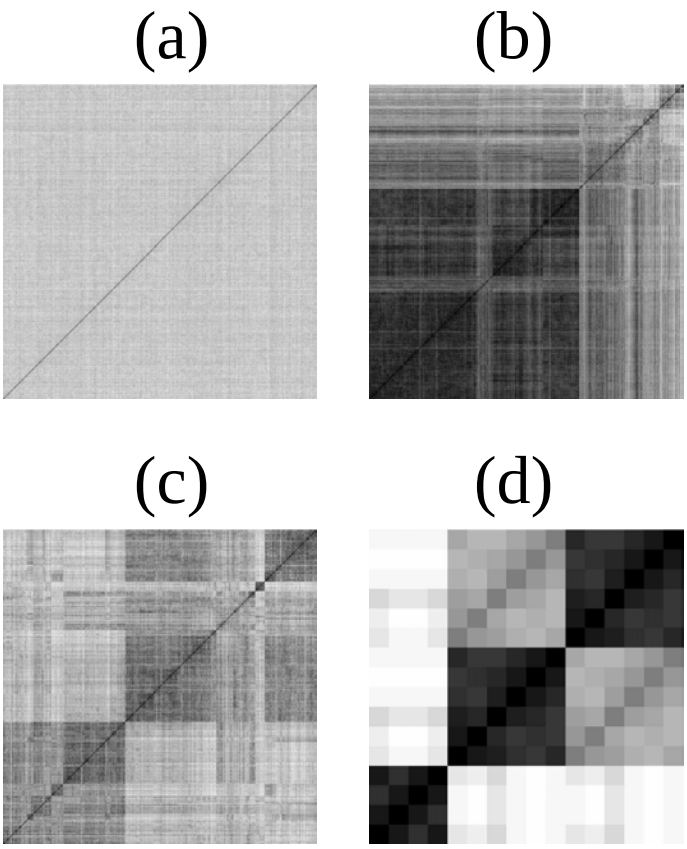


Figure 7.7: Hierarchical structure of the states illustrated by distance matrices. Darker gray scales correspond to large overlaps.
(a) $L = 120$ at $T = 2$. No hierarchical structure could be detected ($s^* \approx 1$)
(b) $L = 120$ at $T = 0.125$ for a realization exhibiting a weak ultrametricity ($s^* \approx 0.74$)
(c) $L = 120$ at $T = 0.125$ for a realization exhibiting stronger ultrametricity ($s^* \approx 0.45$)
(d) $L = 40$ at $T = 0.0$ for a realization having low ground-state degeneracy ($g(E_1)/g(E_0) = 14638/16$) Deviation from ultrametricity was $s^* \approx 0.5$. Realization (d) was also used in Sec. 7.3.4. The corresponding dendrogram is illustrated in Fig. 7.10.

7.3.3 Distribution of tunneling times of the flat histogram random walk

Next, we consider the tunneling time for the flat histogram random walk for sequence lengths up to 120. Recall that the tunneling time was defined as the number of Monte Carlo steps that the generalized ensemble random walk needs to find the ground state starting from the empty structure. The round trip time, i.e. the time to find the ground state and go back to the empty structure is effectively indistinguishable in the system here, because the tour back to empty structure is one order of magnitude faster.

Note that larger systems become infeasible if one wants to span a large energy

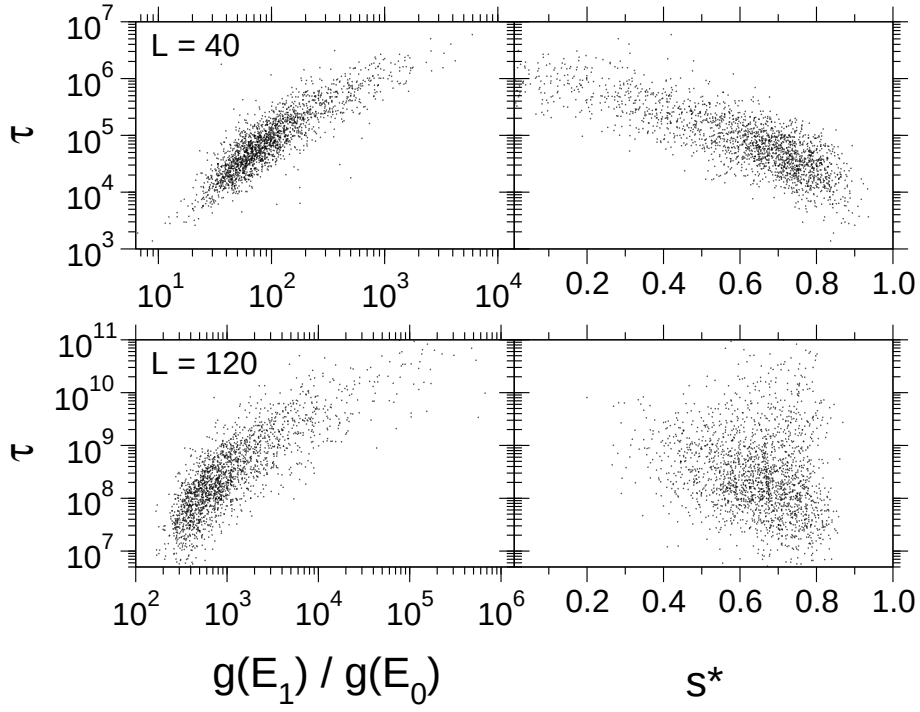


Figure 7.8:
Left column: Scatter plot between the ratio $g(E_1)/g(E_0)$ and tunneling time of the flat histogram sampler for $L = 40$ (top) and $L = 120$ (bottom).
Right column: Scatter plot between deviation from ultrametricity and tunneling time of the flat histogram sampler for $L = 40$ (top) and $L = 120$ (bottom).

interval, which we need to when studying tunneling time distributions. Already for $L = 120$ I found tunneling times fluctuating, in real time, between seconds and days on a modern CPU.

There is a strong correlation between $g(E_1)/g(E_0)$ and the tunneling time, see Fig. 7.8. I found that this is in particular true when this ratio is much larger than all other ratios between neighboring energy densities. The performance of the algorithm is then dominated by the rare event of finding a ground state when starting from a first excitation. Two scatter plots of the ultrametricity index s^* versus tunneling time are shown in Fig. 7.8. Hence, whether there is a true correlation between tunneling time τ and degree of ultrametricity s^* is not clear, because for larger system the correlation appears to be rather weak.

To investigate the issue of correlations between static measures and computational hardness more quantitatively, I calculated the empirical Pearson correlation coefficients for all pairs of quantities $\log \tau$, $\Delta S = \log\left(g(E_1)/g(E_0)\right)$ and s^*. The results are summarized in Tab. 7.1.

The correlation between s^* might be trivial, because it might be induced by correlation of the ratio $g(E_1)/g(E_0)$ and tunneling time. This means, although ultrametricity is usually considered as a landmark of complex and glassy systems, at least for the behavior of RNA secondary structures it is not related to the dynamic glassy behavior

$L = 40$	$\log\tau$	ΔS	$s^*_{(T=0.125)}$	$s^*_{(T=0.033)}$
$\log\tau$	1	0.89	-0.40	-0.33
ΔS		1	-0.37	-0.30
$s^*_{(T=0.125)}$			1	0.87
$s^*_{(T=0.033)}$				1
$L = 120$	$\log\tau$	ΔS	$s^*_{(T=0.125)}$	$s^*_{(T=0.033)}$
$\log\tau$	1	0.82	-0.18	-0.16
ΔS		1	0.02	-0.13
$s^*_{(T=0.125)}$			1	0.28
$s^*_{(T=0.033)}$				1

Table 7.1: Empirical Pearson correlation coefficients for all pairs of quantities $\log\tau$, $\Delta S = \log\left(g(E_1)/g(E_0)\right)$ and s^* for $L = 40$ and $L = 120$.

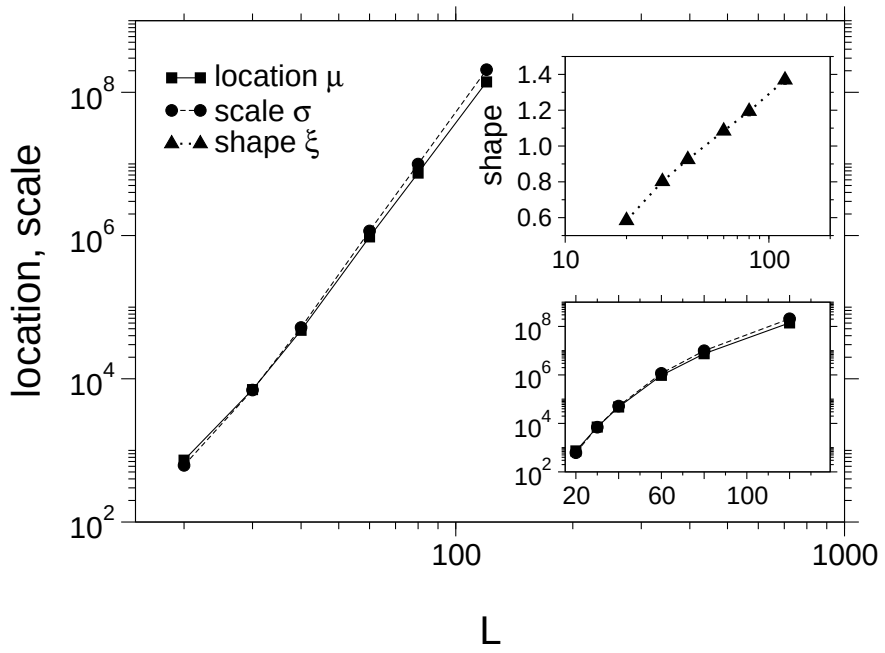


Figure 7.9: Scaling of the location (square symbols) and scale parameters (circles) on the tunneling time. Insets: scaling of the shape parameter using a logarithmic abscissa (top) and scaling of the location and scale parameters using a logarithmic ordinate (bottom).

seen in MC simulations. To my believe the effect of ultrametricity is superimposed by the presence of a large number of metastable states.

I fitted also the distributions of the tunneling time to a generalized extreme-value distribution Eq. (7.4) and analyzed the scaling of the parameters. Location and scale parameter have almost the same algebraic dependence on the sequence length, see Fig.

7.9. As can be seen on the semi-logarithmic plot in the bottom inset in Fig. 7.9, an exponential scaling can safely be excluded (at least on the length scale up to 120). The exponent describing the power-law is roughly $z \approx 7$. On the other hand, the shape parameter seems to scale logarithmically in sequence length, see upper inset in Fig. 7.9. With the same arguments as in Sec. 7.3.1, we cannot exclude that the distribution deviates from a generalized extreme-value distribution.

These results differ from the $\pm J$ Ising model, where an exponential tunneling time was observed in the literature. On the other hand, they are similar to the fully frustrated Ising model, investigated by Dayal et al. [DTW$^+$04]. They argued, that sub-exponential growth of tunneling times and of the ratio $g(E_1)/g(E_0)$ stem from a smaller growth of the number of meta-stable states. These results suggest that the model of RNA is dynamically "simpler" than $\pm J$ spin glasses and has a similar complexity as the fully frustrated model.

On the other hand, sample-to-sample fluctuations are much larger than in the $\pm J$ model, as can be seen by comparing the shape parameter in the range of investigated system sizes. For the largest systems in [DTW$^+$04], the scaling parameter was about 0.9 (see also Appendix B). Hence, although typically RNA instances are not so hard for a MC algorithm, compared to $\pm J$ spin glasses, there is larger fraction of rare hard instances for RNA secondary structures.

7.3.4 Are all ground states visited with equal probability?

From Fig. 7.4 one can also see that the error for the optimized ensemble are one order of magnitude smaller than that of the ParQ method. Since in both cases the data were obtained from the transition matrix, the significant difference must be caused in the underlying MC scheme, probably the non-equilibrium character of ParQ. In order to gain insight to this issue I checked whether the microcanonical property is fulfilled.

In detail, I considered histograms of visited ground states for simulated annealing (ParQ) and the optimized ensemble sampler and checked if the histograms were sufficiently flat. A simple and powerful check for the flatness of a histogram is the Bhattacharyya distance measure (BDM) [Bha43] for two given probability mass functions p and q, which was introduced in Eq. (6.1) in Chapter 6

$$B(p||q) = \sum_i \sqrt{p(i)} \cdot \sqrt{q(i)}.$$

In Chapter 6, this measure was already used for model testing. Here, the *null hypothesis* corresponds to the assertion that all ground states are visited with equal probability. Let $K = g(E_0)$ be the ground state degeneracy and $\hat{h}(i)$ denote the number of events that the random walk visits ground state i. Hence the objective is given by

$$\hat{B} = \sum_{i=1}^{K} \sqrt{\hat{h}(i)/N} \cdot \sqrt{\frac{1}{K}},$$

where $N = \sum_{i=1}^{K} \hat{h}(i)$ is the total number of events, where the sampler hits one of the ground states. We shall make use of the p-value which is associated with the observation $\hat{B}$. This is the probability that an empirical BDM of $\hat{B}$ or larger occurred by pure chance (see Appendix A.4). If the p-value is below 0.05 the evidence that the null-hypthesis is true is very small.

Note that the BDM requires the empirical events to be independent. Hence, I generated histograms of independently visited ground states for a small system ($L = 40$)

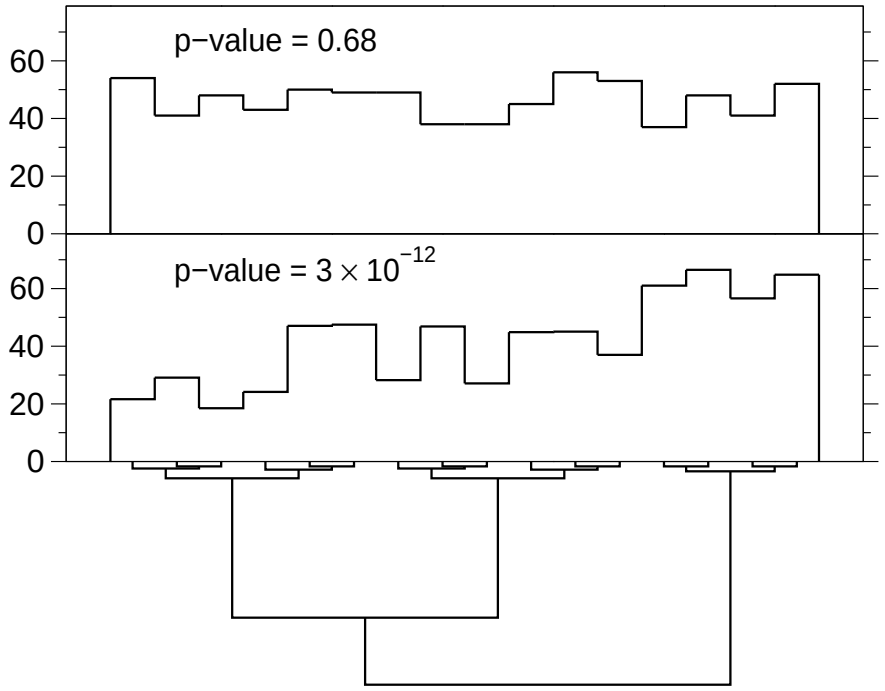


Figure 7.10: Histograms of visited ground states. upper: optimized ensemble random walk, lower simulated annealing. The Bhattacharyya measures and the corresponding p-values indicate, that simulated annealing does not visit all ground states with equal weight.

and low ground-state degeneracy (I selected a realization exhibiting $K = 16$) in the following way: For the optimized ensemble I considered the round-trip time τ and checked each τ'th MC step if the random walk sits currently at the ground state and, if so, the histogram $\hat{h}(i)$ is updated. For simulated annealing, which provides a basis for ParQ, this procedure is not possible in this way, because there is no natural mixing time, which could serve as a thinning interval. Therefore I generated histograms of *all* visited ground states and renormalized the empirical histograms by considering an effective sample size such that each of the $N_{\text{annealing}}$ annealing runs has a "weight" of 1, i.e.

$$\hat{h}^{\text{eff}}(i) = \frac{N_{\text{annealing}}}{N}\hat{h}(i)$$

where $N = \sum_i \hat{h}(i)$ is the total number of events.

Note that in the case that the random variable mixes faster than the number of MC steps for one annealing run the BDM for the effective histogram might be overestimated (and hence the p-values as well). This would yield false positives. However the opposite case might not occur, because all annealing runs are independent from each others. Therefore the so defined effective histograms can only be used to reject the hypthesis, which is exactly what we do here.

The results for both simulation methods are shown in Fig. 7.10. The upper plot shows one of ten histograms of independent runs for the optimized ensemble random walk, which has a large p-value of 0.68. The other nine runs yield p-values between

0.007 and 0.67 ($0.4(1)$ on average) and hence we can accept the null hypthesis. For simulated annealing we find that not all ground states are visited with equal probability (lower plot). The p-values for ten independent runs varied between (7×10^{-13} and 5×10^{-2}). Therefore we have to reject the null hypthesis, hence these simulated annealing runs visit ground states with a bias. For an even faster cooling schedule we find p-values between 2×10^{-24} and 2×10^{-3}. The reason for this bias might be that the random walk gets stuck in preferable local minima. The ground-state structure in form of a dendrogram illustrated in Fig. 7.10 below the histograms supports this argument. The connection between two ground states indicates that these two are merged into one cluster, and the vertical distances are proportional to the Ward-distance, specified in Eq. (A.1) (see Appendix A.3). One can see that *within* the largest clusters the histogram becomes flatter and that the main source of the non-flatness are differences in sampling *between* the largest clusters.

7.4 Rate of convergence in extended state spaces

The relative error of the Monte Carlo estimate of the DOS, shown in Fig. 7.3, suggests that the ground state is hardest to sample. Additionally, as shown in the previous section, simulated annealing fails to find all ground states with equal probability. This leads to the question whether the dynamics of both algorithms can be improved by increasing the number of possible paths from higher excitations to the ground state. It is also desirable to enhance the dynamics such that the random walks are allowed to move from one ground state to another, especially for the ParQ simulations at the finial low-temperature stage, where it is less likely to overcome a barrier through higher excitations.

The main reason of the slow dynamics close to the ground state is due to entropic constraints. In the method that is described in this section, this constraints are partially released in a controlled manner. The idea is to sample from an extended state space $\chi^*_{\mathbf{a}}$, where a certain amount of pseudo knots are allowed. The joint density of states defined on the energy and the number of pseudo knots is then used to obtain the DOS of the original model by a projection. Details on this method are explained now.

In Def. 5.1.1 we have defined a secondary structure $\mathcal{C}$ as a set of bonds, where all pairs of bonds $(i,j),(k,l) \in \mathcal{C}$ with $(i < k)$ are either nested ($i < k < l < j$) or separated ($i < j < k < l$). Here we also allow the case of pseudo knots

$$i < k < j < l \tag{7.5}$$

A new observable, denoted as *number of violations* V, measures the number of violated constraints in the form of Eq. (7.5). The case $V = 0$ corresponds to the original model. Note that non-complementary base pairs and pairs between bases with a shorter distance than $h_{\min}$ in the primary sequence are still excluded.

It is straightforward to generalize the Metropolis algorithm for RNA secondary structures such that also those structures with $V > 0$ are taken into account. Besides the energy, also the numbers of violations V and their potential changes ΔV are to be sampled in the same way as above. This yields the chains of observables in the extended state space $\chi^*_{\mathbf{a}}$,

$$(\hat{E}_1^{(j)}, \hat{V}_1^{(j)}, \Delta\hat{E}_1^{(j)}, \Delta\hat{V}_1^{(j)}, \hat{\tau}_1^{(j)}), \ldots, (\hat{E}_n^{(j)}, \hat{V}_n^{(j)}, \Delta\hat{E}_n^{(j)}, \Delta\hat{V}_1^{(j)}, \hat{\tau}_n^{(j)}).$$

Similar as in Sec. 7.1, j denotes the index of the simulation, i.e. $j = 1, \ldots M$, where M is the total number of independent simulations. A macro state of the

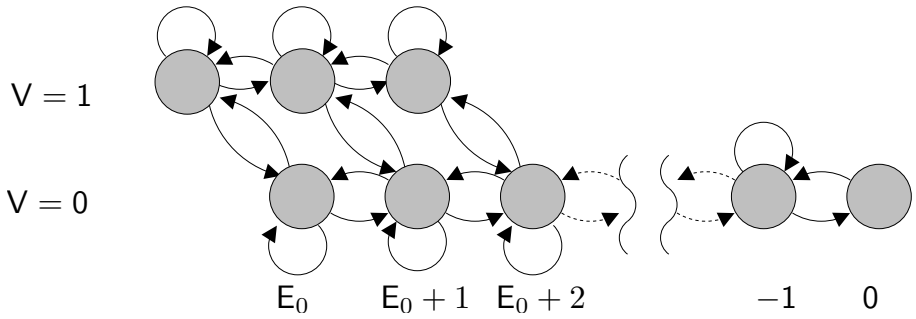


Figure 7.11: Allowed macro states and possible transitions in the extended state space.

extended state space is characterized by the pair (E, V). This also requires a generalization of the matrix $\hat{W}$, that counts the proposed transitions. Instead of counting all transitions between energy states the transitions between all possible transitions between the joint states (E, V) have to be considered. This yields the generalized matrix $\hat{W}_{(E,V),(E+\Delta E,V+\Delta V)}$ from which a stochastic transition matrix $\hat{Q}_{(E,V),(E+\Delta E,V+\Delta V)}$ can be estimated by normalization

$$\hat{Q}_{(E,V),(E+\Delta E,V+\Delta V)} := \frac{\hat{W}_{(E,V),(E+\Delta E,V+\Delta V)}}{\sum_{\Delta E'=-1}^{1} \sum_{\Delta V'} \hat{W}_{(E,V),(E+\Delta E',V+\Delta V')}}.$$

The joint DOS $g(E, V)$ is obtained by iterating the master equation for the joint state transition matrix $\hat{Q}$,

$$g(E_i, V_j; t+1) = \sum_k \sum_l \hat{Q}_{(E_k,V_k),(E_i,E_j)} \cdot g(E_k; V_l; t).$$

An estimate of the DOS of the original model is then obtained by the projection

$$g(E) = \frac{1}{\sum_E g(E, 0)} g(E, 0).$$

In order to estimate the specific heat, we will also need to compute the marginal DOS

$$g^{\text{marg}}(E) = \frac{1}{\sum_{E,V} g(E, V)} \sum_V g(E, V).$$

Because the state space grows drastically, when removing *all* constraints that leads to pseudo knots, we shall concentrate on the interesting region close to the ground state alone. Hence not all states on the $E - V$ plane are considered. A setup, which turned out to be efficient is to allow all states with $V = 0$ as usual and additionally allow for the states $(E_0 - 1, 1)$, $(E_0, 1)$ and $(E_0 + 1, 1)$. Hence, most of the pseudo-knots are still forbidden. Only close to the ground state, where the Monte Carlo simulations exhibit extremely slow dynamics, we extend the state space by three additional macrostates. This setup is illustrated in Fig. 7.11. I also tried a larger set of forbidden states beyond $V = 1$ and also more than three macrostates, but the choice of the three mentioned states seems to be the best compromise I found. Because there are still forbidden structures in this setup, the waiting times have to be taken into account for the extended state space as well.

Here, I considered the same sequence of $L = 80$ that was also considered in the analysis of convergence in Sec. 7.2. The corresponding structure space exhibits a large ratio $g(E_1)/g(E_0)$ and the ground state is therefore hard to sample. The analysis of the extended state space was divided into four parts,

(1) a short ParQ simulation to obtain a rough estimate of $g^{\text{marg}}(E)$ and from that an estimate of the specific heat,

(2) several ParQ simulations to obtain a guess of $g(E, V)$,

(3) several generalized ensemble simulations with flat histogram weights and finally

(4) several generalized ensemble simulations with optimized weights.

In the simulated annealing setup, each proposal is accepted according to the Metropolis algorithm with semi-rejection free dynamics, where the number of violations V did not enter the acceptance rate. For this reason the specific heat based on the marginal DOS was estimated after the primarily ParQ run (1). A visual inspection of the specific heat curve suggested to use the same temperature range ($[-10, 15]$) as for the standard state space at least for this particular realization. In step (2), longer independent ParQ simulations with linear schedule and again 5×10^9 Monte Carlo steps were performed. The resulting rate of convergence of $g(E_1)/g(E_0)$ measured by ϵ_{ratio} is shown in Fig. 7.12. The results were obtained by averaging over ten independent blocks of runs, i.e. 100 runs had been carried out in total. Interestingly, the ParQ method in the extended state space is almost as powerful as the optimized ensemble method in the standard state space.

The joint DOS $g(E, V)$ was then used to choose the weights for the generalized ensemble method as $w(E, V) \propto 1/g(E, V)$. This means, for part (3) and (4) the generalized Metropolis criterion

$$\alpha = \min\left(1, \frac{w(E + \Delta E, V + \Delta V)}{w(E, V)}\right)$$

was used. The generalization of the iteration scheme for the optimized ensemble Eq. (2.21) is straightforward. The labels $+$ and $-$, that the random walk is assigned to in each time step in the extended state space are determined by the observation if the random walk has visited the state $(E_-, 0) \equiv (E_0 + 1, 0)$ or $(E_+, 0) \equiv (0, 0)$ most recently. The histograms $H_\pm(E)$ are defined accordingly and the optimized ensemble iteration is generalized to

$$w^{k+1}(E, V) = w^k(E, V) \cdot \sqrt{\frac{1}{H_+(E) + H_-(E)} \cdot \frac{df}{dE} \cdot \frac{1}{\tau(E)}}.$$

The rate of convergence of $g(E_1)/g(E_0)$ is again obtained by averaging over ten independent runs. As one can see also in Fig. 7.12, the generalized ensemble method in the extended state space enhances further the rate of convergence. Most likely the reason for this kind of enhancement is that the random walk may find ground states more quickly (see Fig. 7.11) . The escape rate from the ground state to higher excitations is also enhanced.

We may conclude that Monte Carlo simulations in extended state spaces enhance the performance for all three cases, for the ParQ algorithm and the generalized ensemble methods in both variants that have been considered in this chapter. In Appendix B a related approach applied to the $\pm J$ spin glass is discussed.

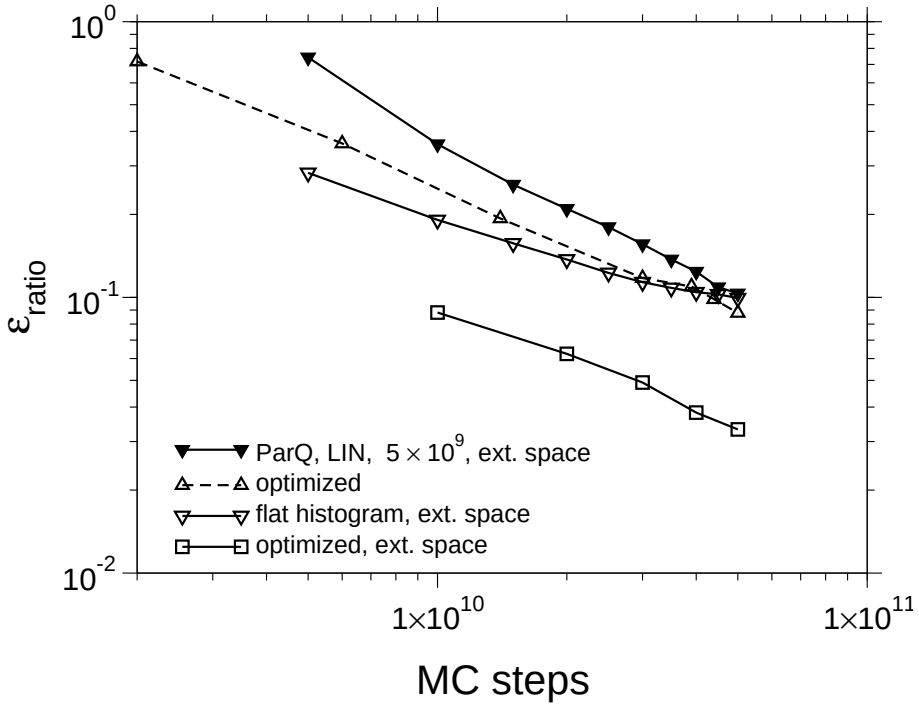


Figure 7.12: Convergence of the relative error of the ratio $g(E_1)/g(E_0)$ in the extended state space. For comparison the data of the best method for the standard state space is also included.

7.5 Conclusion

In this chapter, I discussed the relation between static and (MC) dynamic properties of RNA secondary structures and the relation to the performance of different MC algorithms. This model is an ideal test system for this purpose for three reasons: i) The model exhibits quenched disorder and has a complex low-energy landscape, where an interesting dynamical behavior can be expected. ii) It exhibits a static phase transition at finite temperature. iii) The static behavior of the model can be analyzed exactly using polynomial-time partition-function calculations for each single realization of the disorder.

Analyzing the static behavior, I calculated the DOS for ensembles of sequences of different lengths. In particular, I studied the ratio $g(E_1)/g(E_0)$, which plays the key role in the complexity of MC methods. The distribution of this ratio could be fitted (but not perfectly) to a generalized extreme-value distribution, similar as previously found for the case of $\pm J$ spin glasses. Location, scale and shape of this distribution scale algebraically with system size, in contrast to $\pm J$ spin glasses. I also computed Higgs's measure s^* for the degree of ultrametricity of each realization and used hierarchical clustering approaches to analyze the structure of the landscape.

For the dynamics, I examined two different MC approaches, which served as basis for evaluating the infinite-temperature transition matrix. The nature of the model renders a direct MC implementation very inefficient, hence I also included an N-fold way

sampling scheme. Generalized ensembles with flat-histogram and optimized weights provided examples for equilibrium simulations, whereas ParQ is based on simulated annealing, which is an out-of-equilibrium method. In contrast to $\pm J$ model [HH05], the ParQ method did not yield very accurate results near the ground state and therefore equilibrium methods should be preferred. However, the disadvantage of theses methods is that already suitable initial guesses of the DOS are required. The simulations show that ParQ can provide such guesses and hence a good strategy might be to combine the ParQ method for a first estimate and use the optimized ensemble method for further refinement.

The tunneling time, as a measure of complexity for the flat histogram random walk, is also distributed according to a generalized extreme-value distribution. The scaling of location and scale parameters seems to be algebraic with an exponent of $z \approx 7$, which differs from the spin-glass model studied in the literature. The scaling of the shape parameter indicate much larger sample-to-sample fluctuations than the spin-glass case. Hence, computationally very hard instances occur more often.

Concerning the relation of static properties and dynamical behavior, I found a strong correlation of the MC tunneling times to the value of the ratio $g(E_1)/g(E_0)$.

On the other hand, I could *not* detect a strong direct correlation between MC tunneling times and degree of ultrametricity of the model. Any numerically observed correlations appear only in a trivial way, i.e. due to a correlation between the degeneracy ratio and s^*. Hence, an ultrametric phase space (a kind of global characterization of the energy landscape), as it seems to be present for RNA secondary structures, does not necessarily lead to a complex dynamics. The presence of meta-stable states, which is only a local property of the energy landscape, appears to be much more important.

The analysis of the histogram of visited ground states provides reasons for the failure of the ParQ algorithm: Microstates with equal energy are not visited with equal probability and hence evaluation of the infinite temperature transition matrix does not work correctly. This was not the case for equilibrium methods.

Finally I considered an extended state space, where only one constraint may be violated. In all cases, the rate of convergence of the Monte Carlo algorithms is enhanced. For the ParQ algorithm and the optimized ensemble method the sampling error of the ratio $g(E_1)/g(E_0)$ is one order of magnitude smaller than in the original state space.

Appendix A

Additional algorithms

A.1 Stochastic backtracing for local alignment

This appendix describes the stochastic backtracing procedure by Mückstein, Hofacker and Stadler [MHS02] to sample finite temperature local alignments from the Gibbs-Boltzmann distribution. It depends on the local partition functions $Z^D_{i,j}$, $Z^P_{i,j}$ and $Z^Q_{i,j}$ that are calculated in the dynamic programming algorithm for finite temperature alignment Eq. (3.9).

Firstly a starting point (i, j) of a non-empty alignment or the empty alignment $\mathcal{A} = \{\}$ is selected with probabilities $Z^D_{i,j}/Z$ or $1/Z$ respectively using the inversion method [Dev86]. If the empty alignment has been selected it is returned, otherwise a random walk in backward direction is performed. This walk starts at the randomly chosen starting point (i, j). The probabilities in each step are chosen according to the local partition functions $Z^D_{i,j}, Z^P_{i,j}$ and $Z^Q_{i,j}$ as well as the local scores $\sigma(a_i, b_j)$. Details of are shown in Algorithm A.1.1.

A.2 Pair probabilities of RNA secondary structures and hierarchical backtracing

In the last section a stochastic backtrace procedure for the finite temperature alignment has been discussed. A related method for RNA secondary structures is also available [Hig96]. Here, the algorithm for pair-matching model is presented. The partition functions of structures on the subsequence $a_i \dots a_j$ are available after the course of the dynamic programming algorithm Eq. (5.1).

Due the hierarchical structure, the easiest way is to implement this algorithm recursively. The procedure `hierarchical_backtrace` in Algorithm A.2.1 returns a structure on the sequence $a_i \dots a_j$ drawn according to the Boltzmann weights. First the probability that j is paired with h is computed for all $h = i \dots j - h_{\text{min}}$. Probabilities can be computed from the partition functions $Z_{k,l}$ with $i \le k < l \le j$ [McC90],

$$p_h = \frac{Z_{i,h-1} \cdot \mathrm{e}^{-\epsilon_{h,j}/T} \cdot Z_{h+1,j-1}}{Z_{i,j}}.$$

procedure stochastic_backtrace(Z^D, Z^P, Z^Q, i, j, T)
begin
  $\mathcal{A} \leftarrow \{\}$
  state ← match

  while(state != stop) **do**
    p_diag, p_gap_a, p_gap_b, p_stop ← 0
    case state **of**
      match:
        $\mathcal{A} \leftarrow \mathcal{A} \cup \{(i,j)\}$
        if(i = 1 **or** j = 1) **then** p_stop ← 1
        else
          p_diag ← $(Z^D_{i-1,j-1}/Z^D_{i,j}) \cdot e^{\sigma(a_i,b_j)/T}$
          p_gap_a ← $(Z^P_{i-1,j-1}/Z^D_{i,j}) \cdot e^{\sigma(a_i,b_j)/T}$
          p_gap_b ← $(Z^Q_{i-1,j-1}/Z^D_{i,j}) \cdot e^{\sigma(a_i,b_j)/T}$
          p_stop ← 1 - p_diag - p_gap_a - p_gap_b
        end
        $i \leftarrow i-1$; $j \leftarrow j-1$
      gap_b:
        if i = 2 **and** j = 1 **then** p_diag ← 1
        else
          p_diag ← $(Z^D_{i-1,j}/Z^P_{i,j}) \cdot e^{-\alpha/T}$
          p_gap_b ← $(Z^P_{i-1,j}/Z^P_{i,j}) \cdot e^{-\beta/T}$
          p_gap_a ← 0
        end
        $i \leftarrow i-1$
      gap_a:
        if i = 1 **and** j = 2 **then** p_diag ← 1
        else
          p_diag ← $(Z^D_{i,j-1}/Z^Q_{i,j}) \cdot e^{-\alpha/T}$
          p_gap_b ← $(Z^P_{i,j-1}/Z^Q_{i,j}) \cdot e^{-\alpha/T}$
          p_gap_a ← $(Z^Q_{i,j-1}/Z^Q_{i,j}) \cdot e^{-\beta/T}$
        end
        $j \leftarrow j-1$
    end
    set state to match, gap_a, gap_b, stop
      with probabilities p_diag, p_gap_a, p_gap_q, p_stop
  end
  return $\mathcal{A}$
end

Algorithm A.1.1: The stochastic backtrace procedure for local alignment with affine gap costs [MHS02]. It is assumed that a starting point (i,j) has been chosen with probability $Z^D_{i,j}/Z$.

procedure hierarchical_backtrace(Z, i, j)
begin
  $\mathcal{C} \leftarrow \{\}$
  for h **in** $i \ldots j - h_{\min}$ **do**
    p[h] $\leftarrow Z_{i,h-1} \cdot \mathrm{e}^{-\epsilon_{h,j}/T} \cdot Z_{h+1,j-1}/Z_{i,j}$
    $\mathrm{p}_{\text{unpaired}} \leftarrow 1 - \sum_{h=i}^{j=1} p_h$.
    h $\leftarrow$ choose h or undefined with probabilities p[h],$\mathrm{p}_{\text{unpaired}}$
    if h $\neq$ undefined **then**
      $\mathcal{C} \leftarrow \mathcal{C} \cup \{(h, j)\}$
      $\mathcal{C} \leftarrow \mathcal{C} \cup$ hierarchical_backtrace$(Z, i, h-1)$
      $\mathcal{C} \leftarrow \mathcal{C} \cup$ hierarchical_backtrace$(Z, h+1, j-1)$
    else
      $\mathcal{C} \leftarrow \mathcal{C} \cup$ hierarchical_backtrace$(Z, i, j-1)$
    end
  end

  return $\mathcal{C}$
end

Algorithm A.2.1: Hierarchical backtrace procedure for sampling of RNA secondary structures from the canonical ensemble [Hig96]. It is assumed that the partition functions $Z_{i,j}$ are known. To sample a structure on the complete sequence $a_1 \ldots a_L$ the procedure has to be called as hierarchical_backtrace(Z, 1, L)

The probability that j is not paired with any letter $i \leq h < j$ is given by

$$p_{\text{unpaired}} = 1 - \sum_{h=i}^{j-1} p_h.$$

Next, a new state for j is drawn with the inversion method: It is paired to h with probability p_h or remains unpaired with p_{unpaired}. This kind of selection induces independent subsystems. If j was paired with h the systems $a_i \ldots a_{h-1}$ and $a_{h+1} \ldots a_{j-1}$ are treated in the same way independently of each other. Otherwise a larger subsystem $a_i \ldots a_{j-1}$ is left for the next recursion. This procedure is repeated until $i = j - 1$ in each branch. A related method to sample ground-states microcanonicaly, i.e. with equal weight was proposed by [Har01].

A.3 The clustering method

Fig. 3.13 in Sec. 3.6 and Fig. 7.7 in Sec. 7.3.2 display distance matrices of local alignments and RNA secondary structures in the canonical ensemble. The states have been sorted according to a certain cluster criterion, which is explained in this appendix.

The algorithm used here is Ward's algorithm [JD88], an agglomerative hierarchical matrix updating algorithm, also called minimum variance method as it is designed to minimize the variance of the constructed clusters. The method requires a set of states from a state space χ and a distance measure $d : \chi \times \chi \to \mathbb{R}$. The algorithm works as follows. Initially each state forms a cluster of its own, and the distance matrix Δ_{ab} with the distances of all pairs of clusters (each containing one configuration) is calculated

using the distance d. Then in each step the two clusters p and q with the smallest distance are fused to form a new cluster t. The distance matrix is updated using

$$\Delta_{rt} = \frac{(n_r + n_p)\,\Delta_{rp} + (n_r + n_q)\,\Delta_{rq} - n_r\,\Delta_{pq}}{n_r + n_t} \tag{A.1}$$

where r refers to any of the other clusters left unchanged in the current step and n_x is the number of configurations in cluster x. This is repeated until only one cluster remains which now contains all configurations. Afterward, one can re-order the configurations according to the cluster hierarchy obtained in the fusing process, and draw a color-coded visualization of the distance matrix such as in Fig. 3.13 or Fig. 7.7.

A.4 Statistical significance of the Bhattacharyya distance measure

The Bhattacharyya distance measure (BDM) was introduced in Chapter 6 and used in the discussion of entropy effects of the minimum free-energy distribution of RNA secondary structures. Furthermore it provided a measure for the violation of the microcanonical property of the ParQ algorithm. This issue was discussed in Sec. 7.3.4. In this appendix the Monte Carlo method that allows one to estimate the statistical significance of the BDM is described.

Let X be a discrete random variable with possible outcomes between 1 and k. We want to test the hypothesis that X is described by a certain model, the so called "null-model". This model states that X is described by the PMF p_0.

Let $x_1 \dots x_n$ be the outcomes of an experiment on the random variable X and define the empirical histogram as

$$\hat{p}(i) = \frac{1}{n} \sum_{j=1}^{n} \delta_{i,x_j}.$$

The BDM between two probability mass functions p and q was introduced in Eq. (6.1). Here, we measure the BDM between the empirical distribution $\hat{p}$ and p_0,

$$\hat{B} \equiv B(\hat{p}||p_0) = \sum_{i=1}^{k} \sqrt{\hat{p}(i)} \cdot \sqrt{p_0(i)}.$$

If $\hat{p}(i) = p_0(i)$ for all $i = 1 \dots k$, one would obtain a BDM of 1. Finite samples hardly reach a BDM of 1 even though the null hypothesis was correct. Deviations from 1 strongly depend on k and n.

It is possible to assess a p-value of the observation $\hat{B}$. This is the probability that a BDM of $\hat{B}$ or smaller occurred by pure chance under the assumption that the null model is true. To define this more precisely, let $Y_1 \dots Y_n$ be a random vector where each Y_i is described by p_0. Under this conditions the BDM is also a random variable

$$B = \sum_{i=1}^{k} \sqrt{\frac{1}{n} \sum_{j=1}^{n} \delta_{i,Y_j}} \cdot \sqrt{p_0(i)}.$$

The p-value for the observed BDM $\hat{B}$ is defined as

$$\text{p-value} := \text{Prob}\left[B \leq \hat{B}\right].$$

procedure bdm_update($y[1 \dots n]$,$h[1 \dots k]$,$B_{\max}$)
begin
$B \leftarrow \sum_{i=1}^{k} \sqrt{h[i]} \cdot \sqrt{p_0(i)}$
$j \leftarrow$ random integer between 1 and n
$y^* \leftarrow$ choose random integer between 1 and k with probabilities $p_0(\cdot)$
$B^* \leftarrow B + (\sqrt{h[y[j]] - 1} - \sqrt{h[y[j]]})\sqrt{p_0(y[j])}$
$\quad + (\sqrt{h[y^*] + 1} - \sqrt{h[y^*]})\sqrt{p_0(y^*)}$
if $B^* \leq B_{\max}$ **then**
$B \leftarrow B^*$
$y[j] \leftarrow y^*$
$h[y[j]] \leftarrow h[y[j]] - 1$
$h[y^*] \leftarrow h[y^*] + 1$
end
return (y,h,B)
end

Algorithm A.4.1: Monte Carlo update procedure for the BDM p-value algorithm. It requires the threshold value $B_{\max}$, a working histogram $h[1 \dots k]$ and a working array $y[1 \dots n]$, where each $y[i]$ can be integers between 1 and k. It returns a modified working array, the histogram and the current BDM.

One would accept the model if the p-value was large enough which depends on how conservative the test is desired to be.

By randomization it is possible to compute a p-value for an observed $\hat{B}$ with fixed n and k. One generates independent histograms according to the null model (realizations of the random vector Y_i) and counts the fraction of events, where the BDM is smaller than or equal to $\hat{B}$ [Sco04]. If the p-value is very small this simple sampling method becomes infeasible very quickly. For those cases one may implement an importance sampling approach. Here the method of Wilbur [Wil98] is discussed. It is a Monte Carlo method designed to approximate very small p-values[1].

The method is based on Markov chain Monte Carlo that aims at sampling the random vector $Y_1 \dots Y_n$ from the distribution $(p_0)^n$. The so constructed chain is denoted as $(y_1^{(0)} \dots y_n^{(0)}), \dots, (y_1^{(m)} \dots y_n^{(m)})$, where m is the number of samples. Initially the vector $y_1^{(0)}, \dots y_n^{(0)}$ is drawn randomly from the distribution $(p_0)^n$. At each time step $t \geq 1$, a new vector $y_1^*, \dots, y_n^*$ is proposed by a local modification of the previous vector $y_1^{(t-1)}, \dots, y_n^{(t-1)}$. This is done by choosing an index $1 \leq j^* \leq n$ randomly and then replacing the j^**th* component with some discrete random number y^* drawn from p_0, i.e.

$$y_1^*, \dots, y_n^* = y_1^{(t-1)}, \dots, y_{j^*-1}^{(t-1)}, y^*, y_{j^*+1}^{(t-1)}, \dots, y_n^{(t-1)}.$$

This proposal yields the new BDM

$$B^* := \sum_{i=1}^{k} \sqrt{\frac{1}{n} \sum_{j=1}^{n} \delta_{i,y_j^*}} \cdot \sqrt{p_0(i)}.$$

[1] A related sampling technique in the context of free-energy barriers, *successive umbrella sampling*, was proposed by Virnau and Müller [VM04]

This proposal is accepted, if B^* is smaller than a certain threshold value $B_{\max}$, which is specified below. In that case $y_1^*, \ldots, y_n^*$ is used in the next time step as usual. The update procedure is illustrated in Algorithm A.4.1.

The simulation consists of two stages. In the first one a series of threshold values $B_{\max}^{(0)} \ldots B_{\max}^{(l_{\max})}$ is determined iteratively. In each of those iterations, that are labeled as $0, \ldots, (l_{\max} - 1)$, a predefined number of Monte Carlo steps is performed as described above, where $B_{\max}^{(0)}$ is set to 1 in the first iteration. After each iteration the threshold value of the next iteration is set to the median of the visited BDMs of the current iteration. This is repeated until the BDM that has to be tested, i.e. $\hat{B}$, is smaller than the 25% quantile of the simulated BDMs. This happens in the iteration labeled by $l_{\max} - 1$. The last threshold value is set to $\hat{B}$ instead of the median, i.e. $B_{\max}^{(l_{\max})} = \hat{B}$.

In the second stage, the iterations $0 \ldots (l_{\max} - 1)$ are repeated with the fixed threshold values $B_{\max}^{(0)}, \ldots, B_{\max}^{(l_{\max}-1)}$ that have been determined in the first stage. In order to achieve a better accuracy, it is possible to choose a larger number of Monte Carlo steps per iteration than in the first stage. In each iteration the fraction of events $\hat{f}_l$ with $B \leq B_{\max}^{(l+1)}$ is measured. The series $\hat{f}_0, \ldots, \hat{f}_{l_{\max}-1}$ yields to a Monte Carlo approximation of the p-value,

$$\text{p-value} \approx \prod_{l=0}^{l_{\max}-1} \hat{f}_l.$$

This approach is very general and could in principle also be applied to the local sequence alignment statistics as it was discussed in Chapter 4. But to my believe the parallel tempering or the Wang-Landau approach has better mixing properties because the random walker is allowed to travel across different score levels in two directions. The method that is described here is more simulated annealing like.

Appendix B

The +/- J spin glass: Algebraic tunneling times?

In Sec. 7.4 you have seen that the performance of the ParQ method could be increased by the reduction of frustration in a system with entropic constraints.

This result arises the question whether the principle may be adopted to other systems with high energetic or entropic barriers. In the case study which is presented in this appendix, we consider the two-dimensional $\pm J$ spin glass, which is a prototype of glassy systems with quenched disorder [EA75, BY86, MPV87, You98, FH93]. As already mentioned in Chapter 7, this model is characterized by large sample-to-sample fluctuations, which results in extremely broad tunneling time distributions of generalized ensemble Monte Carlo algorithms. In contrast to the RNA secondary structure (the results of Sec. 7.3.3), typical tunneling times grow exponential with the system size [DTW$^+$04]. The aim of the present study is to check whether this performance limitation can be overcome by extended state spaces similar as in Sec. 7.4.

In the following two sections, the model including its extension will be briefly introduced and, after that, the resulting convergence properties and tunneling time distributions are presented.

B.1 The Edwards-Anderson Hamiltonian

The state space of the two-dimensional $\pm J$ Ising spin glass is a set of Ising spins $\{\sigma_i\}$, i.e. variables that only have $+1$ or -1 as possible values (orientations). In the geometry which is chosen here, these spins sit on the sites of a rectangular lattice with periodic boundary conditions in both directions (see Fig. B.1(a)). Each of the $M = 2N$ bonds of this lattice is assigned a variable $J_{i,j} \in \{+1, -1\}$.

Bonds with $J_{i,j} = +1$ are denoted as *ferromagnetic*, those with $J_{i,j} = -1$ as *anti-ferromagnetic*. In terms of statistical mechanics of disordered systems, the set of random bonds referres to the realizations of the disorder, similar as the molecular sequence in the model of the RNA secondary structure.

The Hamiltonian of the model is given by

$$E(\sigma) := -\sum_{\langle i,j \rangle} \sigma_i J_{i,j} \sigma_j,$$

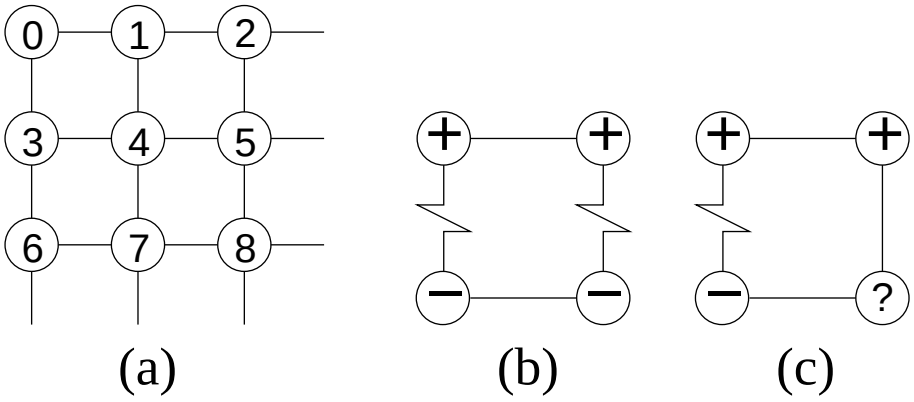


Figure B.1:
(a) Geometry of the two-dimensional Ising spin-glass with periodic boundary conditions. The circles indicate the spins, which are enumerated from 0 to $L^2 - 1$.
(b) A non-frustrated plaquette. Ferromagnetic bonds are shown as straight lines and anti-ferromagnetic bonds as rugged lines. Each bond can be satisfied, i.e. spins that are connected by ferromagnetic (anti-ferromagnetic) bonds have equal (opposite) orientation.
(c) In a frustrated plaquette one bond cannot be satisfied.

where the sum runs over all nearest neighbors. This kind of energy function is referred as Edwards Anderson Hamiltonian with $\pm J$ interaction. In contrast to the model of the RNA secondary structure, there are no entropic constraints, i.e. all 2^N spin configurations are possible states for every realization of the disorder. The frustration is more of energetic nature, due to so called frustrated plaquettes.

A plaquette is closed path on the lattice consisting of four bonds (see Fig. B.1 (b) and (c)). A plaquette is referred as *frustrated*, if the product of the four bond variables equals -1. Spins connected by ferromagnetic bonds energetically favor to take the same orientation and those connected by anti-ferromagnetic bonds prefer to take the opposite orientation. Therefore all bonds of a frustrated plaquette cannot be "satisfied" at the same time.

The $J_{i,j}$ had been drawn from the bimodal distribution

$$P(J) = \frac{1}{2}\left(\delta_{J,-1} + \delta_{J,1}\right).$$

Since the aim is to study the generalized ensemble method in extended state spaces, it is desirable to know the exact DOS for each realization of the disorder as a reference. For that purpose, I used the algorithm of Saul and Kardar [SK94]. This method requires that the number of frustrated and non-frustrated plaquettes equals, which is also the case in the thermodynamic limit. Hence, only such realizations had been considered. In figure Fig. B.2, the DOS for different realizations of system sizes from $L = 4$ to $L = 20$ are illustrated. The data have been produced by the original implementation by Saul and Kardar.

If the number of spins is even, the energy changes its sign under a simultaneous spin flip of every second spin. Hence the DOS is symmetric and it is enough to consider only negative energies. Hence, the maximal possible energy value, $E_{\max}$, is either 0

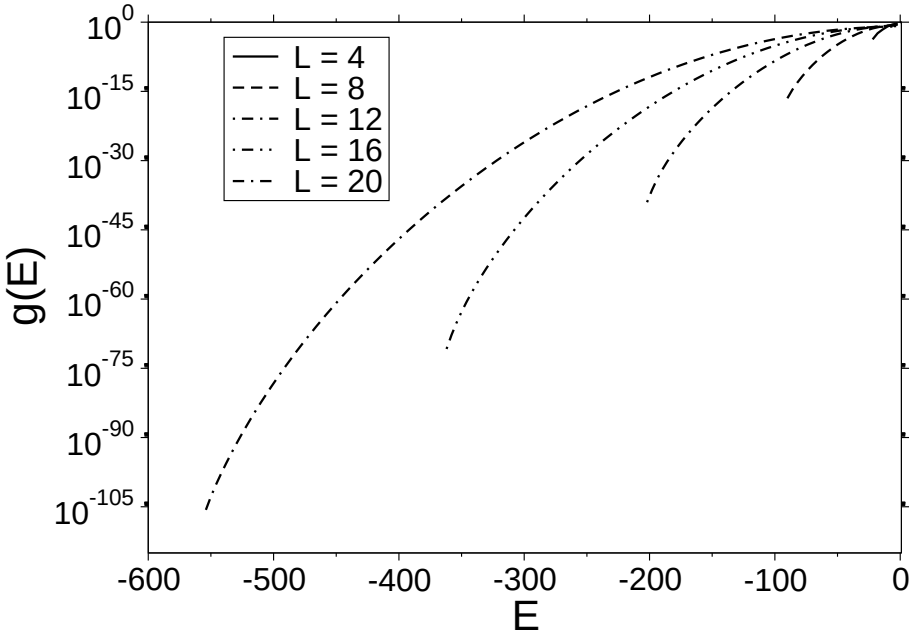


Figure B.2: The DOS of different realizations of $\pm J$ spin-glasses in two dimensions on square lattice obtained by the algorithm by Saul and Kardar [SK94]

or -2 depending on the bond configuration **J**.

B.2 Extension of the state space

In order to remove frustration from the system in a similar but not equivalent way as described in Sec. 7.4, I considered all M_a anti-ferromagnetic bonds as additional degrees of freedom. This means they are allowed to change from ferromagnetic to anti-ferromagnetic. This allows one to interpolate between the original spin glass and a ferromagnet, where all anti-ferromagnetic bonds have flipped. In this case the frustration is removed completely.

Let V denote the magnetization like observable, that measures the amount of changed bonds. We shall denoted this macroscopic quantity as *violation*. The original system has $V = 0$ and the ferromagnet $V = M_a$. Similar as for the RNA secondary structure, we are aiming at simulating the joint density of states $g(E, V)$. The normalized DOS of the original system is then obtained by the projection

$$\hat{g}(E) = \frac{1}{\sum_{E'} g(E', 0)} g(E, 0)$$

The performance is measured by the tunneling times from the state $E_{\max}$ and $V = 0$ to the ground state of the original system E_0 and $V = 0$ in the generalized ensemble.

In order to measure the performance I implemented three variants of the simulation program,

- the Wang-Landau algorithm,

class number	σ_i	number of neighbors with $J_{i,j}\sigma_j = 1$	ΔE	new class
0	+	0	-8	5
1	+	1	-4	6
2	+	2	0	7
3	+	3	4	8
4	+	4	8	9
5	-	0	8	0
6	-	1	4	1
7	-	2	0	2
8	-	3	-4	3
9	-	4	-8	4

class number	$J^*_{i,j}$	σ_i, σ_j	ΔE	new class
0	$+$	$+,+$ or $-,-$	2	2
1	$+$	$+,-$ or $-,+$	-2	3
2	$-$	$+,+$ or $-,-$	-2	0
3	$-$	$+,-$ or $-,+$	2	1

Table B.1: Energy classes for spins and bonds. The table shows a class number, the current spin or bond value, the local environment of the object, the energy change a flip would cause and the class identifier of the spin/bond after a possible flip.

- the generalized ensemble method with Metropolis updates (see. Sec. 2.7) and
- the generalized ensemble method with n-fold way updates (see. Sec. 2.3).

All three algorithms employ weights $w(E, V)$ on a two-dimensional domain. For the n-fold way, all $N + M_a$ variables are divided in only $10+4$ classes, that are characterized by the energy change that a flip would cause. These classes are listed in Tab. B.1 for the spin and bond variables. For Metropolis updates one selects one of the $N + M_a$ variables at random, performs a trial flip, which is accepted with the acceptance criterion given by Eq. (2.4),

$$\alpha = \min\left\{1, \frac{w(E + \Delta E, V + \Delta V)}{w(E, V)}\right\},$$

where ΔE and ΔV are the changes of energy and violation that the flip would cause. When using the n-fold way dynamics, all proposals are accepted and one accounts for the waiting times a spin flip would cause (see Sec. 2.3).

Spin flips may change the energy by multiples of 4 and the energy may change by ± 2 due to bond flips (see Tab. B.1). A spin flip leaves the violation V unchanged, whereas V is increased or decreased by one by a bond flip. This means, only certain macro states are in principle possible. Only those states have to be taken into account, when checking the flatness of the histogram in the Wang-Landau algorithm. Furthermore the state space is restricted to $E \geq E_0$, even for states with $V > 0$. The largest possible value of V is of course M_a. The closeup sketch in Fig. B.3 shows possible states and jumps close to the ground state. However, a doubling of the number of variables is not desirable in systems where the state space grows exponentially with the

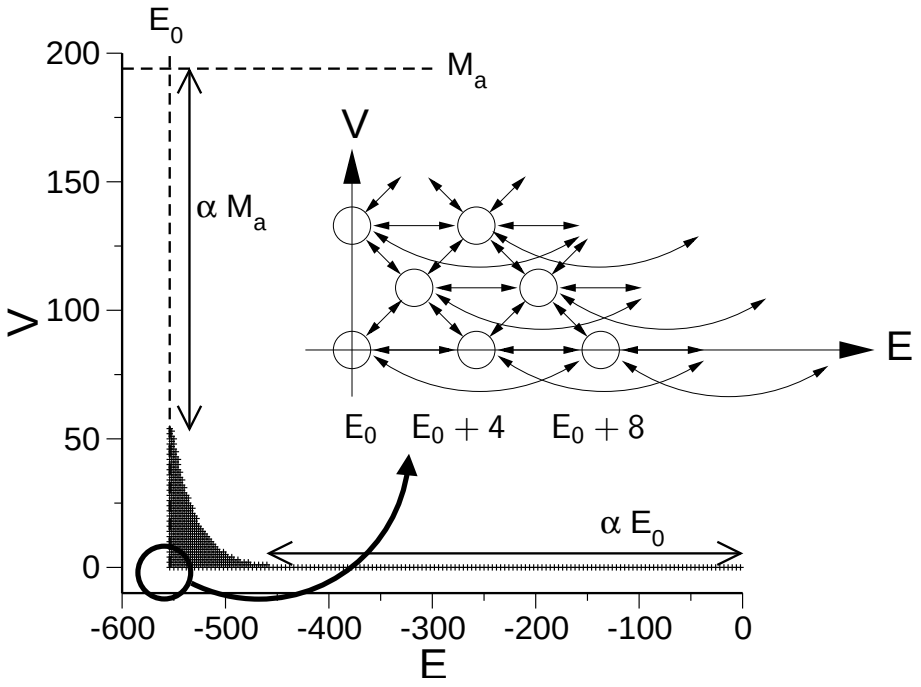


Figure B.3: Allowed macro states in the $E - V$ plane.
closeup: possible states and jumps close to the ground state.

number of variables. Therefore, the set of possible macro states are further reduced. We allow only states that have zero violation $V = 0$ or that sit in the $E - V$ plane below the curve

$$V_{\max}(E) = (1 - \alpha)M_a \cdot \exp[-B(E - E_0)]. \tag{B.1}$$

$\alpha \in [0, 1]$ is a tunable parameter and $B > 0$ is chosen such that $V_{\max}((1 - \alpha)E_0) = 1$,

$$B = -\frac{\log\left[(1 - \alpha)M_a\right]}{\alpha E_0}.$$

Furthermore one has to guarantee that each allowed macro state is reachable from the ground state by a chain of the $10 + 4$ jumps listed in Tab. B.1, i.e. the states have to be ergodic. The states in the $E - V$ plane after this kind of restriction are illustrated in Fig. B.3. The limit $\alpha = 0$ corresponds to all discrete states on the full rectangular support $E_0 \leq E \leq 0$ and $0 \leq V \leq M_a$ that are reachable from the ground state. The other extreme case $\alpha = 1$ restricts the allowed states to the $(V = 0)$-axis, which corresponds to the original state space.

In the following section, the performance of the Monte Carlo dynamics in the extended state space is discussed.

B.3 Performance in the extended state space

In a first step, I generated the generalized ensemble weights $w(E, V)$ for a small number of realizations between $L = 4$ and $L = 20$ by the Wang-Landau algorithm (10

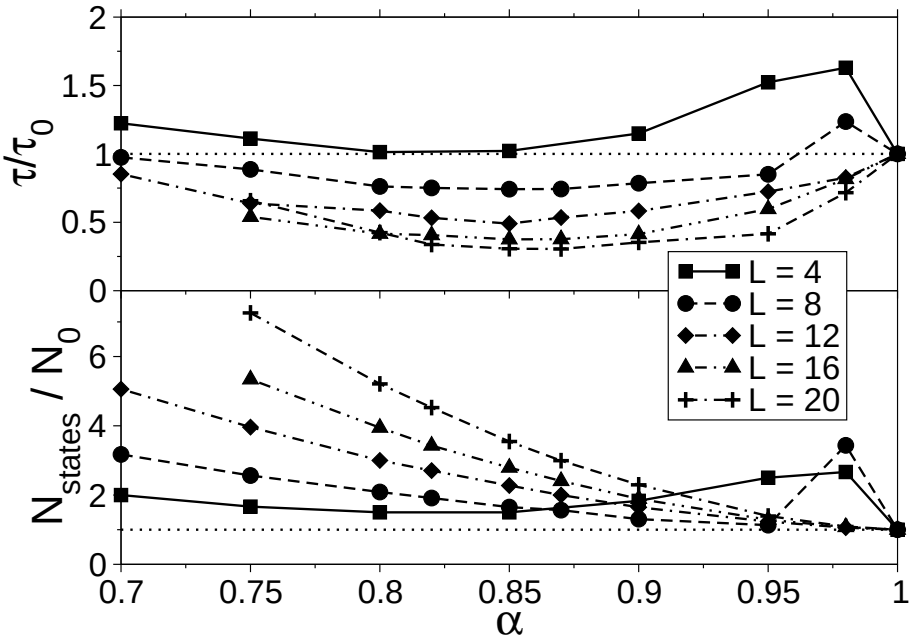


Figure B.4: Top: Performance of the extended state-space method in comparison to the standard method measured by the ratio of tunneling times as a function of α. Bottom: Ratio of number of macro states of the extended state space to the number of energy levels of the original system.

realization per system size). The paramater α that controls the size the of state space was chosen between 0.7 and 1.0 for $L < 16$ and 0.75 for $L = 16$ and $L = 20$. The parameters for the Wang-Landau algorithms were tuned as $\epsilon^{WL} = 0.6$, $\log \phi^0 = 0.1$ and $\log \phi^{\text{final}} = 1 \times 10^{-8}$. For $\alpha = 1$ (the original problem) the exact DOS were used to determine the weights as $w(E) = 1/g(E)$. For each realization and several values of α, a generalized ensemble simulation with n-fold way dynamics was performed and the corresponding tunneling times were measured.

The resulting tunneling times τ_0 for the case $\alpha = 1$ are used as reference and the performance of the extended state space method is measured as the ratio of the observed tunneling time τ for $\alpha < 1$ to the reference τ_0 as a function of α (see Fig. B.4). Obviously, there is a local optimum at $\alpha \approx 0.85$. Possibly, this also a global optimum. However, I also experimented with restrictions where the boundary function $V_{\max}$ defined in Eq. (B.1) was changed by choosing another value of B, such that $V_{\max}(E) = 1$ was shifted towards the ground state, while the vertical offset was kept fixed. I found no significant enhancement.

In realistic applications, where the DOS is not known, it is required to have "efficient" weights such that all states are visited with equal probability. Hence, also the methods to obtain those weights, i.e. the Wang-Landau algorithm in this case, have to be efficient. For example, if the tunneling time decreases by a factor of 2, but the Wang-Landau iteration requires four times as much computational effort as in the original state-space, nothing is gained. Therefore I also checked the dynamics of the Wang Landau iteration with $\alpha = 0.85$. For this purpose, I picked out an "easy" and an "hard"

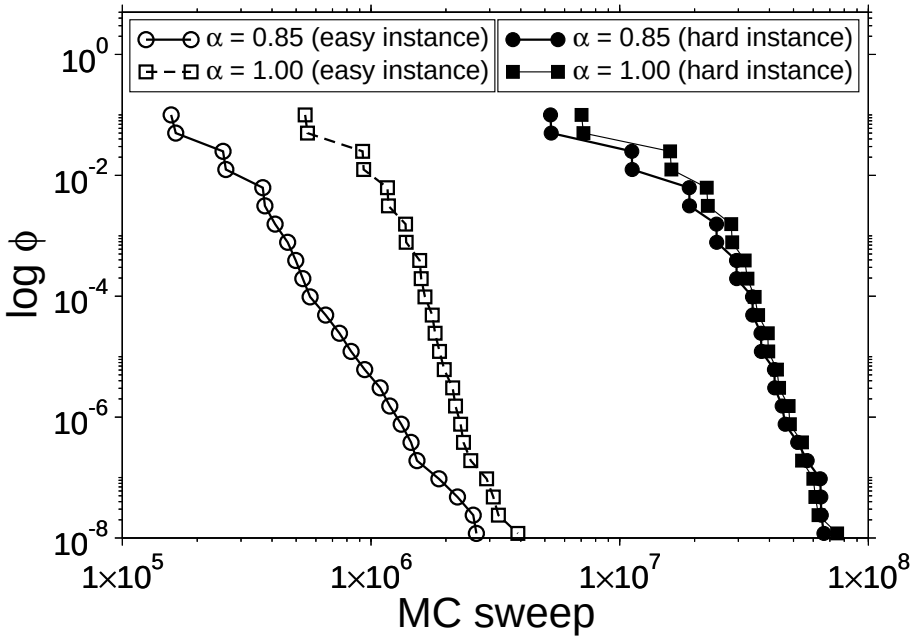


Figure B.5: Convergence of the Wang-Landau iteration for extended state space $\alpha = 0.85$ and the standard case $\alpha = 1$. The plot displays the number of Monte Carlo sweeps required to achieve different modification factors $\log \phi$. A sweep is measured by number of MC steps per spin variable. The data had been averaged over 10 independent runs.

realization of $L = 20$. The easy realization was chosen close to the median of the distribution of the ratio $g(E_1)/g(E_0)$. Again, E_1 denotes the energy of first excitations, which is $E_0 + 4$ for this model. The value of the ratio for the easy instance was $g(E_1)/g(E_0) \approx 972$. The hard instance was taken from the tail, where this ratio is large, i.e. $g(E_1)/g(E_0) \approx 95,651$. This ratio turned out to be the crucial measure for Monte Carlo complexity (see Sec. 7.3.3 and [DTW$^+$04]).

The result is shown in Fig. B.5, where the number of Monte Carlo sweeps vs. modification factors is displayed. Note that a sweep is defined as number of Monte Carlo steps per *spin* in both cases, i.e. number of steps / 400 for $L = 20$. Surprisingly, even though the Wang-Landau histogram is defined on a larger domain for the extended state space, the algorithm converges faster in this case.

Next, we study the rate of convergence of the estimator of the ratio $g(E_1)/g(E_0)$. As shown in Chapter 7, this quantity is very sensitive to the choice of the Monte Carlo method. Its relative error has already been defined by Eq. (7.3) in Chapter 7,

$$\epsilon_{\text{ratio}} = \frac{|g(E_1)/g(E_0) - g^{\text{exact}}(E_1)/g^{\text{exact}}(E_0)|}{g^{\text{exact}}(E_1)/g^{\text{exact}}(E_0)}.$$

Fig. B.6 illustrates the convergence of the extended state space algorithm in comparison with the standard generalized ensemble. Obviously the performance could be improved for the hard and the easy instance, where many meta-stable states prohibit a direct jump from a first excitation to the ground state in the standard algorithm. When

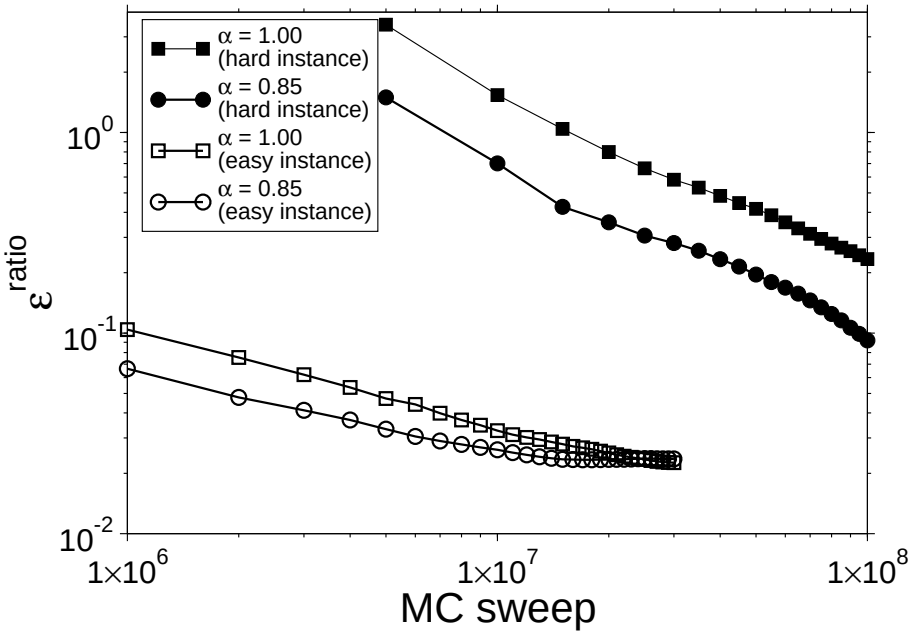


Figure B.6: Rate of convergence of the extended state space ($\alpha = 0.85$) method in comparison with the standard generalized ensemble without violations ($\alpha = 1.0$). Two realizations of $L = 20$ have been considered. The convergence is enhanced for the hard instance, where $g(E_1)/g(E_0)$ is large.

allowing extended states, the random walk is allowed to escape from such local minima resulting in smaller correlation times. Hence, with the same number of n-fold way updates, a smaller error is obtained, even though the state space is much larger than the original one.

This effect is studied in more detail by the properties of the tunneling-time distribution over ensembles of realizations of the disorder. These distributions were studied by Dayal et.al. [DTW$^+$04]. They observed that this distribution is well described by a generalized extreme-value distribution, which was introduced by Eq. (7.4) in Chapter 7,

$$\mathrm{Prob}\,(\tau \geq x) = \exp\left[-\left(1+\xi\,\frac{x-\mu}{\sigma}\right)^{-1/\xi}\right].$$

The typical tunneling time described by the location parameter μ and the scale σ grows exponentially with the system size. Here, I considered a slightly different definition of the tunneling time as in [DTW$^+$04]. Where Dayal et. al. considered the entire range energy range from the ground state E_0 to the anti-ground state $-E_0$, I made use of the symmetry of the system for an even number of spins, as described above, and measured the tunneling times only on the negative energy axis from the maximum $E_{\max}$ to the ground state and back. This causes the typical tunneling time to grow like $\exp\left[c\sqrt{L}\right]$ (instead of $\exp\left[c'L\right]$), which is still exponential. I generated 1000 realizations for each system sizes from $L = 6$ to $L = 20$. For $L = 20$ only 300 realizations were generated. For each of those realizations the generalized ensemble weights for the extended state

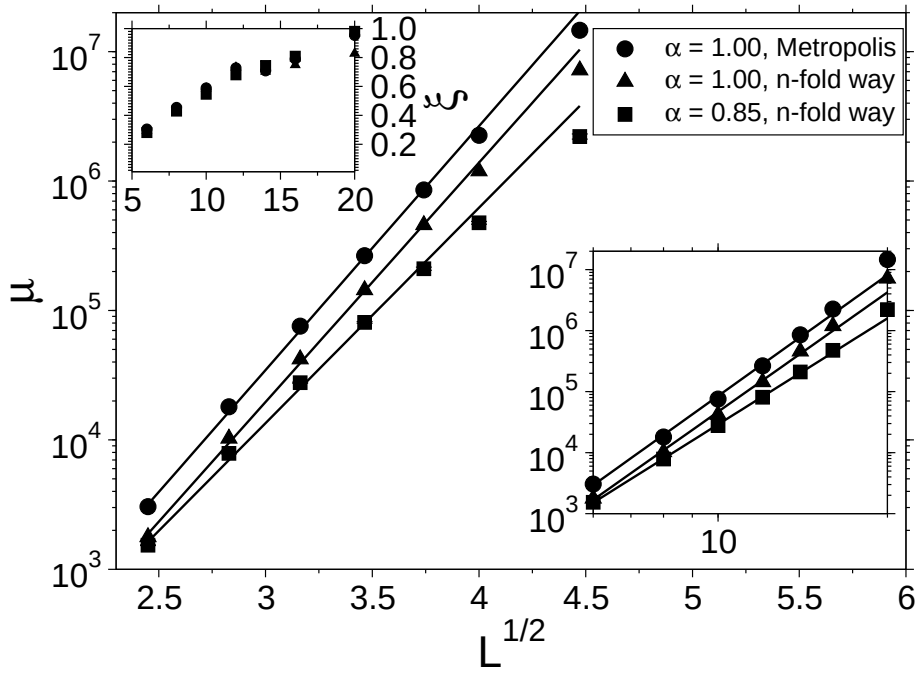


Figure B.7: Scaling of the location μ and shape ξ of the generalized extreme value distribution of tunneling times. Lines show least square fits to the exponential function Eq. (B.3). The scale parameter σ is not shown. It behaves similar as the location parameter.
Top inset: scaling of the shape parameter.
Bottom inset: the location parameter on a log-log plot scale. Lines show least square fits to the algebraic function Eq. (B.2).
Tab. B.2 displays all fit-parameters including reduced χ^2_* values.

space ($\alpha = 0.85$) were determined with the Wang-Landau algorithm. After that the empirical tunneling-time distributions for the generalized ensemble with n-fold way updates were determined. In order to decide how strong the extended state space improves the statistics of tunneling times, the same distribution for the generalized ensemble method with $\alpha = 1.0$ was obtained. Besides the n-fold way, also simulations with standard Metropolis updates were employed for this case.

In a similar way as in Chapter 7, I determined the parameters location μ, scale σ and shape ξ of the generalized extreme-value distribution by a maximum likelihood fit [Hos85, Mac89]. The scaling behavior of those parameters is shown in Fig. B.7. As one can see, the differences between the methods regarding the growth of typical tunneling times μ and the scale parameters differ significantly among different methods, but some evidence for exponential scaling for the extended state space still remains. All methods exhibit a scaling somehow between exponential and algebraic. In order to provide more quantitative evidence in either direction, I fitted the data to algebraic

$$\mu(L) = A_\mu \cdot L^{z_\mu} \quad \text{and} \quad \sigma(L) = A_\sigma \cdot L^{z_\sigma} \tag{B.2}$$

algebraic fit	z_μ	A_μ	χ^2_*	z_σ	$A_\sigma/10^6$	χ^2_*
$\alpha = 1.0$						
Metropolis	6.5(1)	0.022(6)	63.3	7.2(2)	0.003(1)	36.1
$\alpha = 1.0$						
n-fold way	6.5(1)	0.014(4)	72.2	7.2(2)	0.015(6)	28.9
$\alpha = 0.85$						
n-fold way	5.78(6)	0.048(6)	19.6	6.4(1)	0.005(1)	11.3
exponential fit	c_μ	$B_\mu/10^6$	χ^2_*	c_σ	$B_\sigma/10^6$	χ^2_*
$\alpha = 1.0$						
Metropolis	4.34(7)	0.07(1)	36.3	4.63(5)	0.016(2)	5.8
$\alpha = 1.0$						
n-fold way	4.26(8)	0.05(1)	47.7	4.57(6)	0.010(2)	8.6
$\alpha = 0.85$						
n-fold way	3.8(1)	0.13(4)	131.7	4.04(7)	0.031(7)	10.1

Table B.2: Fit parameters of the weighted chi-square fits of the parameters location μ and scale σ of the generalized extreme value distribution to the algebraic function Eq. (B.2) and the exponential function Eq. (B.3). Weighted chi squared values χ^2_* are included.

and exponential functions

$$\mu(L) = B_\mu \cdot \exp[c_\mu \sqrt{L}] \quad \text{and} \quad \sigma(L) = B_\sigma \cdot \exp[c_\sigma \sqrt{L}] \tag{B.3}$$

by a weighted chi-square fit. The resulting fit parameters including the weighted chi-square values χ^2_* are summarized in Tab. B.2. The data, in particular the χ^2_* values, suggests that, for the extended state space, the characteristic exponential growth is less likely than an algebraic growth.

To summarize the results of this appendix, I have considered an extension of the state space of $\pm J$ Ising spin glasses, in order to check whether the performance can be increased. Regarding the performance of the Wang-Landau algorithm and the generalized ensemble method, the answer is in principle positive as can be seen by the convergence properties, especially for hard instances. Typical tunneling times are reduced by a factor of about 3.2 for the largest system and some evidence for algebraic growth of the tunneling times is found by a fit. However, an exponential growth cannot be safely excluded. Hence, in a future work it would be of interest whether the improvement remains for larger systems than $L = 20$ or even three-dimensional systems, where the exact density of states cannot be computed efficiently.

Appendix C

Fit parameters

This appendix summarizes fit parameters for the alignment statistics discussed in Chapter 4.

α	$L_{\mathtt{Q}}, L_{\mathtt{S}}$	λ	$10^4\,\lambda_2$	K	S_0	χ^2_*
10	40	$0.3272 \pm 0.108\%$	$8.6347 \pm 0.412\%$	$0.1028 \pm 0.65\%$	$15.597 \pm 0.0676\%$	79.05
	60	$0.3034 \pm 0.086\%$	$6.2007 \pm 0.285\%$	$0.0751 \pm 0.60\%$	$18.455 \pm 0.0645\%$	49.40
	80	$0.2892 \pm 0.070\%$	$4.8781 \pm 0.222\%$	$0.0612 \pm 0.53\%$	$20.644 \pm 0.0540\%$	21.67
	100	$0.2747 \pm 0.072\%$	$4.3187 \pm 0.330\%$	$0.0472 \pm 0.58\%$	$22.413 \pm 0.0611\%$	39.42
	150	$0.2541 \pm 0.083\%$	$3.2974 \pm 0.529\%$	$0.0303 \pm 0.61\%$	$25.682 \pm 0.0422\%$	39.46
	200	$0.2432 \pm 0.063\%$	$2.6343 \pm 0.344\%$	$0.0241 \pm 0.52\%$	$28.257 \pm 0.0412\%$	10.47
	250	$0.2359 \pm 0.071\%$	$2.1999 \pm 0.454\%$	$0.0198 \pm 0.60\%$	$30.196 \pm 0.0459\%$	9.40
	300	$0.2303 \pm 0.061\%$	$1.9101 \pm 0.348\%$	$0.0174 \pm 0.54\%$	$31.934 \pm 0.0408\%$	2.00
	350	$0.2261 \pm 0.046\%$	$1.6404 \pm 0.239\%$	$0.0153 \pm 0.41\%$	$33.334 \pm 0.0300\%$	1.27
	400	$0.2224 \pm 0.052\%$	$1.4806 \pm 0.266\%$	$0.0136 \pm 0.49\%$	$34.556 \pm 0.0369\%$	1.36
	600	$0.2140 \pm 0.062\%$	$1.0206 \pm 0.384\%$	$0.0106 \pm 0.64\%$	$38.561 \pm 0.0472\%$	2.15
	800	$0.2090 \pm 0.063\%$	$0.7660 \pm 0.419\%$	$0.0088 \pm 0.67\%$	$41.320 \pm 0.0457\%$	1.82
12	40	$0.3366 \pm 0.117\%$	$7.9013 \pm 0.518\%$	$0.1125 \pm 0.74\%$	$15.426 \pm 0.0799\%$	34.31
	60	$0.3178 \pm 0.120\%$	$5.4247 \pm 0.484\%$	$0.0898 \pm 0.85\%$	$18.183 \pm 0.0836\%$	66.67
	80	$0.3044 \pm 0.085\%$	$4.2388 \pm 0.299\%$	$0.0715 \pm 0.61\%$	$20.123 \pm 0.0513\%$	31.31
	100	$0.2987 \pm 0.087\%$	$3.2541 \pm 0.557\%$	$0.0663 \pm 0.65\%$	$21.748 \pm 0.0498\%$	39.77
	150	$0.2896 \pm 0.081\%$	$1.9120 \pm 1.049\%$	$0.0562 \pm 0.63\%$	$24.663 \pm 0.0350\%$	32.49
	200	$0.2843 \pm 0.060\%$	$1.4542 \pm 0.639\%$	$0.0512 \pm 0.51\%$	$26.822 \pm 0.0282\%$	4.58
	250	$0.2815 \pm 0.055\%$	$0.9651 \pm 1.138\%$	$0.0487 \pm 0.47\%$	$28.492 \pm 0.0207\%$	6.61
	300	$0.2761 \pm 0.075\%$	$0.8401 \pm 1.177\%$	$0.0423 \pm 0.69\%$	$29.858 \pm 0.0354\%$	1.49
	350	$0.2754 \pm 0.064\%$	$0.7118 \pm 1.017\%$	$0.0420 \pm 0.59\%$	$31.030 \pm 0.0265\%$	1.13
	400	$0.2715 \pm 0.054\%$	$0.6569 \pm 0.769\%$	$0.0374 \pm 0.53\%$	$32.034 \pm 0.0270\%$	1.26
14	40	$0.3406 \pm 0.132\%$	$7.5641 \pm 0.593\%$	$0.1191 \pm 0.89\%$	$15.411 \pm 0.1074\%$	161.70
	60	$0.3233 \pm 0.150\%$	$5.1473 \pm 0.636\%$	$0.0958 \pm 1.08\%$	$18.072 \pm 0.1091\%$	119.48
	80	$0.3132 \pm 0.134\%$	$3.9083 \pm 0.529\%$	$0.0828 \pm 0.96\%$	$20.031 \pm 0.0739\%$	52.61
	100	$0.3118 \pm 0.089\%$	$2.7370 \pm 0.661\%$	$0.0833 \pm 0.71\%$	$21.570 \pm 0.0560\%$	46.20
	150	$0.3080 \pm 0.070\%$	$1.3077 \pm 1.383\%$	$0.0790 \pm 0.57\%$	$24.296 \pm 0.0298\%$	17.41
	200	$0.3039 \pm 0.061\%$	$0.9113 \pm 1.283\%$	$0.0743 \pm 0.53\%$	$26.313 \pm 0.0252\%$	4.03
	250	$0.3021 \pm 0.044\%$	$0.5421 \pm 1.745\%$	$0.0727 \pm 0.39\%$	$27.878 \pm 0.0169\%$	1.69
	300	$0.2995 \pm 0.058\%$	$0.4089 \pm 2.364\%$	$0.0691 \pm 0.55\%$	$29.162 \pm 0.0248\%$	1.43
	350	$0.2982 \pm 0.038\%$	$0.4102 \pm 1.308\%$	$0.0668 \pm 0.37\%$	$30.212 \pm 0.0151\%$	0.76
	400	$0.2964 \pm 0.044\%$	$0.3697 \pm 1.424\%$	$0.0635 \pm 0.46\%$	$31.122 \pm 0.0232\%$	1.12
16	40	$0.3423 \pm 0.145\%$	$7.4174 \pm 0.624\%$	$0.1207 \pm 0.98\%$	$15.376 \pm 0.1177\%$	127.67
	60	$0.3266 \pm 0.145\%$	$4.9889 \pm 0.631\%$	$0.1019 \pm 1.02\%$	$18.079 \pm 0.0925\%$	96.12
	80	$0.3158 \pm 0.136\%$	$3.7888 \pm 0.548\%$	$0.0852 \pm 1.10\%$	$19.956 \pm 0.1110\%$	102.43
	100	$0.3192 \pm 0.099\%$	$2.4005 \pm 0.864\%$	$0.0951 \pm 0.77\%$	$21.480 \pm 0.0526\%$	38.51
	150	$0.3172 \pm 0.070\%$	$1.0875 \pm 1.574\%$	$0.0963 \pm 0.57\%$	$24.216 \pm 0.0246\%$	22.37
	200	$0.3142 \pm 0.052\%$	$0.6405 \pm 1.585\%$	$0.0923 \pm 0.45\%$	$26.141 \pm 0.0175\%$	1.87
	250	$0.3117 \pm 0.056\%$	$0.4447 \pm 1.701\%$	$0.0885 \pm 0.54\%$	$27.647 \pm 0.0283\%$	5.08
	300	$0.3108 \pm 0.041\%$	$0.3838 \pm 2.118\%$	$0.0871 \pm 0.38\%$	$28.848 \pm 0.0119\%$	0.32
	350	$0.3091 \pm 0.034\%$	$0.2300 \pm 2.586\%$	$0.0845 \pm 0.34\%$	$29.910 \pm 0.0147\%$	0.39
	400	$0.3085 \pm 0.028\%$	$0.1676 \pm 2.554\%$	$0.0838 \pm 0.30\%$	$30.808 \pm 0.0146\%$	0.43
∞	40	$0.3457 \pm 0.141\%$	$7.2030 \pm 0.712\%$	$0.1298 \pm 0.86\%$	$15.437 \pm 0.0773\%$	132.94
	60	$0.3281 \pm 0.151\%$	$4.8936 \pm 0.679\%$	$0.1040 \pm 1.07\%$	$18.060 \pm 0.0983\%$	128.56
	80	$0.3165 \pm 0.163\%$	$3.7511 \pm 0.634\%$	$0.0866 \pm 1.28\%$	$19.959 \pm 0.1208\%$	95.18
	100	$0.3224 \pm 0.125\%$	$2.3191 \pm 1.007\%$	$0.1020 \pm 0.94\%$	$21.485 \pm 0.0514\%$	11.77
	150	$0.3255 \pm 0.039\%$	$0.8343 \pm 1.162\%$	$0.1150 \pm 0.32\%$	$24.139 \pm 0.0078\%$	0.48
	200	$0.3262 \pm 0.032\%$	$0.3662 \pm 3.326\%$	$0.1219 \pm 0.30\%$	$26.029 \pm 0.0164\%$	0.90
	250	$0.3216 \pm 0.064\%$	$0.3022 \pm 3.597\%$	$0.1113 \pm 0.62\%$	$27.510 \pm 0.0260\%$	2.57
	300	$0.3248 \pm 0.016\%$	-	$0.1234 \pm 0.49\%$	$28.684 \pm 0.0503\%$	5.33
	350	$0.3241 \pm 0.009\%$	-	$0.1233 \pm 0.22\%$	$29.690 \pm 0.0207\%$	2.49
	400	$0.3220 \pm 0.013\%$	-	$0.1167 \pm 0.39\%$	$30.541 \pm 0.0371\%$	2.09

Table C.1: Fit parameters of the modified Gumbel distribution Eq. (4.4) for the classical i.i.d. model using the BLOSUM62 scoring matrix and different affine costs α and $\beta = 1$.

α	$L_{\mathtt{Q}}, L_{\mathtt{S}}$	λ	$10^4\ \lambda_2$	K	S_0	χ^2_*
11	40	0.2615 ± 0.154%	5.5852 ± 0.727%	0.0995 ± 0.90%	19.387 ± 0.0882%	96.36
	60	0.2373 ± 0.150%	4.4494 ± 0.612%	0.0729 ± 0.97%	23.477 ± 0.0903%	114.50
	80	0.2227 ± 0.149%	3.6508 ± 0.554%	0.0585 ± 1.01%	26.610 ± 0.0815%	123.28
	100	0.2099 ± 0.139%	3.3410 ± 0.560%	0.0468 ± 1.02%	29.293 ± 0.0906%	79.64
	150	0.1920 ± 0.116%	2.5551 ± 0.400%	0.0320 ± 0.90%	34.266 ± 0.0713%	35.65
	200	0.1825 ± 0.110%	2.0098 ± 0.355%	0.0262 ± 0.98%	38.115 ± 0.0875%	19.07
	250	0.1729 ± 0.110%	1.7807 ± 0.356%	0.0203 ± 1.06%	41.312 ± 0.1004%	18.50
	300	0.1671 ± 0.072%	1.5683 ± 0.254%	0.0168 ± 0.64%	43.796 ± 0.0505%	3.63
	350	0.1608 ± 0.073%	1.4261 ± 0.237%	0.0135 ± 0.70%	46.080 ± 0.0593%	3.78
	400	0.1584 ± 0.052%	1.2870 ± 0.184%	0.0125 ± 0.44%	48.005 ± 0.0269%	0.82
13	40	0.2664 ± 0.117%	5.4163 ± 0.680%	0.1037 ± 0.66%	19.186 ± 0.0550%	57.54
	60	0.2464 ± 0.079%	4.0672 ± 0.369%	0.0824 ± 0.51%	23.101 ± 0.0435%	26.45
	80	0.2336 ± 0.079%	3.2958 ± 0.352%	0.0682 ± 0.56%	26.023 ± 0.0461%	24.76
	100	0.2228 ± 0.065%	2.8870 ± 0.327%	0.0559 ± 0.48%	28.399 ± 0.0407%	12.07
	150	0.2085 ± 0.042%	2.1084 ± 0.203%	0.0414 ± 0.34%	32.797 ± 0.0269%	6.13
	200	0.1999 ± 0.039%	1.6797 ± 0.172%	0.0337 ± 0.34%	36.039 ± 0.0269%	1.57
	250	0.1930 ± 0.030%	1.4174 ± 0.160%	0.0273 ± 0.29%	38.553 ± 0.0237%	1.49
	300	0.1891 ± 0.040%	1.1849 ± 0.216%	0.0248 ± 0.37%	40.779 ± 0.0263%	1.86
	350	0.1852 ± 0.044%	1.0489 ± 0.257%	0.0219 ± 0.40%	42.617 ± 0.0255%	1.80
	400	0.1834 ± 0.058%	0.9090 ± 0.353%	0.0204 ± 0.54%	44.100 ± 0.0323%	2.06
15	40	0.2706 ± 0.090%	5.1148 ± 0.527%	0.1115 ± 0.53%	19.158 ± 0.0480%	32.81
	60	0.2520 ± 0.051%	3.7530 ± 0.273%	0.0898 ± 0.33%	22.935 ± 0.0261%	12.85
	80	0.2403 ± 0.046%	2.9795 ± 0.214%	0.0758 ± 0.32%	25.738 ± 0.0252%	7.17
	100	0.2315 ± 0.036%	2.5361 ± 0.213%	0.0644 ± 0.28%	27.938 ± 0.0245%	4.12
	150	0.2185 ± 0.035%	1.8127 ± 0.199%	0.0492 ± 0.30%	32.074 ± 0.0264%	4.21
	200	0.2107 ± 0.056%	1.4225 ± 0.292%	0.0404 ± 0.50%	35.072 ± 0.0376%	3.14
	250	0.2074 ± 0.051%	1.0914 ± 0.378%	0.0374 ± 0.49%	37.394 ± 0.0367%	4.17
	300	0.2038 ± 0.050%	0.9403 ± 0.380%	0.0340 ± 0.50%	39.377 ± 0.0378%	2.96
	350	0.2017 ± 0.054%	0.7930 ± 0.410%	0.0318 ± 0.57%	41.001 ± 0.0426%	1.68
	400	0.2014 ± 0.056%	0.6437 ± 0.520%	0.0314 ± 0.59%	42.326 ± 0.0414%	1.05
17	40	0.2734 ± 0.089%	4.8288 ± 0.468%	0.1166 ± 0.54%	19.130 ± 0.0544%	47.17
	60	0.2551 ± 0.053%	3.5604 ± 0.292%	0.0947 ± 0.34%	22.861 ± 0.0253%	12.72
	80	0.2442 ± 0.044%	2.8332 ± 0.287%	0.0811 ± 0.30%	25.605 ± 0.0210%	4.65
	100	0.2370 ± 0.036%	2.3250 ± 0.267%	0.0715 ± 0.27%	27.733 ± 0.0210%	4.37
	150	0.2256 ± 0.043%	1.6046 ± 0.262%	0.0574 ± 0.38%	31.750 ± 0.0305%	5.45
	200	0.2195 ± 0.058%	1.2004 ± 0.379%	0.0499 ± 0.51%	34.622 ± 0.0341%	2.42
	250	0.2170 ± 0.054%	0.9041 ± 0.511%	0.0470 ± 0.54%	36.797 ± 0.0411%	4.89
	300	0.2144 ± 0.051%	0.7326 ± 0.470%	0.0441 ± 0.50%	38.652 ± 0.0334%	3.01
	350	0.2136 ± 0.054%	0.5908 ± 0.605%	0.0440 ± 0.56%	40.238 ± 0.0351%	2.37
	400	0.2131 ± 0.062%	0.4726 ± 0.828%	0.0437 ± 0.63%	41.541 ± 0.0350%	2.03
∞	40	0.2737 ± 0.078%	4.9254 ± 0.494%	0.1163 ± 0.46%	19.091 ± 0.0420%	30.06
	60	0.2586 ± 0.032%	3.3695 ± 0.183%	0.1016 ± 0.22%	22.827 ± 0.0178%	2.82
	80	0.2500 ± 0.026%	2.5857 ± 0.133%	0.0921 ± 0.18%	25.517 ± 0.0128%	1.87
	100	0.2439 ± 0.049%	2.0502 ± 0.394%	0.0840 ± 0.38%	27.606 ± 0.0287%	7.82
	150	0.2341 ± 0.072%	1.3787 ± 0.537%	0.0707 ± 0.61%	31.490 ± 0.0423%	14.51
	200	0.2324 ± 0.074%	0.9453 ± 0.649%	0.0709 ± 0.66%	34.209 ± 0.0386%	3.20
	250	0.2327 ± 0.066%	0.5876 ± 0.828%	0.0744 ± 0.62%	36.294 ± 0.0323%	4.97
	300	0.2331 ± 0.042%	0.3915 ± 0.718%	0.0773 ± 0.42%	37.957 ± 0.0204%	1.32
	350	0.2330 ± 0.037%	0.2084 ± 1.565%	0.0792 ± 0.36%	39.395 ± 0.0120%	0.48
	400	0.2324 ± 0.030%	0.1296 ± 3.598%	0.0786 ± 0.29%	40.620 ± 0.0081%	0.32

Table C.2: Fit parameters of the modified Gumbel distribution Eq. (4.4) for the classical i.i.d. model using the `PAM250` scoring matrix and affine gap costs α and $\beta = 1$.

		FQPS			corresponding RQGS		
L_{Q}	L_{S}	λ	$10^4\lambda_2$	K	λ	$10^4\lambda_2$	K
P08100	50				0.3016 ±0.40%	7.5741 ±0.77%	0.0654 ±3.34%
348	100	0.1747 ±0.19%	3.2202 ±0.32%	0.0132 ±1.49%	0.2829 ±0.17%	3.6884 ±0.36%	0.0463 ±4.09%
	200	0.1617 ±0.09%	1.7968 ±0.18%	0.0100 ±1.31%	0.2685 ±0.15%	1.8498 ±0.40%	0.0315 ±2.77%
	300	0.1478 ±0.14%	1.3962 ±0.21%	0.0059 ±2.20%	0.2664 ±0.14%	1.1900 ±0.47%	0.0292 ±3.49%
	320	0.1466 ±0.15%	1.3775 ±0.28%	0.0056 ±2.33%	0.2674 ±0.11%	1.1059 ±0.51%	0.0295 ±2.05%
	348	0.1432 ±0.22%	1.4131 ±0.33%	0.0051 ±2.69%	0.2681 ±0.10%	0.9909 ±0.43%	0.0307 ±2.18%
	360	0.1426 ±0.17%	1.4322 ±0.22%	0.0047 ±3.17%	0.2678 ±0.10%	0.9883 ±0.42%	0.0302 ±2.49%
	400	0.1418 ±0.10%	1.4201 ±0.17%	0.0047 ±1.43%	0.2648 ±0.12%	1.0238 ±0.50%	0.0248 ±3.89%
	500	0.1399 ±0.26%	1.4517 ±0.35%	0.0043 ±3.94%	0.2638 ±0.17%	1.0248 ±0.65%	0.0255 ±5.65%
	600	0.1405 ±0.16%	1.4392 ±0.20%	0.0047 ±2.87%	0.2650 ±0.14%	0.9917 ±0.74%	0.0245 ±3.85%
P50052	50				0.3024 ±0.85%	7.4294 ±1.70%	0.0657 ±6.19%
363	100	0.1795 ±0.16%	3.1869 ±0.26%	0.0132 ±1.42%	0.2818 ±0.25%	3.6993 ±0.55%	0.0458 ±3.44%
	200	0.1660 ±0.18%	1.8701 ±0.30%	0.0096 ±1.98%	0.2698 ±0.21%	1.8027 ±0.58%	0.0341 ±4.60%
	300	0.1550 ±0.22%	1.3995 ±0.36%	0.0066 ±2.97%	0.2643 ±0.14%	1.2232 ±0.42%	0.0273 ±3.55%
	330	0.1512 ±0.12%	1.4130 ±0.23%	0.0057 ±1.30%	0.2654 ±0.18%	1.0822 ±0.68%	0.0274 ±5.32%
	363	0.1509 ±0.18%	1.3881 ±0.27%	0.0057 ±3.53%	0.2687 ±0.24%	0.9676 ±1.00%	0.0332 ±7.75%
	380	0.1489 ±0.12%	1.4138 ±0.19%	0.0051 ±1.17%	0.2651 ±0.30%	0.9806 ±1.28%	0.0270 ±11.76%
	400	0.1474 ±0.20%	1.4335 ±0.32%	0.0048 ±3.27%	0.2634 ±0.15%	0.9773 ±0.75%	0.0271 ±11.41%
	500	0.1471 ±0.08%	1.4350 ±0.16%	0.0049 ±1.13%	0.2613 ±0.21%	0.9998 ±1.05%	0.0226 ±7.60%
	600	0.1457 ±0.28%	1.4640 ±0.54%	0.0046 ±3.24%	0.2662 ±0.15%	0.9498 ±0.79%	0.0250 ±7.76%
Q18179	50				0.3008 ±0.70%	7.6673 ±1.23%	0.0625 ±5.34%
455	100	0.1798 ±0.33%	3.7190 ±0.59%	0.0103 ±2.84%	0.2845 ±0.16%	3.5814 ±0.35%	0.0485 ±2.86%
	200	0.1723 ±0.16%	1.9839 ±0.32%	0.0087 ±1.50%	0.2685 ±0.14%	1.8391 ±0.49%	0.0302 ±3.81%
	300	0.1609 ±0.25%	1.4302 ±0.40%	0.0059 ±4.49%	0.2632 ±0.16%	1.2382 ±0.53%	0.0262 ±4.69%
	420	0.1569 ±0.27%	1.3665 ±0.52%	0.0050 ±2.90%	0.2636 ±0.17%	0.8441 ±0.59%	0.0222 ±9.17%
	450	0.1590 ±0.25%	1.3225 ±0.61%	0.0052 ±2.86%	0.2611 ±0.13%	0.8203 ±0.43%	0.0209 ±4.93%
	455	0.1548 ±0.26%	1.4038 ±0.52%	0.0049 ±2.76%	0.2655 ±0.12%	0.7670 ±0.49%	0.0246 ±8.35%
	480	0.1557 ±0.38%	1.3664 ±0.67%	0.0051 ±7.10%	0.2610 ±0.10%	0.7929 ±0.41%	0.0197 ±6.70%
	500	0.1521 ±0.45%	1.4145 ±0.77%	0.0044 ±5.30%	0.2615 ±0.17%	0.7783 ±0.62%	0.0204 ±5.09%
	600	0.1540 ±0.25%	1.3886 ±0.43%	0.0043 ±3.72%	0.2596 ±0.14%	0.7706 ±0.60%	0.0174 ±5.71%
P35348	50				0.3046 ±0.61%	7.3443 ±1.17%	0.0668 ±4.85%
466	100	0.1809 ±0.18%	3.1996 ±0.28%	0.0135 ±2.06%	0.2839 ±0.22%	3.6314 ±0.49%	0.0465 ±2.49%
	200	0.1625 ±0.12%	1.8687 ±0.18%	0.0079 ±1.63%	0.2696 ±0.15%	1.8030 ±0.48%	0.0315 ±3.97%
	300	0.1643 ±0.10%	1.2089 ±0.15%	0.0086 ±2.23%	0.2620 ±0.13%	1.2472 ±0.47%	0.0241 ±5.52%
	400	0.1510 ±0.24%	1.2641 ±0.39%	0.0051 ±2.76%			
	450	0.1521 ±0.33%	1.2357 ±0.55%	0.0050 ±5.39%	0.2647 ±0.16%	0.7874 ±0.67%	0.0246 ±3.93%
	466	0.1485 ±0.17%	1.2982 ±0.35%	0.0046 ±2.93%			
	480	0.1517 ±0.23%	1.2359 ±0.34%	0.0056 ±5.27%	0.2609 ±0.25%	0.7981 ±1.25%	0.0207 ±9.36%
	500	0.1492 ±0.22%	1.2845 ±0.35%	0.0048 ±3.64%	0.2668 ±0.09%	0.7124 ±0.49%	0.0265 ±6.00%
	600	0.1509 ±0.28%	1.2383 ±0.40%	0.0050 ±3.86%			

Table C.3: Fit parameters λ, λ_2 and K of the modified Gumbel distribution for (FQPS) and (RQGS).

		HMM n=0			HMM n=1		
L_Q	L_S	λ	$10^4\lambda_2$	$10^3 K$	λ	$10^4\lambda_2$	$10^3 K$
348	150	0.2890 ±0.85%		49.4722 ±7.27%	0.2310 ±9.32%		21.4600 ±66.56%
	200	0.2894 ±2.84%		50.0796 ±24.47%	0.2274 ±1.74%		20.1017 ±13.25%
	300	0.2895 ±2.69%		53.3472 ±24.00%	0.2240 ±4.86%		17.8934 ±37.22%
	348	0.2988 ±3.24%		72.2356 ±30.15%	0.2234 ±2.39%		16.8704 ±18.79%
	360	0.2895 ±1.79%		51.9056 ±16.04%	0.2220 ±2.14%		16.3757 ±16.52%
	400	0.2859 ±3.49%		48.4496 ±31.10%	0.2232 ±2.40%		17.5141 ±18.94%
	500	0.2912 ±6.63%		54.0687 ±61.22%	0.2182 ±2.39%		14.7371 ±19.10%
	600	0.2901 ±3.38%		51.9412 ±31.74%	0.2180 ±2.59%		14.2439 ±20.86%
		HMM n=2			HMM n=3		
L_Q	L_S	λ	$10^4\lambda_2$	K	λ	$10^4\lambda_2$	K
348	150	0.1968 ±0.70%	2.9247 ±1.37%	12.0400 ±6.48%	0.1767 ±0.44%	2.6797 ±1.01%	7.4435 ±3.72%
	200	0.1947 ±2.12%		9.8704 ±14.29%	0.1795 ±0.46%	2.3586 ±0.92%	8.5733 ±3.87%
	300	0.1937 ±3.60%		9.9597 ±25.32%	0.1863 ±0.41%	2.0008 ±0.94%	11.7859 ±5.63%
	348	0.1888 ±3.19%		8.1338 ±22.42%	0.1876 ±0.32%	1.9328 ±0.89%	12.1223 ±3.83%
	360	0.1926 ±3.17%		9.7957 ±22.82%	0.1853 ±0.27%	1.9530 ±0.65%	10.8640 ±2.65%
	400	0.1934 ±1.05%		9.9321 ±8.22%	0.1757 ±1.64%		7.1756 ±11.58%
	500	0.1919 ±1.61%		9.3630 ±12.32%	0.1783 ±0.98%		7.7945 ±7.18%
	600	0.1912 ±1.70%		9.3303 ±13.25%	0.1768 ±1.01%		7.4165 ±8.19%
		HMM n=4			HMM n=5		
L_Q	L_S	λ	$10^4\lambda_2$	$10^3 K$	λ	$10^4\lambda_2$	$10^3 K$
348	150	0.1732 ±0.47%	2.2119 ±1.14%	7.4991 ±6.08%	0.1710 ±0.38%	2.0698 ±0.92%	8.1950 ±3.70%
	200	0.1686 ±0.28%	2.1187 ±0.72%	6.4162 ±3.14%	0.1657 ±0.39%	1.8231 ±1.14%	6.9148 ±3.82%
	300	0.1682 ±0.36%	1.9635 ±0.79%	6.5436 ±4.22%	0.1599 ±0.37%	1.7836 ±0.79%	5.4451 ±3.85%
	348	0.1685 ±0.35%	1.9408 ±0.74%	7.3851 ±3.34%	0.1580 ±0.28%	1.7930 ±0.68%	5.3049 ±2.61%
	360	0.1678 ±0.42%	1.9421 ±0.92%	6.5775 ±4.07%	0.1605 ±0.23%	1.7481 ±0.50%	5.7512 ±2.89%
	400	0.1662 ±0.18%	1.9782 ±0.40%	6.4164 ±2.32%	0.1587 ±0.28%	1.7828 ±0.73%	5.4513 ±2.57%
	500	0.1693 ±0.24%	1.9047 ±0.51%	7.0735 ±2.11%	0.1587 ±0.16%	1.7957 ±0.40%	5.4770 ±2.31%
	600	0.1693 ±0.17%	1.8994 ±0.39%	7.1112 ±2.06%	0.1575 ±0.29%	1.8330 ±0.58%	5.2125 ±2.68%

Table C.4: The table shows the fit parameters of the score distribution $\mathrm{Prob}(S = s|$ # of helices = n$)$ for $0 \leq n \leq 5$ for $L_Q = 348$ and different subject lengths. For entries, where λ_2 is left out, a suitable fit (with a small reduced χ^2 value) to the modified Gumbel distribution Eq. (4.4) was not possible and only the Gumbel parameters of the high probability region are shown.

		HMM n=0			HMM n=1		
		HMM n=6			HMM n=7		
L_Q	L_S	λ	$10^4\lambda_2$	$10^3 K$	λ	$10^4\lambda_2$	$10^3 K$
348	150	0.1663 ±0.49%	2.1403 ±1.04%	7.9392 ±5.83%	0.1646 ±0.30%	2.1396 ±0.65%	8.7088 ±4.21%
	200	0.1614 ±0.25%	1.7767 ±0.65%	6.7568 ±2.30%	0.1574 ±0.41%	1.7687 ±1.17%	6.5219 ±3.81%
	300	0.1551 ±0.28%	1.5986 ±0.80%	5.2551 ±3.18%	0.1514 ±0.26%	1.4638 ±0.62%	5.0238 ±4.34%
	348	0.1531 ±0.20%	1.5993 ±0.55%	4.9132 ±2.71%	0.1482 ±0.33%	1.4755 ±0.77%	4.4535 ±4.13%
	360	0.1536 ±0.34%	1.6036 ±1.02%	4.9160 ±3.41%	0.1490 ±0.39%	1.4479 ±0.93%	4.6858 ±3.28%
	400	0.1537 ±0.27%	1.5713 ±0.62%	4.9524 ±3.05%	0.1494 ±0.24%	1.4328 ±0.70%	4.6867 ±2.08%
	500	0.1519 ±0.23%	1.6229 ±0.67%	4.6812 ±2.14%	0.1472 ±0.29%	1.4706 ±0.63%	4.2881 ±2.50%
	600	0.1489 ±0.15%	1.7148 ±0.33%	4.2283 ±2.16%	0.1460 ±0.18%	1.5193 ±0.49%	4.2679 ±1.74%
		HMM n=8			HMM n=9		
L_Q	L_S	λ	$10^4\lambda_2$	$10^3 K$	λ	$10^4\lambda_2$	$10^3 K$
348	150	0.1595 ±0.47%	2.2162 ±1.01%	7.5355 ±4.01%	0.1603 ±0.23%	2.1517 ±0.48%	8.0273 ±2.17%
	200	0.1534 ±0.55%	1.8019 ±1.46%	5.9224 ±5.25%	0.1508 ±0.14%	1.7854 ±0.28%	6.3535 ±1.89%
	300	0.1473 ±0.47%	1.3916 ±1.24%	4.8483 ±4.01%	0.1413 ±0.12%	1.4118 ±0.35%	4.2141 ±1.43%
	348	0.1458 ±0.32%	1.3409 ±0.85%	4.6141 ±3.69%	0.1398 ±0.10%	1.3281 ±0.33%	3.9661 ±1.44%
	360	0.1469 ±0.34%	1.2868 ±0.90%	4.9271 ±2.73%	0.1400 ±0.16%	1.2888 ±0.43%	4.0126 ±1.79%
	400	0.1440 ±0.34%	1.3591 ±1.05%	4.0064 ±3.48%	0.1382 ±0.25%	1.2954 ±0.67%	3.7257 ±2.14%
	500	0.1433 ±0.29%	1.3382 ±0.85%	3.9952 ±2.70%	0.1352 ±0.14%	1.3472 ±0.42%	3.1780 ±1.68%
	600	0.1416 ±0.33%	1.3760 ±0.94%	3.7782 ±3.14%	0.1359 ±0.13%	1.3399 ±0.38%	3.3536 ±1.49%
		HMM n=10			HMM n=11		
L_Q	L_S	λ	$10^4\lambda_2$	$10^3 K$	λ	$10^4\lambda_2$	$10^3 K$
348	150	0.1552 ±0.14%	2.2225 ±0.30%	6.7936 ±2.08%	0.1455 ±0.14%	2.3813 ±0.15%	4.9660 ±3.82%
	200	0.1459 ±0.22%	1.8336 ±0.37%	5.7585 ±3.30%	0.1417 ±0.17%	1.8428 ±0.35%	5.1264 ±2.07%
	300	0.1370 ±0.22%	1.4024 ±0.56%	3.8087 ±1.79%	0.1324 ±0.27%	1.3842 ±0.68%	3.2129 ±2.79%
	348	0.1353 ±0.15%	1.2962 ±0.38%	3.5507 ±1.68%	0.1316 ±0.22%	1.2518 ±0.69%	3.1546 ±1.94%
	360	0.1343 ±0.13%	1.2830 ±0.36%	3.4674 ±1.39%	0.1297 ±0.25%	1.2737 ±0.52%	2.9445 ±2.81%
	400	0.1334 ±0.16%	1.2602 ±0.38%	3.2164 ±1.71%	0.1302 ±0.20%	1.2160 ±0.56%	2.9704 ±1.59%
	500	0.1307 ±0.16%	1.3013 ±0.46%	2.8331 ±1.22%	0.1280 ±0.30%	1.2426 ±0.86%	2.7433 ±2.73%
	600	0.1305 ±0.23%	1.3097 ±0.56%	2.8239 ±1.82%	0.1257 ±0.22%	1.2908 ±0.55%	2.4921 ±1.79%

Table C.5: The table shows the fit parameters of the score distribution $\mathrm{Prob}(S = s|$ # of helices = n$)$ for $6 \leq n \leq 11$ for $L_Q = 348$ and different subject lengths. For entries, where λ_2 is left out, a suitable fit (with a small reduced χ^2 value) to the modified Gumbel distribution Eq. (4.4) was not possible and only the Gumbel parameters of the high probability region are shown.

Appendix D

List of acronyms

BDM	Bhattacharyya distance measure
BLAST	Basic Local Alignment Search Tool
BLOSUM	BLOcks of Amino Acid SUbstitution Matrix
CE	Combinatorial extension
DDBJ	DNA Data Bank of Japan
DNA	DeoxyriboNucleic Acid
DOS	Density Of States
DPRM	Directed Paths in Random Media
EMBL	European Molecular Biology Laboratory
EXP	EXPonential schedule
FQPS	Fixed Query - Position-dependent Scoring
HMM	Hidden Markov Model
INSDC	International Nucleotide Sequence Database Collaboration
i.i.d.	identically and independent distributed
INV	INVerse schedule
LIN	LINear schedule
MC	Monte Carlo
MCMC	Markov chain Monte Carlo
MCMCMC	Metropolis Coupled Markov Chain Monte Carlo
mRNA	messenger RNA
PAM	Point Accepted Mutation

ParQ	Parallel Q
PDB	Protein Data Bank
PHAT	Predicted Hydrophobic And Transmembrane
PSI-BLAST	Position Specific Iterative BLAST
RNA	Ribosomal Nucleic Acid
RQGS	Random Query - General-purpose Scoring
rRNA	ribosomal RNA
SLIM	Scorematrix Leading to Intra-Membrane domains
TM	TransMembrane
TMHMM	TransMembrane Hidden Markov Model
TrEMBL	Translated from EMBL
UniProtKB	Universal Protein Resource Knowledgebase

Bibliography

[ABM04] A. Andreanov, F. Barbieri, and O. C. Martin. Large deviations in spin-glass ground-state energies. *The European Physical Journal B - Condensed Matter and Complex Systems*, 41(3):365–375, October 2004.

[ABOH01] S. F. Altschul, R. Bundschuh, R. Olsen, and T. Hwa. The estimation of statistical parameters for local alignment score distributions. *Nucleic Acids Res.*, 29(2):351–361, 2001.

[ACH+00] Mark D. Adams, Susan E. Celniker, Robert A. Holt, Cheryl A. Evans, Jeannine D. Gocayne, Peter G. Amanatides, Steven E. Scherer, Peter W. Li, Roger A. Hoskins, Richard F. Galle, Reed A. George, Suzanna E. Lewis, Stephen Richards, Michael Ashburner, Scott N. Henderson, Granger G. Sutton, Jennifer R. Wortman, Mark D. Yandell, Qing Zhang, Lin X. Chen, Rhonda C. Brandon, Yu-Hui C. Rogers, Robert G. Blazej, Mark Champe, Barret D. Pfeiffer, Kenneth H. Wan, Clare Doyle, Evan G. Baxter, Gregg Helt, Catherine R. Nelson, George L. Gabor Miklos, Josep F. Abril, Anna Agbayani, Hui-Jin An, Cynthia Andrews-Pfannkoch, Danita Baldwin, Richard M. Ballew, Anand Basu, James Baxendale, Leyla Bayraktaroglu, Ellen M. Beasley, Karen Y. Beeson, P. V. Benos, Benjamin P. Berman, Deepali Bhandari, Slava Bolshakov, Dana Borkova, Michael R. Botchan, John Bouck, Peter Brokstein, Phillipe Brottier, Kenneth C. Burtis, Dana A. Busam, Heather Butler, Edouard Cadieu, Angela Center, Ishwar Chandra, J. Michael Cherry, Simon Cawley, Carl Dahlke, Lionel B. Davenport, Peter Davies, Beatriz de Pablos, Arthur Delcher, Zuoming Deng, Anne Deslattes Mays, Ian Dew, Suzanne M. Dietz, Kristina Dodson, Lisa E. Doup, Michael Downes, Shannon Dugan-Rocha, Boris C. Dunkov, Patrick Dunn, Kenneth J. Durbin, Carlos C. Evangelista, Concepcion Ferraz, Steven Ferriera, Wolfgang Fleischmann, Carl Fosler, Andrei E. Gabrielian, Neha S. Garg, William M. Gelbart, Ken Glasser, Anna Glodek, Fangcheng Gong, J. Harley Gorrell, Zhiping Gu, Ping Guan, Michael Harris, Nomi L. Harris, Damon Harvey, Thomas J. Heiman, Judith R. Hernandez, Jarrett Houck, Damon Hostin, Kathryn A. Houston, Timothy J. Howland, Ming-Hui Wei, Chinyere Ibegwam, Mena Jalali, Francis Kalush, Gary H. Karpen, Zhaoxi Ke, James A. Kennison, Karen A. Ketchum, Bruce E. Kimmel, Chinnappa D. Kodira, Cheryl Kraft, Saul Kravitz, David Kulp, Zhongwu Lai, Paul Lasko, Yiding Lei, Alexander A. Levitsky, Jiayin Li, Zhenya Li, Yong Liang, Xiaoying Lin, Xiangjun Liu, Bettina Mattei, Tina C. McIntosh, Michael P. McLeod, Duncan McPherson,

Gennady Merkulov, Natalia V. Milshina, Clark Mobarry, Joe Morris, Ali Moshrefi, Stephen M. Mount, Mee Moy, Brian Murphy, Lee Murphy, Donna M. Muzny, David L. Nelson, David R. Nelson, Keith A. Nelson, Katherine Nixon, Deborah R. Nusskern, Joanne M. Pacleb, Michael Palazzolo, Gjange S. Pittman, Sue Pan, John Pollard, Vinita Puri, Martin G. Reese, Knut Reinert, Karin Remington, Robert D. C. Saunders, Frederick Scheeler, Hua Shen, Bixiang Christopher Shue, Inga Sidn-Kiamos, Michael Simpson, Marian P. Skupski, Tom Smith, Eugene Spier, Allan C. Spradling, Mark Stapleton, Renee Strong, Eric Sun, Robert Svirskas, Cyndee Tector, Russell Turner, Eli Venter, Aihui H. Wang, Xin Wang, Zhen-Yuan Wang, David A. Wassarman, George M. Weinstock, Jean Weissenbach, Sherita M. Williams, Trevor Woodage, Kim C. Worley, David Wu, Song Yang, Q. Alison Yao, Jane Ye, Ru-Fang Yeh, Jayshree S. Zaveri, Ming Zhan, Guangren Zhang, Qi Zhao, Liansheng Zheng, Xiangqun H. Zheng, Fei N. Zhong, Wenyan Zhong, Xiaojun Zhou, Shiaoping Zhu, Xiaohong Zhu, Hamilton O. Smith, Richard A. Gibbs, Eugene W. Myers, Gerald M. Rubin, and J. Craig Venter. The Genome Sequence of Drosophila melanogaster. *Science*, 287(5461):2185–2195, 2000.

[AF01] S. Auer and D. Frenkel. Prediction of absolute crystal-nucleation rate in hard-sphere colloids. *Nature*, 409:1020–1023, 2001.

[AG96] S.F. Altschul and W. Gish. Local alignment statistics. *Meth. Enzym.*, 266:460, 1996.

[AHM+88] B. Andresen, K. H. Hoffmann, K. Mosegaard, J. Nulton, J. M. Pedersen, and P. Salamon. On Lumped Models for Thermodynamic Properties of Simulated Annealing Problems. *J. Phys. (France)*, 49:1485, 1988.

[AJL+08] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Molecular Biology of the Cell*. TAYLOR & FRANCIS, 5th edition, 2008.

[AMI+04] M. Arai, H. Mitsuke, M. Ikeda, J.-X. Xia, T. Kikuchi, M. Satake, and T. Shimizu. ConPred II: a consensus prediction method for obtaining transmembrane topology models with high reliability. *Nucleic Acids Res.*, 32(suppl 2):W390–393, 2004.

[AMS+97] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, 25(17):3389–3402, 1997.

[ATHT04] S. Adler, S. Trebst, A. K. Hartmann, and M. Troyer. Dynamics of the Wang-Landau algorithm and complexity of rare events for the three-dimensional bimodal Ising spin glass. *J. Stat. Mech.*, 2004(07):P07008, 2004.

[AW94] R. Arratia and M.S. Waterman. A phase transition for the score in matching random sequences allowing deletions. *Ann. Appl. Prob.*, 4:200–225, 1994.

[BDSN81] J. Brosius, T. J. Dull, D. D. Sleeter, and H. F. Noller. Gene organization and primary structure of a ribosomal RNA operon from Escherichia coli. *J. Mol. Biol.*, 148(2):107–127, May 1981.

[BH00] R. Bundschuh and T. Hwa. An analytic study of the phase transition line in local sequence alignment with gaps. *Discr. Appl. Math.*, 104(1-3):113–142, August 2000.

[BH02a] R. Bundschuh and T. Hwa. Phases of the secondary structures of RNA sequences. *Europhys. Lett.*, 59(6):903–909, 2002.

[BH02b] R. Bundschuh and T. Hwa. Statistical mechanics of secondary structures formed by random RNA sequences. *Phys. Rev. E*, 65(3):031903, 2002.

[BH05] B. Burghardt and A. K. Hartmann. Dependence of RNA secondary structure on the energy model. *Phys. Rev. E*, 71:021913, 2005.

[Bha43] A. Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bull. Cal. Math. Soc.*, 35:99–110, 1943.

[BIB+97] F. R. Blattner, G. Plunkett III, N. T. Bloch, C. A.and Perna, V. Burland, M. Riley, J. Collado-Vides, J. D. Glasner, C. K. Rode, G. F. Mayhew, J. Gregor, N. W. Davis, H. A. Kirkpatrick, M. A. Goeden, D. J. Rose, B. Mau, and Y. Shao. The Complete Genome Sequence of Escherichia coli K-12. *Science*, 277(5331):1453–1462, 1997.

[BKL75] A. B. Bortz, M. H. Kalos, and J. L. Lebowitz. A new algorithm for Monte Carlo simulation of Ising spin systems. *J. Comp. Phys.*, 17(1):10–18, January 1975.

[BLA] Ncbi blast, http://blast.ncbi.nlm.nih.gov.

[BN92] B. A. Berg and T. Neuhaus. Multicanonical ensemble: A new approach to simulate first-order phase transitions. *Phys. Rev. Lett.*, 68:9, 1992.

[BY86] K. Binder and A. P. Young. Spin glasses: Experimental facts, theoretical concepts, and open questions. *Rev. Mod. Phys.*, 58(4):801–976, Oct 1986.

[Car06] R. Cartwright. Logarithmic gap costs decrease alignment accuracy. *BMC Bioinformatics*, 7(1):527, 2006.

[CB05] P. Clote and R. Backofen. *Computational Molecular Biology: An Introduction*. John Wiley & Sons, Ltd., 2005.

[CC96] M. K. Cowles and B. P. Carlin. Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review. *JASA*, 91(434):883–904, 1996.

[CDS05] J. M. Cuthbertson, D. A. Doyle, and M. S. P. Sansom. Transmembrane helix prediction: a comparative evaluation and analysis. *Protein Eng.*, 18(6):295–308, 2005.

[CFKK05] P. Clote, F. Ferré, E. Kranakis, and D. Krizanc. Structural RNA has lower folding energy than random RNA of the same dinucleotide frequency. *RNA*, 11(5):578–591, 2005.

[CLR02] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. B&T, 2002.

[Cri66] F. H. C. Crick. Codon-Anticodon Pairing: The Wobble Hypothesis. *J. Mol. Biol.*, 19:548–555, 1966.

[CWS+97] M. Cserzo, E. Wallin, I. Simon, G. von Heijne, and A. Elofsson. Prediction of transmembrane alpha-helices in prokaryotic membrane proteins: the dense alignment surface method. *Protein Eng.*, 10(6):673–676, 1997.

[CZG81] T. R. Cech, A. J. Zaug, and P. J. Grabowski. In vitro splicing of the ribosomal RNA precursor of tetrahymena: Involvement of a guanosine nucleotide in the excision of the intervening sequence. *Cell*, 27(3):487–496, 1981.

[DDB] DNA Data Bank of Japan (DDBJ), http://www.ddbj.nig.ac.jp.

[Dev86] L. Devroye. *Non-Uniform Random Variate Generation*. Springer, New York, 1986.

[dG68] P.-G. de Gennes. Statistics of branching and hairpin helices for the dAT copolymer. *Biopolymers*, 6(5):715–729, 1968.

[DHL00] D. Drasdo, T. Hwa, and M. Lässig. Scaling Laws and Similarity Detection in Sequence Alignment with Gaps. *J. Comp. Biol.*, 7(1-2):115–141, 2000.

[DKZ94] A. Dembo, S. Karlin, and O. Zeitouni. Limit Distribution of Maximal Non-Aligned Two-Sequence Segmental Score. *Ann. Prob.*, 22:2022–2039, 1994.

[DSO78] M.O. Dayhoff, R.M. Schwartz, and B.C. Orcutt. A model of Evolutionary Change in Proteins. In M.O. Dayhoff, editor, *Atlas of Protein Sequence and Structure*, volume 5,Suppl.3, pages 345–352. National Biomedical Research Foundation, Washington,D.C., 1978.

[DTW+04] P. Dayal, S. Trebst, S. Wessel, D. Würtz, M. Troyer, S. Sabhapandit, and S. N. Coppersmith. Performance Limitations of Flat-Histogram Methods. *Phys. Rev. Lett.*, 92(9):097201–4, 2004.

[DWL+01] C. M. Deber, C. Wang, L.-P. Liu, A. S. Prior, S. Agrawal, B. L. Muskat, and A. J. Cuticchia. TM Finder: A prediction program for transmembrane protein segments using a combination of hydrophobicity and nonpolar phase helicity scales. *Protein Sci.*, 10(1):212–219, 2001.

[EA75] S. F. Edwards and P. W. Anderson. Theory of spin glasses. *J. Phys. F*, 5(5):965–974, 1975.

[ED05] D. J. Earl and M. W. Deem. Parallel tempering: Theory, applications, and new perspectives. *Phys. Chem. Chem. Phys.*, 7:3910–3916, 2005.

[EMB] European Molecular Biology Laboratory (EMBL), http://www.embl.org/.

[FFB04] F. Frommlet, A. Futschik, and M. Bogdan. On the significance of sequence alignments when using multiple scoring matrices. *Bioinformatics*, 20(6):881–887, 2004.

[FH91] D. S. Fisher and D. A. Huse. Directed paths in a random potential. *Phys. Rev. B*, 43(13):10728–, 1991.

[FH93] K. H. Fischer and J. A. Hertz. *Spin Glasses*. Cambridge University Press, 1993.

[FHS99] C. Flamm, I. L. Hofacker, and P. F. Stadler. RNA In Silico The Computational Biology of RNA Secondary Structures. *Advances in Complex Systems*, 2:65 – 90, 1999.

[FKJ+86] S. M. Freier, R. Kierzek, J. A. Jaeger, N. Sugimoto, M. H. Caruthers, T. Neilson, and D. H. Turner. Improved free-energy parameters for predictions of RNA duplex stability. *Proc Natl Acad Sci USA*, 83(24):9373?9377, 1986.

[FKM02] M. Mézard F. Krzakala and M. Müller. Nature of the glassy phase of RNA secondary structure. *Europhys. Lett.*, 57(5):752–758, 2002.

[FKSS93] W. Fontana, D. A. M. Konings, P. F. Stadler, and P. Schuster. Statistics of RNA secondary structures. *Biopolymers*, 33(9):1389–1404, 1993.

[FP89] H. Flyvbjerg and H. G. Petersen. Error estimates on averages of correlated data. *J. Chem. Phys.*, 91(1):461–466, 1989.

[FS89] A. M. Ferrenberg and R. H. Swendsen. Optimized Monte Carlo data analysis. *Phys. Rev. Lett.*, 63(12):1195, 1989.

[GBB+96] A. Goffeau, B. G. Barrell, H. Bussey, R. W. Davis, B. Dujon, H. Feldmann, F. Galibert, J. D. Hoheisel, C. Jacq, M. Johnston, E. J. Louis, H. W. Mewes, Y. Murakami, P. Philippsen, H. Tettelin, and S. G. Oliver. Life with 6000 Genes. *Science*, 274(5287):546–567, 1996.

[GCB92] G. H. Gonnet, M. A. Cohen, and S. A. Benner. Exhaustive matching of the entire protein sequence database. *Science*, 256(5062):1443–1445, 1992.

[Gey91] C. J. Geyer. Monte Carlo Maximum Likelihood for Depend Data. In *Proceedings of the 23rd Symposium on the Interface*, pages 156–163, 1991.

[GG84] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell*, 6:721–741, 1984.

[GL95] X. Gu and W.-H. Li. The size distribution of insertions and deletions in human and rodent pseudogenes suggests the logarithmic gap penalty for sequence alignment. *J. Mol. Evol.*, 40(4):464–473, 1995.

[GNU] gnuplot, http://www.gnuplot.info/.

[Got82] O. Gotoh. An improved algorithm for matching biological sequences. *J. Mol. Biol.*, 162:705, 1982.

[GTGM+83] C. Guerrier-Takada, K. Gardiner, T. Marsh, N. Pace, and S. Altman. The RNA moiety of ribonuclease P is the catalytic subunit of the enzyme. *Cell*, 35(3):849–857, 1983.

[Gum58] E. J. Gumbel. *Statistics of Extremes*. Columbia University Press, New York, 1958.

[Har01] A. K. Hartmann. Comment on "Glassy Transition in a Disordered Model for the RNA Secondary Structure". *Phys. Rev. Lett.*, 86(7):1382, 2001.

[Har02] A. K. Hartmann. Sampling rare events: Statistics of local sequence alignments. *Phys. Rev. E*, 65:056102, 2002.

[Har03] R. C. Hardison. Comparative Genomics. *PLoS Biology*, 1(2):e58, 2003.

[Har09] A. K. Hartmann. *Practical Guide to Computer Simulations*. World Scientific, 2009. to be published.

[Has70] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[HBCM98] T. Hirokawa, S. Boon-Chieng, and S. Mitaku. SOSUI: classification and secondary structure prediction system for membrane proteins. *Bioinformatics*, 14(4):378–379, 1998.

[HFS+94] I. L. Hofacker, W. Fontana, P. F. Stadler, L. S. Bonhoeffer, M. Tacker, and P. Schuster. Fast folding and comparison of RNA secondary structures. *Monatshefte für Chemie*, 125(2):167–188, 1994.

[HH64] J. M. Hammersley and D. C. Handscomb. *Monte Carlo Methods*. Chapman and Hall, London & New York, 1964.

[HH85] D. A. Huse and C. L. Henley. Pinning and Roughening of Domain Walls in Ising Systems Due to Random Impurities. *Phys. Rev. Lett.*, 54(25):2708–, 1985.

[HH92] S. Heinkoff and J.G. Heinkoff. Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci USA*, 89:10915–10919, 1992.

[HH04] J. Houdayer and A. K. Hartmann. Low-temperature behavior of two-dimensional Gaussian Ising spin glasses. *Phys. Rev. B*, 70(1):014418, 2004.

[HH05] F. Heilmann and K. H. Hoffmann. ParQ: high-precision calculation of the density of states. *Europhys. Lett.*, 70(2):155, 2005.

[Hig93] P. G. Higgs. RNA secondary structure: a comparison of real and random sequences. *J. Phys. I France*, 3:43–59, 1993.

[Hig96] P.G. Higgs. Overlaps between RNA Secondary Structures. *Phys. Rev. Lett.*, 76:704, 1996.

[Hir75] D. S. Hirschberg. A linear space algorithm for computing maximal common subsequences. *Commun. ACM*, 18(6):341–343, 1975.

[HL96] T. Hwa and M. Lässig. Similarity Detection and Localization. *Phys. Rev. Lett.*, 76(14):2591–2594, 1996.

[HL98] T. Hwa and M. Lässig. Optimal Detection of Sequence Similarity by Local Alignment. In S. Istrail, P. Pevzner, and M.S. Waterman, editors, *Proceedings of the Second Annual International Conference on Computational Molecular Biology (RECOMB98)*, page 109, 1998.

[HN96] K. Hukushima and K. Nemoto. Exchange Monte Carlo Method and Application to Spin Glass Simulations. *J. Phys. Soc. Jpn.*, 65:1604–1608, 1996.

[Hos85] J. R. M. Hosking. Algorithm AS 215: Maximum-Likelihood Estimation of the Parameters of the Generalized Extreme-Value Distribution. *Appl. Stat.*, 34(3):301–310, 1985.

[HR01] A. K. Hartmann and H. Rieger. *Optimization Algorithms in Physics*. Wiley-VCH, 2001.

[HR04] A. K. Hartmann and H. Rieger, editors. *New Optimization Algorithms in Physics*. Wiley-VCH, 2004.

[HT06] S. Hui and L.-H. Tang. Ground state and glass transition of the RNA secondary structure. *Euro. Phys. J. B*, 53(1):77–84, 2006.

[HW05] A. K. Hartmann and M. Weigt. *Phase Transitions in Combinatorial Optimization Problems. Basics, Algorithms and Statistical Mechanics*. Wiley-VCH, 2005.

[INS] International Nucleotide Sequence Database Collaboration (INSDC), http://www.insdc.org.

[Int01] International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409:860–921, 2001.

[Jan02] W. Janke. *NIC Series*, volume 10, chapter Statistical Analysis of Simulations: Data Correlations and Error Estimation, pages 423–445. John von Neumann Institut für Computing, 2002.

[JD88] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, Englewood Cliffs, NJ, 1988.

[JJS06] S. Kirkpatrick J. J. Schneider. *Stochastic Optimization*. Springer, Berlin, 2006.

[JLG02] L. Jaroszewski, W. Li, and A. Godzik. In search for more accurate alignments in the twilight zone. *Protein Sci*, 11(7):1702–1713, 2002.

[Jon07] D. T. Jones. Improving the accuracy of transmembrane protein topology prediction using evolutionary information. *Bioinformatics*, 23(5):538–544, 2007.

[JPG98] L. Jaroszewski, K. Pawlowski, and A. Godzik. Multiple Model Approach: Exploring the Limits of Comparative Modeling. *J. Mol. Model.*, 4(10):294–309, 1998.

[JTT94] D. T. Jones, W. R. Taylor, and J. M. Thornton. A mutation data matrix for transmembrane proteins. *FEBS Letters*, 339(3):269–275, 1994.

[JTZ89] J. A. Jaeger, D. H. Turner, and M. Zuker. Improved predictions of secondary structures for RNA. *Proc Natl Acad Sci USA*, 86(20):7706–7710, 1989.

[KA90] S. Karlin and S. F. Altschul. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc Natl Acad Sci USA*, 87(6):2264–2268, 1990.

[Kar87] M. Kardar. Replica Bethe ansatz studies of two-dimensional interfaces with quenched random impurities. *Nucl. Phys. B*, 290:582–602, 1987.

[Kar94] M. Kardar. Directed paths in random media. In F. David, P. Ginzparg, and J. Zinn-Justin, editors, *Fluctuating Geometries in Statistical Mechanics and Field Theory: Les Houches Summer School, Session LXII, 2 August - 9 September 1994*, 1994.

[KD92] S. Karlin and A. Dembo. Limit Distributions of Maximal Segmental Score among Markov-Dependent Partial Sums. *Adv. Appl. Prob.*, 24(1):113–140, 1992.

[KGV83] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi. Optimization by Simulated Annealing. *Science*, 220:671–680, 1983.

[KKH06] M. Körner, H. G. Katzgraber, and A. K. Hartmann. Probing tails of energy distributions using importance-sampling in the disorder with a guiding function. *J. Stat. Mech.*, page P04005, 2006.

[KKK04] R. Koike, K. Kinoshita, and A. Kidera. Probabilistic description of protein alignments for sequences and structures. *Proteins*, 56(1):157–166, 2004.

[KKL+05] H. G. Katzgraber, M. Körner, F. Liers, M. Jünger, and A. K. Hartmann. Universality-class dependence of energy distributions in spin glasses. *Phys. Rev. B*, 72(9):094421, 2005.

[KKS04] L. Käll, A. Krogh, and E. L. L. Sonnhammer. A Combined Transmembrane Topology and Signal Peptide Prediction Method. *J. Mol. Biol.*, 338(5):1027–1036, 2004.

[KKS05] L. Käll, A. Krogh, and E. L. L. Sonnhammer. An HMM posterior decoder for sequence feature prediction that includes homology information. *Bioinformatics*, 21(suppl 1):i251–257, 2005.

[KL00] M. Kschischo and M. Lässig. Finite-temperature sequence alignment. In *Pacific Symposium on Biocomputing 5*, 2000.

[KLvHS01] A. Krogh, B. Larsson, G. von Heijne, and E. L. L. Sonnhammer. Predicting transmembrane protein topology with a hidden markov model: application to complete genomes. *J. Mol. Biol.*, 305(3):567–580, 2001.

[Kni06] R. Knippers. *Molekulare Genetik*. Thieme, Stuttgart, 2006.

[KPY01] H. G. Katzgraber, M. Palassini, and A. P. Young. Monte Carlo simulations of spin glasses at low temperatures. *Phys. Rev. B*, 63:1844221–18442210, 2001.

[LB05] D. P. Landau and K. Binder. *A Guide to Monte Carlo Simulations in Statistical Physics*. Cambridge University Press, Cambridge, 2nd edition, 2005.

[Lee93] J. Lee. New Monte Carlo algorithm: Entropic sampling. *Phys. Rev. Lett.*, 71(2):211–214, 1993.

[Liu02] J. S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer, 2002.

[LV02] M. Lässig and A. Valleriani, editors. *Biological Evolution and Statistical Physics*, Berlin, 2002. Springer-Verlag.

[LW06] M. Lässig and K. J. Wiese. Freezing of Random RNA. *Phys. Rev. Lett.*, 96(22):228101, 2006.

[Mac89] A. J. Macleod. Remark AS R76: A Remark on Algorithm AS 215: Maximum-Likelihood Estimation of the Parameters of the Generalized Extreme-Value Distribution. *Appl. Stat.*, 38(1):198–199, 1989.

[MB80] I. Morgenstern and K. Binder. Magnetic correlations in two-dimensional spin-glasses. *Phys. Rev. B*, 22(1):288–303, 1980.

[McC90] J. S. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29(6-7):1105–1119, 1990.

[MEJN99] G. T. Barkema M. E. J. Newman. *Monte Carlo Methods in Statistical Physics*. Oxford Univ Press, 1999.

[Mez90] M. Mezard. On the glassy nature of random directed polymers in two dimensions. *J. Phys. France*, 51:1831, 1990.

[MG90] J. E. G. McCarthy and C. Gualerzi. Translational control of prokaryotic gene expression. *Trends in Genetics*, 6:78–85, 1990.

[MG06] C. Monthus and T. Garel. Probing the tails of the ground-state energy distribution for the directed polymer in a random medium of dimension d = 1,2,3 via a Monte Carlo procedure in the disorder. *Phys. Rev. E*, 74(5):051109, 2006.

[MG08] C. Monthus and T. Garel. Disorder-dominatedphases of random systems: relations between the tail exponents and scaling exponents. *J. Stat. Mech.*, 2008(01):P01008, 2008.

[MHS02] U. Mückstein, I. L. Hofacker, and P. F. Stadler. Stochastic pairwise alignments. *Bioinformatics*, 18(2):153–160, 2002.

[Miy95] S. Miyazawa. A reliable sequence alignment method based on probabilities of residue correspondences. *Protein Eng.*, 8(10):999–1009, 1995.

[MPRL98] E. Marinari, G. Parisi, and J. Ruiz-Lorenzo. Numerical Simulations of Spin Glass Systems. In A. Young, editor, *Spin Glasses and Random Fields, Directions in Condensed Matter Physics Vol. 12*, page 109. World Scientific, 1998.

[MPV87] M. Mézard, G. Parisi, and M. Virasoro. *Spin Glass Theory and Bbeyond An Introduction to the Replica Method and Its Applications*. World Scientific, 1987.

[MRR+53] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equation of state calculations by fast computing maschines. *J. Chem. Phys.*, 21:1087, 1953.

[MRR01] T. Müller, S. Rahmann, and M. Rehmsmeier. Non-symmetric score matrices and the detection of homologous transmembrane proteins. *Bioinformatics*, 17(1):182–189, 2001.

[MSD+00] E. W. Myers, G. G. Sutton, A. L. Delcher, I. M. Dew, D. P. Fasulo, M. J. Flanigan, S. A. Kravitz, C. M. Mobarry, K. H. J. Reinert, K. A. Remington, E. L. Anson, R. A. Bolanos, H.-H. Chou, C. M. Jordan, A. L. Halpern, S. Lonardi, E. M. Beasley, R. C. Brandon, L. Chen, P. J. Dunn, Z. Lai, Y. Liang, D. R. Nusskern, M. Zhan, Q. Zhang, X. Zheng, G. M. Rubin, M. D. Adams, and J. C. Venter. A Whole-Genome Assembly of Drosophila. *Science*, 287(5461):2196–2204, 2000.

[MSZT99] D. H. Mathews, J. Sabina, M. Zuker, and D. H. Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J. Mol. Biol.*, 288(5):911–940, 1999.

[MU49] N. Metropolis and S. Ulam. The Monte Carlo Method. *J. Americ. Stat. Assoc.*, 44:335–341, 1949.

[MV00] T. Müller and M. Vingron. Modeling Amino Acid Replacement. *J. Comp. Biol.*, 7(6):761–776, 2000.

[MW96] X. Meng and W. H. Wong. Simulating Ratios of Normalization Constants via a Simple Identity: A Theoretical Exploration. *Statistica Sinica*, 6:831–860, 1996.

[NCB] National Center for Biotechnology Information, http://www.ncbi.nlm.nih.gov/.

[New08] L. A. Newberg. Significance of Gapped Sequence Alignments. *J. Comp. Biol.*, 15(9):1187–1194, 2008.

[NHH00] P. C. Ng, J. G. Henikoff, and S. Henikoff. PHAT: a transmembrane-specific substitution matrix. *Bioinformatics*, 16(9):760–766, 2000.

[NJ80] R. Nussinov and A. B. Jacobson. Fast algorithm for predicting the secondary structure of single-stranded RNA. *Proc Natl Acad Sci USA*, 77(11):6309–6313, 1980.

[NK92] K. Nakai and M. Kanehisa. A knowledge base for predicting protein localization sites in eukaryotic cells. *Genomics*, 14(4):897–911, 1992.

[Nol91] H. F. Noller. Ribosomal RNA and Translation. *Ann. Rev. Biochem.*, 60:191–227, 1991.

[NPGK78] R. Nussinov, G. Pieczenik, J. R. Griggs, and D. J. Kleitman. Algorithms for Loop Matchings. *J. Appl. Math.*, 35(1):68–82, 1978.

[NW70] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, 48:443–453, 1970.

[PA97] B. Persson and P. Argos. Prediction of Membrane Protein Topology Utilizing Multiple Sequence Alignments. *J. Protein Chem.*, 16(5):453–457, 1997.

[Pal03] M. Palassini. Ground-state energy fluctuations in the Sherrington-Kirkpatrick model. *cond-mat/0307713*, 2003.

[PDB] RCSB Protein Data Bank, http://www.rcsb.org.

[PFTV92] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Cambridge (UK) and New York, 2nd edition, 1992.

[PJ01] M. Pagni and C. V. Jongeneel. Making sense of score statistics for sequence alignments. *Brief Bioinform*, 2(1):51–67, 2001.

[PPRT00] A. Pagnani, G. Parisi, and F. Ricci-Tersenghi. Glassy Transition in a Disordered Model for the RNA Secondary Structure. *Phys. Rev. Lett.*, 84:2026–2029, 2000.

[PQK+07] E. Pruesse, C. Quast, K. Knittel, B. M. Fuchs, W. Ludwig, J. Peplies, and F. O. Glockner. SILVA: a comprehensive online resource for quality checked and aligned ribosomal RNA sequence data compatible with ARB. *Nucl. Acids Res.*, 35(21):7188–7196, 2007.

[Rab89] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[RDM98] A. Krogh R. Durbin, S. Eddy and G. Mitchison. *Biological Sequence Analysis*. Cambridge University Press, 1998.

[RFC96] B. Rost, P. Fariselli, and R. Casadio. Topology prediction for helical transmembrane proteins at 86% accuracy. *Protein Sci*, 5(8):1704–1718, 1996.

[RO99] M. Lässig R. Olsen, T. Hwa. Optimizing smith-waterman alignments. In *Pacific Symposium on Biocomputing 4*, 1999.

[RP02] J.T. Reese and W.R. Pearson. Empirical determination of effective gap penalties for sequence comparison. *Bioinformatics*, 18(11):1500–1507, 2002.

[RR91] A. B. Robinson and L. R. Robinson. Distribution of glutamine and asparagine residues and their near neighbours in peptides and proteins. *Proc Natl Acad Sci USA*, 88:8880–8884, 1991.

[RTV86] R. Rammal, G. Toulouse, and M. A. Virasoro. Ultrametricity for physicists. *Rev. Mod. Phys.*, 58(3):765–, 1986.

[SA94] D. Stauffer and A. Aharony. *Introduction To Percolation Theory*. CRC, 1994.

[San85] D. Sankoff. Simultaneous Solution of the RNA Folding, Alignment and Protosequence Problems. *SIAM J. Appl. Math.*, 45(5):810–825, 1985.

[SAY05] M. E. Sardiu, G. Alves, and Y. Yu. Score statistics of global sequence alignment from the energy distribution of a modified directed polymer and directed percolation problem. *Phys. Rev .E.*, 72:061917, 2005.

[SB98] I. N. Shindyalov and P. E. Bourne. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Eng.*, 11(9):739–747, 1998.

[Sco04] M. L. J. Scott. Critical Values for the Test of Flatness of a Histogram Using the Bhattacharyya Measure. Technical Report 2004-010, TINA, 2004.

[SD78] R.M. Schwartz and M.O. Dayhoff. Matrices for detecting distant relationships. In M.O. Dayhoff, editor, *Atlas of Protein Sequence and Structure*, volume 5,Suppl.3, pages 353–358. National Biomedical Research Foundation, Washington,D.C., 1978.

[SD99] W. Seffens and D. Digby. mRNAs have greater negative folding free energies than shuffled or codon choice randomized sequences. *Nucleic Acids Res.*, 27(7):1578–1584, 1999.

[SK94] L. Saul and M. Kardar. The 2D+/-J Ising spin glass: exact partition functions in polynomial time. *Nucl. Phys. B*, 432(3):641–667, 1994.

[SKSH86] E. C. Strauss, J. A. Kobori, G. Siu, and L. E. Hood. Specific-primer-directed DNA sequencing. *Anal. Biochem.*, 154(1):353–360, 1986.

[SNC77] F. Sanger, S. Nicklen, and A. R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proc Natl Acad Sci USA*, 74(12):5463–5467, 1977.

[SS91] M. A. S. Saqi and M. J. E. Sternberg. A simple method to generate non-trivial alternate alignments of protein sequences. *J. Mol. Biol.*, 219(4):727–732, 1991.

[SvHK98] E. L.L. Sonnhammer, G. von Heijne, and A. Krogh. A hidden Markov model for predicting transmembrane helices in protein sequences. In J. Glasgow et al., editor, *Proc. Sixth Int. Conf. on Intelligent Systems for Molecular Biology*, pages 175–182. AAAI Press, 1998.

[SW81] T. F. Smith and M. S. Waterman. Identification of Common Molecular Subsequences. *J. Mol. Biol.*, 147:195–197, 1981.

[TB99] I. Tinoco,Jr. and C. Bustamante. How RNA folds. *J. Mol. Biol.*, 293(2):271–281, 1999.

[THT04] S. Trebst, D. A. Huse, and M. Troyer. Optimizing the ensemble for equilibration in broad-histogram Monte Carlo simulations. *Phys. Rev. E*, 70(4):046701, 2004.

[TS98] G. E. Tusnady and I. Simon. Principles governing amino acid composition of integral membrane proteins: application to topology prediction. *J. Mol. Biol.*, 283(2):489–506, 1998.

[TV77] G. M. Torrie and J. P. Valleau. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *J. Comp. Phys.*, 23(2):187–199, 1977.

[TW96] Craig A. Tracy and Harold Widom. On orthogonal and symplectic matrix ensembles. *Commun. Math. Phys.*, 177(3):727–754, 1996.

[Uni] Universal Protein Resource (UniProt), http://www.uniprot.org/.

[VAS+98] J. C. Venter, M. D. Adams, G. G. Sutton, A. R. Kerlavage, H. O. Smith, and M. Hunkapiller. GENOMICS: Shotgun Sequencing of the Human Genome. *Science*, 280(5369):1540–1542, 1998.

[VEA95] G. Vogt, T. Etzold, and P. Argos. An Assessment of Amino Acid Exchange Matrices in Aligning Protein Sequences: The Twilight Zone Revisited. *J. Mol. Biol.*, 249(4):816–831, 1995.

[vH92] G. von Heijne. Membrane protein structure prediction : Hydrophobicity analysis and the positive-inside rule. *J. Mol. Biol.*, 225(2):487–494, 1992.

[Vil77] J. Villain. Spin glass with non-random interactions. *J. Phys. C*, 10:1717–1734, 1977.

[VM04] P. Virnau and M. Müller. Calculation of free energy through successive umbrella sampling. *J. Chem. Phys.*, 120(23):10925–10930, 2004.

[Wan99a] J. S. Wang. Is the broad histogram random walk dynamics correct? *Eur. Phys. J. B*, 8(2):287–291, March 1999.

[Wan99b] J. S. Wang. Transition matrix Monte Carlo method. *Comput. Phys. Commun.*, 121-122:22–25, 1999.

[Wat94] M. S. Waterman. Parametric and ensemble sequence alignment algorithms. *Bull. Math. Biol.*, 56(4):743–767, 1994.

[WBH07] S. Wolfsheimer, B. Burghardt, and A. K. Hartmann. Local sequence alignments statistics: deviations from Gumbel statistics in the rare-event tail. *Algor. Mol. Biol.*, 2(1):9, 2007.

[WEL92] M.S. Waterman, M. Eggert, and E. Lander. Parametric sequence comparisons. *Proc Natl Acad Sci USA*, 89:6090–6093, 1992.

[WGA87] M. S. Waterman, L. Gordon, and R. Arratia. Phase transitions in sequence matches and nucleic acid structure. *Proc Natl Acad Sci USA*, 84(5):1239?–1243, 1987.

[WHRH] S. Wolfsheimer, I. Herms, S. Rahmann, and A. K. Hartmann. Accurate Statistics for Local Sequence Alignment with Position-Dependent Scoring by Rare-Event Sampling. submitted to BMC bioinformatics.

[Wil98] W. H. Wilbur. Accurate Monte Carlo Estimation of Very Small P-Values In Markov Chains. *Comp. Stat.*, 13:153–168, 1998.

[WK99] C. Workman and A. Krogh. No evidence that mRNAs have lower folding free energies than random sequences with the same dinucleotide distribution. *Nucleic Acids Res.*, 27(24):4816–4822, 1999.

[WL00] J. S. Wang and L. W. Lee. Monte Carlo algorithms based on the number of potential moves. *Comput. Phys. Commun.*, 127(1):131–136, 2000.

[WL01] F. G. Wang and D. P. Landau. Efficient, multiple-range random walk algorithm to calculate the density of states. *Phys. Rev. Lett.*, 86:2050, 2001.

[WS88] J. S. Wang and R. H. Swendsen. Low-temperature properties of the $\pm J$ Ising spin glass in two dimensions. *Phys. Rev. B*, 38:4840–4844, 1988.

[WTK+94] A. E. Walter, D. H. Turner, J. Kim, M. H. Lyttle, P. Mller, D. H. Mathews, and M. Zuker. Coaxial stacking of helixes enhances binding of oligoribonucleotides and improves predictions of RNA folding. *Proc Natl Acad Sci USA*, 27(20):9218–9222, 1994.

[WTS99] J. S. Wang, T. K. Tay, and R. H. Swendsen. Transition Matrix Monte Carlo Reweighting and Dynamics. *Phys. Rev. Lett.*, 82(3):476–479, 1999.

[XSB+98] T. Xia, J. SantaLucia, 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):14719–14735, 1998.

[YA05] Y.-K. Yu and S. F. Altschul. The construction of amino acid substitution matrices for the comparison of proteins with non-standard compositions. *Bioinformatics*, 21(7):902–911, 2005.

[You98] A. P. Young, editor. *Spin glasses and random fields*. World Scientific, Singapore, 1998.

[Yu04] Y.-K. Yu. Replica model for an unusual directed polymer in $1 + 1$ dimensions and prediction of the extremal parameter of gapped sequence alignment statistics. *Phys. Rev. E*, 69(6):061904, 2004.

[YWA03] Y.-K. Yu, J. C. Wootton, and S. F. Altschul. The compositional adjustment of amino acid substitution matrices. *Proc Natl Acad Sci USA*, 100(26):15688–15693, 2003.

[ZBG01] R. Zhou, B. J. Berne, and R. Germain. The free energy landscape for β hairpin folding in explicit water. *Proc Natl Acad Sci USA*, 98:14931–14936, 2001.

[ZM95] M. Q. Zhang and T. G. Marr. Alignment of Molecular Sequences Seen as Random Path Analysis. *J. Theor.Biol.*, 174:119–129, 1995.

[ZS81] M. Zuker and P. Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Res.*, 9(1):133–148, 1981.

[ZS84] M. Zuker and D. Sankoff. RNA secondary structures and their prediction. *Bull. of Math. Biol.*, 46(4):591–621, 1984.

[Zuk89] M. Zuker. On finding all suboptimal foldings of an RNA molecule. *Science*, 244:48, 1989.

[Zuk03] M. Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.*, 31(13):3406–3415, 2003.

Lebenslauf und Veröffentlichungen

Persönliche Daten

Name	Stefan Wolfsheimer
Geburtsdatum	20.9.1977
Geburtsort	Frankfurt a. Main
Staatsangehörigkeit	deutsch
Familienstand	ledig
Anschrift	Artillerieweg 42a 26129 Oldenburg Germany
E-Mail	wolfsh@theorie.physik.uni-oldenburg.de
Internet	http://www.physik.uni-oldenburg.de/compphys/en/28271.html

Schulbildung

1984-1990	Mittelpunktschule Weil-Ems (Weilrod)
1990-1997	Gymnasium, Pestalozzischule (Idstein) Abitur Juni 1997

Hochschule

1997-2005	Studium der Physik an der Johannes Gutenberg-Universität Mainz
April 2005	Hochschulabschluss als Diplom-Physiker Diplomarbeit im Gebiet Theorie kondensierter Materie unter Betreuung von Prof. Dr. Kurt Binder und Dr. Tanja Schilling.

August 2005 - Juni 2007	Promotionsstudium an der Georg-August Universität Göttingen in der Nachwuchsgruppe “Complex Ground States in Disordered Systems” unter Betreuung von PD Dr. Alexander Hartmann
Juni 2007 - jetzt	Fortsetzung des Promotionsstudiums an der Carl von Ossietzky Universität Oldenburg in der Arbeitsgruppe “Computerorientierte Theoretische Physik” unter Betreuung von Prof. Dr. Alexander Hartmann.

Berufserfahrung

1997-2004	Nebenberufliche Tätigkeit als Web- und Datenbankprogrammierer
2004-2005	Studentische Hilfskraft in Forschung und Lehre an der Universität Mainz
2005-2007	Wissenschaftlicher Mitarbeiter der Universität Göttingen
2007-jetzt	Wissenschaftlicher Mitarbeiter der Universität Oldenburg

Veröffentlichungen

Veröffentlichungen, auf die diese Dissertation basiert sind mit $*$ markiert.

- *Isotropic-nematic interfacial tension of hard and soft rods: application of advanced grand canonical biased sampling techniques,*
  R. L. C. Vink, S. Wolfsheimer, T. Schilling
  J. Chem. Phys. **123**, 074901 (2005)

- *The isotropic-nematic interface in suspensions of hard rods: Mean-field properties and capillary waves.*,
  S. Wolfsheimer, C. Tanase, K. Shundyak, R. van Roij, T. Schilling
  Phys. Rev. E 73, 061703 (2006)

- *Monte Carlo study of the isotropic-nematic interface in suspensions of spherocylinders*
  T. Schilling, R.L.C. Vink and S. Wolfsheimer
  in Computer Simulation Studies in Condensed Matter Physics,
  Eds. D.P.Landau, S.P. Lewis, and H.B. Schuettler, (Springer,Berlin, 2005).

- $*$ *Local sequence alignment: Deviations from Gumbel statistics in the rare-event-tail.*
  S. Wolfsheimer, B. Burghardt and A.K. Hartmann
  Algorithms for Molecular Biology 2007, 2:9

- $*$ *RNA secondary structures: complex statics and glassy dynamics*
  S. Wolfsheimer, B. Burghardt, A. Mann and A. K. Hartmann
  J. Stat. Mech. (2008) P03005

Veröffentlichungen unter Begutachtung

- $*$ *Accurate Statistics for Local Sequence Alignment with Position-Dependent Scoring by Rare-Event Sampling*
  S. Wolfsheimer, I. Herms, A.K. Hartmann and S. Rahmann,
  submitted to BMC Bioinformatics

- $*$ *Minimum-Free-Energy Distribution of RNA Secondary Structures: Entropic and Thermodynamic Properties of Large Deviations*
  S. Wolfsheimer and A.K. Hartmann,
  submitted to Phys. Rev. E

Stellungnahme zur Selbständigkeit

Hiermit versichere ich, dass ich diese Arbeit selbstständig verfasst, keine anderen als die angegebenen Hilfsmittel und Quellen benutzt und die den benutzten Werken wrtlich oder inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Ferner versichere ich, dass die vorliegende Dissertation weder in ihrer Gesamtheit noch in Teilen einer anderen wissenschaftlichen Hochschule zur Begutachtung in einem Promotionsverfahren vorliegt oder vorgelegen hat.

Oldenburg, 17.12.08

Stefan Wolfsheimer

Danksagung

An dieser Stelle möchte ich mich bei allen Personen und Institutionen bedanken, die zum Gelingen der Arbeit beigetragen haben. Prof. Dr. Alexander Hartmann danke ich für die freundliche Aufnahme in seine Arbeitsgruppe und die exzellente Betreuung. Ich hatte die Gelegenheit, viele fruchtbare Diskussionen mit ihm zu führen. Dadurch konnte ich viel über wissenschaftliche Arbeit lernen. Prof. Dr. Andreas Engel danke ich für die Übernahme des Zweitgutachtens.

Bei Dr. Bernd Burghardt bedanke ich mich für die gute Zusammenarbeit und Unterstützung. Bei Prof. Dr. Sven Rahmann und Inke Herms bedanke ich mich für die gute Zusammenarbeit und freundliche Einladung nach Bielefeld. Durch den Forschungsaufenthalt hat sich mein Blick auf Hidden Markov Modelle und ihre Anwendungen auf die Bioinformatik erweitert.

Alexander Mann und Oliver Melchert haben mir netterweise ihre Programme zum hierarchischen Clustern bzw. zum Finite-Size Scaling überlassen, wofür ich mich an dieser Stelle bedanke. Ebenso bedanke ich mich bei Prof. Dr. Lawrence Saul, der mir sein Programm zur Bestimmung der Zustandsdichte des $\pm J$ Spin Glasses zur Verfügung stellte.

Bei der Gesellschaft für wissenschaftliches Rechnen mbH, dem Rechenzentrum und der Arbeitsgruppe Computerchemie der Universität Oldenburg, dem Center for Biotechnology (CeBiTec) der Universität Bielefeld bedanke ich mich für die Bereitstellung der notwendigen Rechenzeit. Den Administratoren des Instituts für Theoretische Physik in Göttingen sowie der Arbeitsgruppen der Theoretischen Physik in Oldenburg, Jürgen Holm und Stefan Krautwald bedanke ich mich für die geduldige Unterstützung in Computerfragen.

Mein herzlicher Dank gilt weiterhin allen Mitgliedern der Arbeitsgruppen “Complex Ground States in Disordered Systems” in Göttingen, “Computerorientierte Theoretische Physik” in Oldenburg sowie Kollegen anderer Arbeitsgruppen, mit denen ich das Büro teilte, für die angenehme Atmosphäre, zahlreiche Diskussionen und private Gespräche. Während meiner Promotionszeit waren das Björn Ahrens, Luis Apolo, Dr. Wolfgang Barthel, Dr. Bernd Burghardt, Andrea Fiege, Thomas Heiser, Magnus Jungsbluth, Till Kranz, Alexander Mann, Kristian Marx, Oliver Melchert, Christian Schöne, Bruno Sciolla, Taha Yasseri, Dr. Emmanuel Yewande und Martin Zumsande.

Der Volkswagenstiftung danke ich für die finanzielle Unterstützung.