TU Delft

Bachelor End Project

The Random Energy Model for Compact Heteropolymer Folding

Author
Stan Hennekes

Supervisors
Dr. W.M. Ruszel &
Dr. J.M. Thijssen
Other Committee Members
Dr. J.L.A. Dubbeldam &
Dr. A. Akhmerov

August 14, 2017

Abstract

In this thesis, the Random Energy Model (REM) is applied to the highly complex problem of random heteropolymer folding. The relevance of the REM lies in the fact that it is an analytically solvable model, which makes it possible to learn more about the (thermodynamical) behaviour of folding proteins.
Firstly, a rigorous proof of the existence of a critical point in the REM with non-zero mean is presented. The mathematical properties of the REM with non-zero mean are used to derive the thermodynamical properties of this special case of the REM, such as the free energy.
In the second part of this thesis, applications of the model to folding polymers are investigated and a simple simulation of protein folding using the REM is suggested. It turns out that this simulation is barely useful, so a more realistic version of the REM for polymer folding is suggested.

Acknowledgements

I would like to thank Wioletta Ruszel, who helped me a lot during this project, and Jos Thijssen, who gave a lot on feedback in writing this thesis.

Contents

Chapter 1

Introduction

The folding of heteropolymers is one of the most crucial biological processes within the human body. For instance, our DNA consists of long chains of polymers that are folded in extremely dense configurations, and are partially unfolded for DNA replication. Interactions between proteins - a kind of heteropolymers - guide many processes in the cell, such as proliferation (cell growth) and differentiation (changing cell type)[1]. For example, many allergies are caused by a misfolding of certain polymers, as the human immune system produces antibodies only for certain protein structures[2]. The folding of heteropolymers is a complex process that is not fully understood and existing models require much computing time. In this thesis, an analytically solvable model is applied to the problem of protein folding. This model has its origin in the field of spin glasses and is called the Random Energy Model.

In this introduction chapter, we begin with a brief summary of the theory of spin glasses. We try to explain the origin and relevance of the Random Energy Model and describe the system we want to apply the theory to.

1.1 Spin glasses and the Random Energy Model

In 1980, the Random Energy Model (REM) was created by Bernard Derrida[3] as a simple toy model to try to understand the behaviour of disordered systems. This model later became widely accepted in the field of spin glasses.[4]
The field of spin glass theory has its origin in the attempt to describe the behaviour of glass and systems with a glasslike build-up. The difficulty in describing such a system lies in the way glasslike materials are built up. At first sight, these kind of materials seem to be ordered in patterns on the microscopic scale, but they turn out to be fairly unstructured. This will be further explained in the next chapter. The lack of symmetry of glass leads to many mathematical complications in trying to describe a glasslike system deterministically. Therefore, the REM describes the system stochastically.

1.1.1 Description and relevance of the original model

In the original paper of Derrida, a very simple form of the Random Energy Model was formulated. The model consisted of a set of N spins that can either be spin up or spin down. The energy of each of the spins is described in terms of the Hamiltonian. The Hamiltonian[5] of each spin is not correlated with the state of the others, but is driven with a centred Gaussian distribution. We will go into the details of this model later, but for now it is enough to observe the relative simplicity of the model. Despite this simplicity, Derrida was able to show the existence of a phase transition - a sudden change in behaviour - in this model. That is, there exists a certain critical point (for instance a critical temperature) at which some properties (for example the free energy) of the system change drastically.
It can be claimed[6] that the REM is the simplest statistical physics model of a disordered system which exhibits a phase transition. The relevance of the REM lies in the existence of this phase transition, which is useful in many different contexts. Most obviously, the REM is used as a toy model to study mathematical and physical properties of disordered systems, and is in some cases good enough to represent an actual system[7]. However, there are applications besides of that as well. For instance, the REM was used to model number partitioning in the field of stochastic optimization [8]. That is, given n numbers $X_1, ..., X_n$ drawn i.i.d. from some distribution, one is asked to find the partition into two subsets such that the sum of the numbers in one subset is as close as possible to the sum of the numbers in the other set.

1.2 Application

In this report, we will adjust the REM in such a way that it is applicable to describe the behaviour of folding compact heteropolymers[9], like proteins. As mentioned before, this is a difficult process that is very relevant in chemistry and nanobiology, for example to study the structure of DNA. Due to the complexity of these polymers, it is hard to quantify the way heteropolymers structure and the REM can help to gain more insight in this problem, as it can be studied analytically. However, the regular REM is hard to implement in the context of folding polymers, as it is centred around states with zero energy[10]. Therefore, we will have to study a REM with a non-zero mean in the rest of this report.

Chapter 2

The original REM

In this chapter, we take a closer look to the original Random Energy Model of Derrida. The model will be quantified, different properties will be studied and a few useful results that are already known for this model will be stated. Furthermore, we will introduce some relevant lemmas and definitions.

2.1 Quantification of the original model

The REM stated in the first chapter will now be fully characterized as a standard statistical mechanics model, which is usually defined by a set of configurations and an energy function defined on this space. In our model, the set of the possible spins is called the state space $S_N = \{-1, +1\}^N$. This set consist of N different elements on a lattice which can be either spin up or spin down. The number of possible configurations now is 2^N and the space looks similar to Figure 2.1.
Note that the spacing between the spin elements is periodic. This is different to the build-up of glasslike materials on the atomic scale, as shown in Figure 2.2. It is clear from this Figure that the structure of glass is only seemingly symmetric: there is some kind of pattern, but we can not recognize any symmetries. These kind of systems are sometimes called quasi-symmetric. To understand why the REM can describe such a quasi-symmetric system, a slight change of view should be made. Instead of considering the atoms to be located randomly according to each other (which is the case in a glasslike material), the energies of the different components are taken to be random and the locations to be periodic in the REM. This approach essentially leads to the same properties, but has mathematical advantages.

The energy in our system will be fully determined by the Hamiltonian, which has the following form:

$$H_N(\sigma) = \sum_{i=1}^{N} E_i \sigma_i \tag{2.1}$$

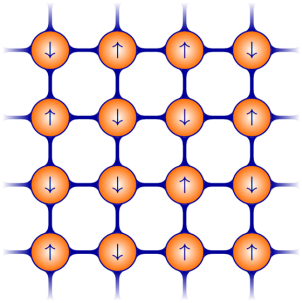

Figure 2.1: State space with spins up and down

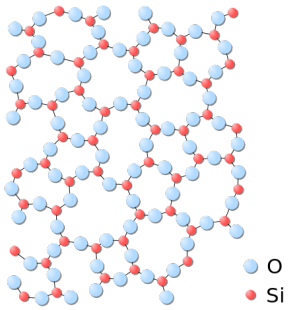

Figure 2.2: Glasslike structure

where E_i are i.i.d. standard normal random variables and $\sigma_i \in S_N$. So, every spot on the lattice contributes a random energy E_i multiplied by a $\sigma_i \in \{+1, -1\}$ to the Hamiltonian. Note that for large N the sum of Gaussian variables is Gaussian again, so the Hamiltonian in the REM is a Gaussian.

2.2 Free energy, the partition function

To study the properties of statistical physics models, a few concepts should be known, first of which is the partition function Z. We can define this function as[11]:

$$Z_N = \sum_{\sigma \in S_N} e^{-\beta H_N(\sigma)}$$

where $\beta = \frac{1}{k_B T}$ is a parameter ≥ 0 representing the inverse temperature (up to the Boltzmann constant). The partition function can be used to measure the (Gibbs) probability of the system to be in a certain state[12]:

$$\mathbb{P}(\sigma) = \frac{1}{Z_N} e^{-\beta H_N(\sigma)} \tag{2.2}$$

Note that the probability in (2.2) is a stochastic variable itself. One could say that the REM contains two different layers of randomness.
Furthermore, the partition function can be used to compute the free energy F_N:

$$F_N = -\frac{1}{\beta} \ln Z_N$$

This free energy is the total energy needed to create the system at a temperature T, minus the heat you can get 'for free' from an environment at T. It is convenient to look at a quantity f that is very much alike the free energy (we will call it the free energy density[12]):

$$f = \frac{1}{N} \ln Z_N$$

As claimed before, this quantity exhibits a phase transition (for example: see Bovier[7]). The computation gives the following important result. Note the dramatic change in free energy at the critical point β_c.

Theorem 1. *In the REM as described above* [1]

$$Lim_{N\to\infty}\mathbb{E}(f) = \begin{cases} \frac{\beta^2}{2}, & \text{if } \beta \leq \beta_c \\ \frac{\beta^2}{2} + (\beta - \beta_c)\beta_c, & \text{if } \beta \geq \beta_c \end{cases} \tag{2.3}$$

where $\beta_c = \sqrt{2\ln 2}$.

2.3 Entropy, density of states

Another important concept from statistical physics is the density of states Ω, which represents the number of ways arranging things in the system. If we call the total number of states in the system g'_N, then the density of states is just that number multiplied by the probability distribution function:

$$\Omega(E) = g'_N p(E)$$

From this we can calculate the entropy S of the system:

$$S(E) = k_B \ln \Omega(E) \tag{2.4}$$

This quantity has a few interesting physical properties. Entropy and temperature are linked via $T^{-1} = dS/dE$. Temperature is always positive, so $dS/dE > 0$. Furthermore, the entropy can never be negative.

2.4 Useful theorems and definitions

In the next chapter, we will go deeper into the mathematical structure behind the phase transition of the REM. To do so, we need some definitions and theorems from statistics. The first two are standard.

Markov's inequality: For any non-negative random variable X: $\mathbb{P}(X \geq t) \leq \frac{\mathbb{E}(X)}{t}$ for $t > 0$.

Chebyshev's inequality: For any random variable Y with $\mathbb{E}(Y) < \infty$:
$\mathbb{P}(|Y| \geq t) \leq \frac{1}{t^2}\mathbb{E}(Y^2)$.

The most important lemmas for the next chapter are those of Borel-Cantelli and Varadhan. The lemma of Borel-Cantelli is useful for proving a statement to be almost sure. Almost sure convergence is a strong convergence of a sequence of random variables and is widely used in

[1] That is: using the Hamiltonian with i.i.d. standard normal energies as described in equation (2.1) and the state space S_N.

probability theory. We say that the sequence $(X_n, n \geq 1)$ converges almost surely to X if $\mathbb{P}(\{\omega \in \Omega : Lim_{n\to\infty} X_n(\omega) = X(\omega)\}) = 1$. It is more common to state the Borel-Cantelli lemma reversed, so for an event to happen almost surely not (probability 0)[13].

Borel-Cantelli lemma: Let $E_1, E_2, ...$ be a sequence of events in some probability space. If $\sum_{n=1}^{\infty} \mathbb{P}(E_n) < \infty$, then $\mathbb{P}(\limsup_{n\to\infty} E_n) = \mathbb{P}(\bigcap_{n=1}^{\infty} \bigcup_{k\geq n}^{\infty} E_k) = 0$

Varadhan's lemma requires a function with large deviation properties, so we have to introduce these first. Large deviations have to do with the asymptotic behaviour of remote tails of sequences of probability distributions and are closely linked to disordered systems. Therefore we will use them later in this thesis. To get a feeling about what a *large* deviation is, we formulate the central limit theorem in the following way[14]:

Central Limit Theorem Let $X_1, X_2, ...$ be i.i.d. random variables. Let $\mathbb{E}(X_1) = \mu$, Var $(X_1) = \sigma^2$ and $S_n = X_1 + ... + X_n$. Then:

$$\frac{1}{\sigma\sqrt{n}}(S_n - \mu n) \xrightarrow[n\to\infty]{} Z$$

Where Z is a standard normal random variable.

We see that the CLT quantifies the probability that S_n and μn differ an amount of order $\sqrt{n}$. When we speak about large deviations, we mean differences or order n. This means the deviations we talk about are indeed *large*.

We now state a few definitions on large deviations:

Definition 1. *A function I is called lower semi-continuous[15] if $\forall l \geq 0$ the sets $\{x : I(x) \leq l\}$ are closed for x in the domain of I.*

Definition 2. *A function $I : X \to [0, \infty]$ is called a rate function [16] if it is lower semi-continuous*

Definition 3. *A rate function $I : X \to [0, \infty]$ is called a good rate function[16] if $\{x : I(x) \leq l\}$ is compact in X $\forall l < \infty$.*

Definition 4. *Let I be a rate function on X, (γ_n) a sequence with $\gamma_n \to \infty$ on X and (μ_n) a sequence of probability measures on X. We say that (μ_n) satisfies a large deviation principle[16] (LDP) with rate I and speed γ_n, if:*

1. *For all open sets $O \subset X$ we have $\liminf_{(n)} \frac{1}{\gamma_n} \ln \mu_n(O) \geq -\inf_{x\in O} I(x)$.*
2. *For all closed sets $C \subset X$ we have $\limsup_{(n)} \frac{1}{\gamma_n} \ln \mu_n(C) \geq -\inf_{x\in C} I(x)$*

Varadhan's lemma states a useful property of a function that satisfies a LPD. We will use this lemma in the next chapter.

Varadhan's lemma[17]: Suppose P_N satisfies a LDP with rate function I. If $F : \chi \to \mathbb{R}$ is continuous and bounded above, then $\lim_{N\to\infty} \frac{1}{N} \ln \int_\chi e^{NF(x)} \, \mathrm{d}P_N(x) = sup_{(x\in\chi)}(F(x) - I(x))$.

Chapter 3

Adding a mean

In this chapter, a minor modification of the Random Energy Model is studied. Instead of a Gaussian distribution centred around zero, we will look at the properties of the REM with Gaussian distributed variables around a non-zero mean. This means that the random energies as described in the Hamiltonian in (2.1) are shifted. This leads to a new (and more general) version of the model described in the previous chapter, so we want to explore what effect adding a mean has on the free energy. Our goal is to find out whether or not the phase transition in the model mentioned before still remains. If so, the new model could be applicable in the context of heteropolymers.

3.1 Define model

The basic setup is equivalent to the normal REM as described in the introductory chapter. We still define a spin system with independently distributed random variables, but we will add a mean μ (and a less important variance term J^2). All X_σ's have the following probability density:

$$p_N(E) = \frac{1}{\sqrt{2\pi J^2 N}} e^{-(E-N\mu)^2/(2NJ^2)} \tag{3.1}$$

And we define for any given set of energies $\{E_i\}$ the distribution function F_N by:

$$F_N(x) = \frac{1}{2^N} \#\{i : E_i \leq xN\} \tag{3.2}$$

Where # counts the number of elements in the set. Note that F_N is random itself, so we have 'two layers of randomness'. We can write the partition function $Z_N(\beta) = \sum_{i=1}^{2^N} e^{-\beta E_i}$ as:

$$Z_N(\beta) = 2^N \int_{-\infty}^{\infty} e^{-N\beta x} \, \mathrm{d}F_N(x)$$

We can define the free energy density of the system as:

$$f(\beta) = -\frac{1}{\beta N} \ln Z_N(\beta) \tag{3.3}$$

3.2 Large deviation properties

We want to show that these distribution functions satisfy a large deviation property and that there exists a certain critical value x_c at which a sudden change in behaviour occurs. We describe the system for values smaller and larger than this x_c. The proof will follow a similar approach as that of T.C. Dorlas[18], but with a nonzero mean added.

3.2.1 $|x| > x_c$

Lemma 1. *For* $|x| > x_c = J\sqrt{2\ln 2} + \mu$*:* $\limsup_{N\to\infty} \frac{1}{N} \ln(1 - F_N(x)) = -\infty$

Proof. First, consider:

$$1 - F_N(x) = \frac{2^N}{2^N} - \frac{\sum_{i=1}^{2^N} \mathbb{1}_{\{E_i \leq xN\}}}{2^N} = \frac{1}{2^N} \sum_{i=1}^{2^N} \mathbb{1}_{\{E_i > xN\}}$$

Therefore:

$$\{\{E_i\} : F_N(x) = 1\} = \{\{E_i\} : \sum_{i=1}^{2^N} \mathbb{1}_{\{E_i > xN\}} = 0\}$$

But every $\mathbb{1}_{\{E_i > xN\}}$ gives either 0 or 1, so equivalently:

$$\{\{E_i\} : F_N(x) = 1\} = \{\{E_i\} : \sum_{i=1}^{2^N} \mathbb{1}_{\{E_i > xN\}} < 1\}$$

Introduce $A_N := \{\{E_i\} : \sum_{i=1}^{2^N} \mathbb{1}_{\{E_i > xN\}} \geq 1\}$. To be able to use the Borel-Cantelli lemma, we want to show that:

$$\sum_{N=1}^{\infty} \mathbb{P}(A_N) < \infty \tag{3.4}$$

Noticing that $\sum_{i=1}^{2^N} \mathbb{1}_{\{E_i > xN\}} \geq 0$, Markov's inequality implies:

$$\mathbb{P}(A_N) = \mathbb{P}\{\sum_{i=1}^{2^N} \mathbb{1}_{\{E_i > xN\}} \geq 1\} \leq \mathbb{E}\{\sum_{i=1}^{2^N} \mathbb{1}_{\{E_i > xN\}}\}$$

$$= 2^N \mathbb{P}\{E_i > xN\} = 2^N \int_{xN}^{\infty} p_N(E)\,\mathrm{d}E$$
$$= \frac{2^N}{\sqrt{2\pi J^2 N}} \int_{xN}^{\infty} e^{-(E-N\mu)^2/(2NJ^2)}\,\mathrm{d}E$$

We consider two cases, the first of which is $x > \mu$. We first change the boundaries of the integral.

$$\frac{2^N}{\sqrt{2\pi J^2 N}} \int_{xN}^{\infty} e^{-(E-N\mu)^2/(2NJ^2)}\,\mathrm{d}E = \frac{2^N}{\sqrt{2\pi J^2 N}} \int_{(x-\mu)N}^{\infty} e^{-E^2/(2NJ^2)}\,\mathrm{d}E \tag{3.5}$$

Within the boundaries of this integral we see that: $E \geq (x-\mu)N$, thus $\frac{E}{N(x-\mu)} \geq 1$ (where we used that $x > \mu$) and we obtain:

$$\frac{2^N}{\sqrt{2\pi J^2 N}} \int_{(x-\mu)N}^{\infty} e^{-E^2/(2NJ^2)}\,\mathrm{d}E \leq \frac{2^N}{\sqrt{2\pi J^2 N}} \int_{(x-\mu)N}^{\infty} \frac{E}{(x-\mu)N} e^{-E^2/(2NJ^2)}\,\mathrm{d}E$$
$$= \frac{2^N}{\sqrt{2\pi J^2 N}} \frac{J^2}{x-\mu} e^{-(x-\mu)^2 N/2J^2} = \frac{2^N}{\sqrt{2\pi N}} \frac{J}{x-\mu} e^{-(x-\mu)^2 N/2J^2}$$

In short, for $x > \mu$:

$$\frac{2^N}{\sqrt{2\pi J^2 N}} \int_{(x-\mu)N}^{\infty} e^{-E^2/(2NJ^2)}\,\mathrm{d}E \leq \frac{2^N}{\sqrt{2\pi N}} \frac{J}{x-\mu} e^{-(x-\mu)^2 N/2J^2} \tag{3.6}$$

We now consider the case where $x \leq \mu$. Equation (3.5) still holds, but now $x - \mu \leq 0$. We split the integral in two parts:

$$\frac{2^N}{\sqrt{2\pi J^2 N}} \int_{(x-\mu)N}^{\infty} e^{-E^2/(2NJ^2)}\,\mathrm{d}E = \frac{2^N}{\sqrt{2\pi J^2 N}} \left(\int_{(x-\mu)N}^{0} e^{-E^2/(2NJ^2)}\,\mathrm{d}E + \int_{0}^{\infty} e^{-E^2/(2NJ^2)}\,\mathrm{d}E\right)$$

The second integral is just a normal distribution over a half-infinite interval, so we can say that this is integral is no larger than $\frac{1}{2}$ (it even is $\frac{1}{2}$). Within the boundaries of the first integral we see that: $0 \geq E \geq (x-\mu)N$, thus $\frac{E}{N(x-\mu)} \geq 1$ and we obtain a similar upper bound as before:

$$\int_{(x-\mu)N}^{0} e^{-E^2/(2NJ^2)}\,\mathrm{d}E \leq \int_{(x-\mu)N}^{0} \frac{E}{N(x-\mu)} e^{-E^2/(2NJ^2)}\,\mathrm{d}E = \frac{J^2}{x-\mu}[e^{-(x-\mu)^2 N/2J^2} - 1]$$

This gives:

$$\int_{(x-\mu)N}^{\infty} e^{-E^2/(2NJ^2)}\,\mathrm{d}E \leq \left(\frac{J^2}{x-\mu} e^{-(x-\mu)^2 N/2J^2} - 1\right) + 1/2$$

And so:

$$\frac{2^N}{\sqrt{2\pi J^2 N}} \int_{(x-\mu)N}^{\infty} e^{-E^2/(2NJ^2)}\,\mathrm{d}E \leq \frac{2^N}{\sqrt{2\pi J^2 N}} \left[\left(\frac{J^2}{x-\mu} e^{-(x-\mu)^2 N/2J^2} - 1\right) + 1/2\right]$$

$$= \frac{2^N}{\sqrt{2\pi J^2 N}}[\frac{J^2}{x-\mu}e^{-(x-\mu)^2 N/2J^2} - 1/2]$$

Thus for $x < \mu$:

$$\frac{2^N}{\sqrt{2\pi J^2 N}}\int_{(x-\mu)N}^{\infty} e^{-E^2/(2NJ^2)}\,\mathrm{d}E \leq \frac{2^N}{\sqrt{2\pi J^2 N}}[\frac{J^2}{x-\mu}e^{-(x-\mu)^2 N/2J^2} - 1/2] \tag{3.7}$$

So in both cases it is true that (see equations (3.6) and (3.7)):

$$\mathbb{P}(A_N) \leq \frac{2^N}{\sqrt{2\pi J^2 N}}\int_{xN}^{\infty} e^{-(E-N\mu)^2/(2NJ^2)}\,\mathrm{d}E \leq \frac{2^N}{\sqrt{2\pi N}}\frac{J}{x-\mu}e^{-(x-\mu)^2 N/2J^2} \tag{3.8}$$

We assumed that $|x| > x_c = J\sqrt{2\ln 2} + \mu$, so $\frac{(x-\mu)^2}{2J^2} > \ln 2$. Furthermore, the following series converges to a value c:

$$\sum_{N=1}^{\infty} \frac{J}{\tilde{x}\sqrt{2\pi N}} e^{N(\ln 2 - \tilde{x}^2/(2J^2))} \to c \tag{3.9}$$

Noticing that $e^{N\ln 2} = 2^N$ we obtain by equation (3.9) (replacing $\tilde{x} = x - \mu$):

$$\sum_{N=1}^{\infty} \frac{\tilde{J}}{\tilde{x}\sqrt{2\pi N}} e^{N(\ln 2 - \tilde{x}^2/(2J^2))} = \sum_{N=1}^{\infty} 2^N \frac{2J^2}{(x-\mu)\sqrt{2\pi N}} e^{-N(x-\mu)^2/(2J^2)} \geq \sum_{N=1}^{\infty} \mathbb{P}(A_N) \tag{3.10}$$

This proves equation (3.4). By the Borel-Cantelli lemma:

$$\mathbb{P}[\bigcap_{k=1}^{\infty}\bigcup_{N=k}^{\infty} A_N] = 0$$

But then:

$$1 = \mathbb{P}[\{\bigcap_{k=1}^{\infty}\bigcup_{N=k}^{\infty} A_N\}^c] = \mathbb{P}[\{E_i\} \in \{\bigcap_{k=1}^{\infty}\bigcup_{N=k}^{\infty} A_N\}^c] = \mathbb{P}[\{E_i\} \in \bigcap_{k=1}^{\infty}\bigcup_{N=k}^{\infty} A_N^c]$$

So $\{E_i\} \in \bigcap_{k=1}^{\infty}\bigcup_{N=k}^{\infty} A_N^c$ almost surely for all i. This means that:

$$\forall i \exists k \in \mathbb{N} : \forall N \geq k : \{E_i\} \in A_N^c$$

But:

$$\{E_i\} \in A_N^c \implies \{E_i\} \in \{\{E_i\} : \frac{1}{2^N}\sum_{i=1}^{2^N} \mathbb{1}_{\{E_i > (x+\mu)N\}<1}\} = \{\{E_i\} : F_N(x) = 1\}$$

So almost surely for $|x| > x_c$: $\{E_i\} \in \{\{E_i\} : F_N(x) = 1\}$. This proves the lemma:

$$\limsup_{N\to\infty} \frac{1}{N}\ln(1 - F_N(x)) = \lim_{x\to 1}\ln(1-x) = -\infty\,, a.s.$$

□

3.2.2 $|x| < x_c$

Lemma 2. *For* $0 < |x| < x_c$*:* $\lim_{N\to\infty} \frac{1}{N} \ln(1 - F_N(x)) = -\frac{(x-\mu)^2}{2J^2}$

Proof. Let G_N be the scaled distribution function to the density p_N:

$$G_N(x) = \int_{-\infty}^{xN} p_N(E)\,\mathrm{d}E \tag{3.11}$$

Then, by Chebyshev's inequality, for $\epsilon \in (0, 1)$:

$$P[|G_N(x) - F_N(x)| \geq \epsilon(1 - G_N(x))] \leq \frac{1}{\epsilon^2(1 - G_N(x))^2} E[(G_N(x) - F_N(x))^2] \tag{3.12}$$

Working out the last term, by noticing that the function $G_N(x)$ is deterministic and using the linearity of the expectation:

$$E[(G_N(x) - F_N(x))^2] = E[G_N^2(x) - 2G_N(x)F_N(x) + F_N^2(x)]$$

$$= G_N^2(x) - 2E[F_N(x)]G_N(x) + E[F_N^2(x)]$$

Recall that $F_N(x) = \frac{1}{2^N} \sum_{i=1}^{2^N} \mathbb{1}_{\{E_i \leq xN\}}$ to obtain:

$$E[F_N(x)] = \frac{1}{2^N} 2^N \int_{-\infty}^{xN} p_N(E)\,\mathrm{d}E = G_N(x) \tag{3.13}$$

We are now left with:

$$G_N^2(x) - 2E[F_N(x)]G_N(x) + E[F_N^2(x)] = G_N^2(x) - 2G_N(x)G_N(x) + E[F_N^2(x)] \tag{3.14}$$

$$= -G_N^2(x) + (\frac{1}{2^N} \sum_{i=1}^{2^N} \mathbb{1}_{\{E_i \leq xN\}})^2$$

$$= \frac{1}{2^{2N}} \sum_{i=1}^{2^N} \sum_{j=1}^{2^N} (E[\mathbb{1}_{\{E_i \leq xN\}}]E[\mathbb{1}_{\{E_j \leq xN\}}]) - G_N^2(x)$$

If $i = j$, both terms in the expected value will be the same (either 0 or 1), so the power 2 can be left out. We can thus split the sum in the above equation and rewrite to:

$$\frac{1}{2^{2N}} [\sum_{i \neq j} (E[\mathbb{1}_{\{E_i \leq xN\}}]E[\mathbb{1}_{\{E_j \leq xN\}}]) + \sum_{i=1}^{2^N} E[\mathbb{1}_{\{E_i \leq xN\}}]] - G_N^2(x)$$

$$= \frac{1}{2^{2N}} [\sum_{i \neq j} G_N^2(x) + \sum_{i=1}^{2^N} G_N(x)] - G_N^2(x)$$

As the term in the sums does not depend on i or j whatsoever, we can say:

$$= \frac{1}{2^{2N}}[2^N(2^N-1)G_N^2(x) + 2^N G_N(x)] - G_N^2(x)$$

$$= 2^{-N}(2^N - 1 - 2^N)G_N^2(x) + 2^{-N}G_N(x) = 2^{-N}G_N(x)(1-G_N(x))$$

So altogether we can say that

$$E[(G_N(x) - F_N(x))^2] = 2^{-N}(2^N - 1 - 2^N)G_N^2(x) + 2^{-N}G_N(x) = 2^{-N}G_N(x)(1-G_N(x)) \tag{3.15}$$

Plugging in equation (3.15) into equation (3.12) gives:

$$P[|G_N(x) - F_N(x)| \geq \epsilon(1-G_N(x))] \leq \frac{G_N(x)}{\epsilon^2 2^N(1-G_N(x))}$$

To get an upper bound for the last term in the above equation, we use the following inequality[19]:

$$\int_a^\infty e^{-u^2/2}\,\mathrm{d}u > \frac{1}{a+a^{-1}}e^{-a^2/2} \tag{3.16}$$

To use this inequality we rewrite our expression for $G_N(x)$, using equations (3.1) and (3.11) and then change variables:

$$G_N(x) = 1 - \int_{xN}^\infty \frac{1}{\sqrt{2\pi J^2 N}} e^{-(E-N\mu)^2/(2NJ^2)}\,\mathrm{d}E$$

$$= 1 - \frac{1}{\sqrt{2\pi J^2 N}} \int_{\sqrt{N}\frac{x-\mu}{J}}^\infty e^{-u^2/2}\,\mathrm{d}u$$

So with the inequality (3.16) we obtain:

$$G_N(x) < 1 - \frac{1}{\sqrt{2\pi}} \frac{1}{\sqrt{N}\frac{x-\mu}{J} + \frac{J}{\sqrt{N}(x-\mu)}} e^{-\frac{N(x-\mu)^2}{2J^2}}$$

In a similar way:

$$1 - G_N(x) > \frac{1}{\sqrt{2\pi}} \frac{1}{\sqrt{N}\frac{x-\mu}{J} + \frac{J}{\sqrt{N}(x-\mu)}} e^{-\frac{N(x-\mu)^2}{2J^2}}$$

This leads to:

$$P[|G_N(x) - F_N(x)| \geq \epsilon(1-G_N(x))] \leq \frac{1}{\epsilon^2 2^N}$$

But then for N large enough, by the Borel-Cantelli lemma almost surely:

$$|G_N(x) - F_N(x)| < \epsilon(1-G_N(x))$$

And therefore:

$$(1-\epsilon)(1-G_N(x)) < 1 - F_N(x) < (1+\epsilon)(1-G_N(x))$$

This proves the lemma. □

3.2.3 $|x| = x_c$

Lemma 3. *For $x = x_c$:* $\limsup_{N\to\infty} \frac{1}{N} \ln(1 - F_N(x_c)) \leq -\frac{(x_c-\mu)^2}{2J^2}$

Proof. Use Chebyshev (notice that $F_N(x) < 1$):

$$P[1 - F_N(x_c) > N2^{-N}] = P[|1 - F_N(x_c)| > N2^{-N}] \leq \frac{2^{2N}}{N^2} E[(1 - F_N(x_c))^2]$$

$$= \frac{1}{N^2} \sum_{i=1}^{2^N} \sum_{j=1}^{2^N} E[\mathbb{1}_{\{E_i > x_c N\}} \mathbb{1}_{\{E_i > x_c N\}}]$$

$$= \frac{1}{N^2} [\sum_{i\neq j} (1 - G_N(x_c))^2 + \sum_{i=1}^{2^N} 1 - G_N(x_c)]$$

$$= \frac{1}{N^2} [2^N(2^N - 1)(1 - G_N(x_c))^2 + 2^N(1 - G_N(x_c))]$$

$$= \frac{2^N}{N^2}(1 - G_N(x_c))(1 + (2^N - 1))(1 - G_N(x_c)) \leq \frac{1}{N^2\sqrt{N}}$$

Where in the last inequality we used that:

$$1 - G_N(x) = \int_{x_c N}^{\infty} \frac{1}{\sqrt{2\pi J^2 N}} e^{-(E-N\mu)^2/(2NJ^2)} \, \mathrm{d}E$$

So by equation (3.11) with $x = x_c$:

$$1 - G_N(x) \leq \frac{1}{\sqrt{2\pi N}} \frac{J}{x_c - \mu} e^{-(x_c-\mu)^2 N/2J^2} = \frac{1}{2\sqrt{\pi N \ln 2}} e^{-N \ln 2} = \frac{2^{-N}}{2\sqrt{\pi N \ln 2}}$$

Now we see that $\frac{1}{N^2\sqrt{N}} \to 0$ for $N \to \infty$, so for N large enough we have with probability 1 that $1 - F_N(x_c) \leq N2^{-N}$. We finish the proof of this lemma by noticing that $-\frac{(x_c-\mu)^2}{2J^2} = -\ln 2$.

□

3.2.4 Open and closed sets

We now found a function that can possibly describe a large deviation property (LDP). This is the function $I(x)$:

$$I(x) = \begin{cases} \frac{(x-\mu)^2}{2J^2} := K(x), & \text{if } |x| \leq x_c \\ \infty, & \text{if } |x| > x_c \end{cases} \tag{3.17}$$

where $x_c = J\sqrt{2 \ln 2} + \mu$. This function $I : \mathbb{R} \to [0, \infty]$ is a good rate function. Firstly, we can easily see that this function is lower semi-continuous (the value x_c is $\frac{x_c^2}{2J^2}$ and not $+\infty$ and has only two discontinuities). Furthermore $\{x : I(x) \leq l\}$ has to be compact in $\mathbb{R}$ $\forall l < \infty$.

We see that the set $\{x : I(x) \leq l\}$ is bounded and closed as all $x < \infty$ lie in the interval $[-x_c, x_c]$.

There are two things left to prove in order to show that $I(x)$ is the rate of the LDP on the distribution functions F_N. First we show that

$$\forall O \subset \mathbb{R} \text{ open} : \liminf_{N\to\infty} \frac{1}{N} \ln(1 - F_N(O)) \geq \inf_{x\in O} I(x) \tag{3.18}$$

If $|x| > x_c$, we can use the first lemma. In this case, we have that $I(x) = +\infty$ and thus $-I(x) = -\infty$. So for all x in the open set $(x_c, \infty) \subset \mathbb{R}$ we can say that $I(x) = -\infty$ as well. We can use a similar approach for $(-\infty, -x_c)$. Choose an open set O in the subset (x_c, ∞). Then in O we have that $\liminf_{N\to\infty} \frac{1}{N} \ln(1-F_N(O)) \leq \limsup_{N\to\infty} \frac{1}{N} \ln(1-F_N(O)) = -\infty$. It now follows that $\liminf_{N\to\infty} \frac{1}{N} \ln(1 - F_N(O)) \geq -\inf_{x\in O} I(x)$, which shows that (3.18) is true.

The second thing to prove is:

$$\forall C \subset \mathbb{R} \text{ closed} : \limsup_{N\to\infty} \frac{1}{N} \ln(1 - F_N(C)) \leq -\inf_{x\in C} I(x) \tag{3.19}$$

If $|x| \leq x_c$, we can use lemma 2 and 3. In this case, we have that for $0 < |x| < x_c$: $\lim_{N\to\infty} \frac{1}{N} \ln(1 - F_N(x)) = -K(x)$. Then by definition $\limsup_{N\to\infty} \frac{1}{N} \ln(1 - F_N(x)) = -K(x)$. For $x = x_c$ we have that: $\limsup_{N\to\infty} \frac{1}{N} \ln(1 - F_N(x_c)) \leq -K(x)$, so for the total closed interval $[-x_c, x_c]$, we have that $\limsup_{N\to\infty} \frac{1}{N} \ln(1 - F_N(C)) \leq -K(x)$. Since $K(x)$ is strictly convex on the interval $[-x_c, x_c]$, there exists an infimum of K, namely at $x = \mu$. Notice that $K(x = \mu) = 0$ and that $\mu \leq x_c$. For every closed set C within $-x_c, x_c$ we can find such an infimum. It then follows that $\limsup_{N\to\infty} \frac{1}{N} \ln(1 - F_N(C)) \leq -\inf_{x\in C} K(x)$, which proves the statement in (3.19).

3.2.5 Main theorem

Using the lemmas obtained in the previous section we can set up the following theorem to describe a LDP of the system.

Theorem 2. *With probability 1, the random probability measures with distribution functions F_N satisfy a LDP with rate function I given by*

$$I(x) = \begin{cases} \frac{(x-\mu)^2}{2J^2}, & \text{if } |x| \leq x_c \\ \infty, & \text{if } |x| > x_c \end{cases} \tag{3.20}$$

where $x_c = J\sqrt{2\ln 2} + \mu$.

3.3 Finding the free energy

We now want to prove the next expression for the free energy in the system by making use of Varadhan's lemma:

Theorem 3. *The free energy density f of the REM as described in section 3.1 is:*

$$-\beta f(\beta) = \begin{cases} \ln 2 + -\beta^2 J^2 + \beta\mu - \frac{(\beta J^2 - \mu)^2}{2J^2}, & \text{if } \beta J^2 - \mu \leq x_c \\ \beta(J\sqrt{2\ln 2} + \mu), & \text{if } \beta J^2 - \mu > x_c \end{cases}$$

Proof. Take a function $F : \chi \to \mathbb{R}$ that maps $x \mapsto -\beta x$. We choose χ to be the interval $[-x_c, x_c] \subset \mathbb{R}$. It is obvious that the function F is continuous and bounded above on χ, as x_c is finite and β is a constant (with respect to N and x). From the above theorem, we take the rate function I on the specific interval. Then by Varadhan's lemma:

$$\lim_{N\to\infty} \frac{1}{N} \ln \int_\chi e^{-\beta N x} \, \mathrm{d}F_N(x) = \sup_{(x\in\chi)} [-\beta x - \frac{(x-\mu)^2}{2J^2}] \tag{3.21}$$

We see from equation (3.3) that the left hand side is equal to $-\beta f(\beta)$ up to a factor $\ln 2$. So:

$$-\beta f(\beta) - \ln 2 = \sup_{(x\in\chi)} [-\beta x - \frac{(x-\mu)^2}{2J^2}] \tag{3.22}$$

To find the value of this supremum, split the expression in two separate parts. If $\beta J^2 - \mu \leq x_c$ we use just that value to calculate the supremum. So the supremum is $(-\beta x - \frac{x^2}{2J^2})|_{x=\beta J^2-\mu} = -\beta^2 J^2 + \beta\mu - \frac{(\beta J^2-\mu)^2}{2J^2}$. If $\beta J^2 - \mu > x_c$, the supremum will arise if we choose $-\beta x$ as large as possible, so $x = -x_c$ and we get $(-\beta x - \frac{x^2}{2J^2})|_{x=-x_c} = \beta(J\sqrt{2\ln 2} + \mu) - \ln 2$. So if we plug these results into equations (3.21) and (3.22) we obtain Theorem 3. □

This result is remarkable in the existence of the critical point x_c. That is, there exists a point where the free energy of the system suddenly makes a transition between two regimes. This means there still exists a certain critical temperature T_c in the REM with non-zero mean Gaussian distributed energies.

3.4 Entropy of the system

In the first chapter we introduced the concept of entropy. Using the fact that each component in our system consists of only two different spin states, we can say that (using equations (3.1) and (2.4)):

$$S(E) = k_B \ln(2^N p(E)) = k_B[\ln 2^N + \ln(\frac{1}{\sqrt{2\pi J^2 N}}) - \frac{(E-N\mu)^2}{2NJ^2}]$$

$$= k_B[N\ln 2 - \frac{1}{2}\ln(2\pi J^2 N) - \frac{(E-N\mu)^2}{2NJ^2}]$$

If we assume that $\ln(2\pi J^2 N)$ is small enough compared to the other terms, we see that:

$$S(E) \approx k_B[N \ln 2 - \frac{(E - N\mu)^2}{2NJ^2}] \tag{3.23}$$

Which can be written as:

$$\frac{S(E)}{Nk_B} \approx -\frac{1}{2J^2}[\frac{E}{N}]^2 - \frac{\mu}{J^2}[\frac{E}{N}] + \ln 2 + \frac{\mu^2}{2J^2}$$

This is a parabolic expression and has the following form:

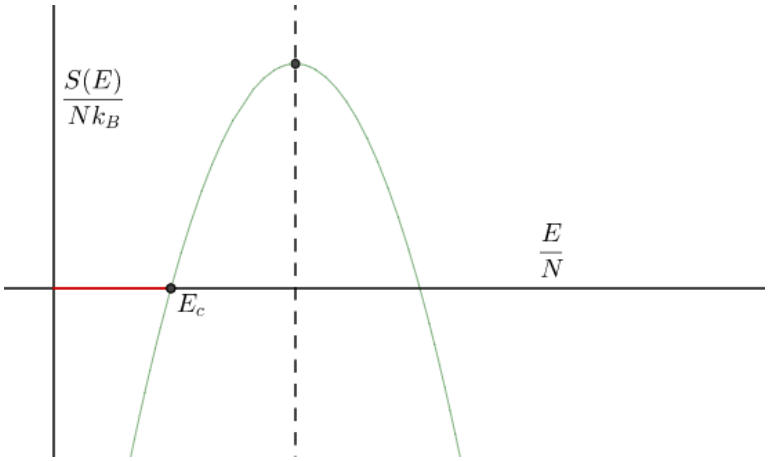

Figure 3.1: Shape of the entropy function. Only the first half positive part of the parabola is physical

Via equation (3.23) we can now easily obtain an approximate value for E_c, as $S(E_c) = 0$:

$$\frac{(E_c - N\mu)^2}{2NJ^2} = N \ln 2 \implies E_c - N\mu = NJ\sqrt{2 \ln 2}$$

And this expression leads to a critical temperature T_c:

$$\frac{1}{T_c} = \frac{dS}{dE}|_{E=E_c} = -k_B \left(\frac{E_c - N\mu}{NJ^2} \right) = \frac{k_B\sqrt{2 \ln 2}}{J}$$

This leads to a critical $\beta_c = \frac{\sqrt{2 \ln 2}}{J}$. Furthermore, note that $\frac{E_c}{N} = \mu + J\sqrt{2 \ln 2}$ is exactly the x_c we found in the first theorem of this chapter. This means that the assumption we made in (3.23) is true almost surely for $N \to \infty$.

Chapter 4

Application

4.1 Problem description

We will now take a deeper look into the problem of compact heteropolymers. A heteropolymer is a long molecule formed from subunits (monomers) that are not all the same, such as a protein composed of various amino acid subunits. The long strings of the proteins can interact with other proteins, making them fold together. Biophysicists usually view them as shown in Figure 4.1, which simplifies the underlying chemical structure. Interactions between proteins play a key role in many biological processes[1], as it is closely linked to the folding and copying of DNA. Due to the complexity of the structure of a protein, most models that describe protein folding require heavy mathematics, leading to high simulation and computing times. To avoid this, the REM was suggested in the context of heteropolymers.

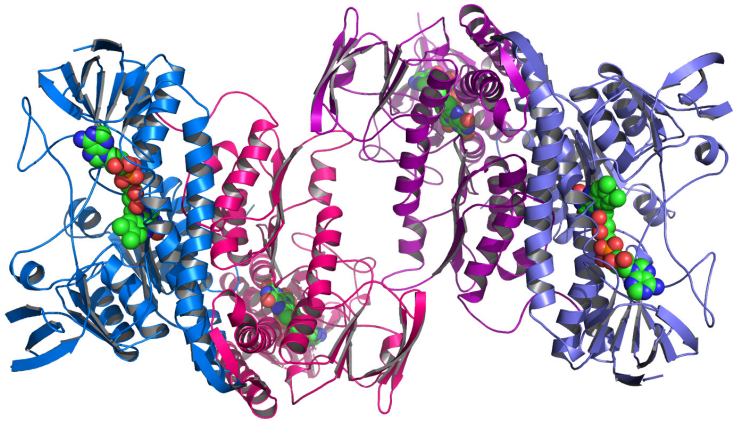

Figure 4.1: A typical example of folded proteins

We will first make a rough approximation about the shape of heteropolymers. In the light of this report, it should be enough to regard them shaped as chains of monomers. Looking at a collection of heteropolymers, the configuration is remarkably compact in many biological

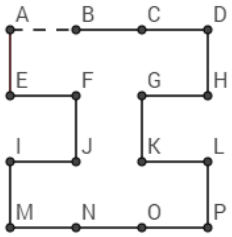

Figure 4.2: A compact configuration in 2D

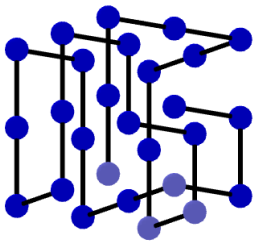

Figure 4.3: A compact configuration in 3D

processes. For instance, a typical human DNA molecule would have an approximate length of 0.2 mm[9] if it behaved like a random chain, whereas a typical cell nucleus diameter is in the order of microns. Therefore, the structure of the folded polymers has to be extremely compact, which we can see mathematically as a collection of all maximally compact configurations on a lattice. Figure 4.1 shows a maximally compact configuration of a 4x4 lattice (N=16): every lattice point is visited once by the polymer. This Hamiltonian walk is by no means unique and does not necessarily have to be a circuit. For a fixed volume, a higher number of N will lead to a more closely packed lattice. In Figure 4.2 a 3D polymer on a 3x3x3 cubic lattice is shown. This 27-mer has become a standard for heteropolymer studies. In general biophysicists are most interested in three-dimensional globular shapes [20].

Every configuration of the polymers will cost a different amount of energy, so the way the polymers are structured will be dependent of temperature. For low temperatures, the low energy states will most likely be preferred. A decreasing temperature will force the polymers to rearrange and there could even exist a certain temperature at which the system 'freezes' into a single configuration: a transition to the ground state. Such a transition typically exists [21], which motivates the application of the REM.

4.2 REM for heteropolymers

To model a system of folding proteins, we start with stating that there are still two possibilities for each point on the lattice: it is either occupied by a certain monomer, or not. The state space thus remains $S_N = \{-1, +1\}^N$ (or one could rename -1 to 0). 1 represents a lattice point that is occupied by a monomer, -1 an empty spot. The energy of each configuration of the heteropolymer will be dependent on the interaction between neighbouring lattice points. The closer two monomers are, the more they will influence each other. Furthermore, interactions amongst proteins appear to be dominated by the short-range role of hydrophobicity[21]. For each pair of monomers we define the interaction energy B between the two momomers[1]. To avoid too many mathematical complications, we make a nearest-neighbour approximation.

[1]For now we will talk about monomers, but normally a single lattice point is not necessarily representing a single monomer, but a small arrangement of amino acids[22]. So to be precise, we should talk about "quasimonomers".

Let us define the Hamiltonian of a certain monomer s_i at a certain position $\boldsymbol{r}_i$ on the lattice in the following way:

$$H(s_i, \boldsymbol{r}_i) = \sum_{i,j}^{N} B_{ij} \Delta(\boldsymbol{r}_i - \boldsymbol{r}_j) \tag{4.1}$$

Where we have a finite number q of different monomers such that $s_i \in \{1, ..., q\}$ on the lattice. The dirac function Δ makes the nearest neighbour approximation: $\Delta(a) = 1$ and $\Delta(a > r) = 0$ where a is the lattice spacing.

The interaction energies B are still chosen according to the REM:

$$\mathbb{P}(B_{ij}) = \frac{1}{\sqrt{2\pi N J^2}} e^{-(B_{ij} - N\mu)^2 / 2NJ^2} \tag{4.2}$$

So, in order to apply the REM, we have to assume that the overall energy levels of the different configuration possibilities are Gaussian i.i.d. distributed (and thus do not influence each other). We also see now the necessity of adding a mean to the REM. We can interpret the variable μ as the mean interaction energy $\bar{B}$, which is typically nonzero.

4.3 Different states of the heteropolymers

In the heteropolymer model we used in this chapter, the total system is defined by two parameters: μ and J. Therefore, we expect the model to have three different phases. These 3 phases are respectively the folded, glassy and random phase and are indeed found in the REM[24].
In the random state, an exponential number of globular conformations dominate equilibrium. In the glassy state, conformations that are not the target conformation dominate below the critical temperature T_c. In the folded state, the target conformation dominates equilibrium.

4.4 Folding mechanism and the Metropolis algorithm

Let us return to the REM for folding of random heteropolymers and consider a system in 2D. We will study the folding behaviour of a polymer by using the Metropolis algorithm[25]. In our case, this algorithm first chooses a random configuration of a polymer string on the lattice. We will later adjust this configuration step by step, creating a Markov chain. We take the probability for the configuration X to arise according to the Boltzmann distribution, so the distribution ρ of the configurations is:

$$\rho(X) = e^{-\beta H\{X\}} \tag{4.3}$$

We now select a monomer randomly and calculate how much the total energy would change if we place it to another position. We then get a new configuration X', with probability $e^{-\beta H\{X'\}}$.
The difference in energy can easily be calculated using the Hamiltonian defined in equation (4.1):

$$\Delta E = H(X) - H(X') \tag{4.4}$$

But what is the probability of such a transition from configuration X to X' to occur? If $H(X') < H(X)$, the transition is energetically favourable, so we take the transition probability $T(X \to X')$ equal to 1.
If $H(X') > H(X)$, we have to take a closer look. It is unclear what the probability $T(X \to X')$ will be, but we know the probability distribution of being in configuration X at a certain step t in the Markov chain: $\rho(X,t)$. Two values of $\rho(X)$ at different time steps t and $t+1$ will differ because of transitions between state X and the possible states X'. The next equation, which is called the master equation, evaluates this difference in order to arrive at a time evolution equation.

$$\rho(X,t+1) - \rho(X,t) = -\sum_{X'} T(X \to X')\rho(X,t) + \sum_{X'} T(X' \to X)\rho(X',t) \tag{4.5}$$

We are looking for the folded state of the polymer, which is a stationary state (we assume the protein is stable once it has been folded). Therefore, the master equation should equal 0 and we immediately see a special solution:

$$T(X \to X')\rho(X) = T(X' \to X)\rho(X') \tag{4.6}$$

Note that the exact value of t is in this case not relevant, as the stationary solution should be valid for all $t \geq t_f$ for some t_f. Rewrite equation (4.6) to obtain:

$$\frac{T(X \to X')}{T(X' \to X)} = \frac{\rho(X')}{\rho(X)} = \frac{e^{-\beta H(X')}}{e^{-\beta H(X)}} = e^{-\beta(H(X')-H(X))} \tag{4.7}$$

We have assumed that $H(X') > H(X)$, so $T(X' \to X) = 1$. But then:

$$T(X \to X') = e^{-\beta(H(X')-H(X))} \tag{4.8}$$

We can now summarize the Metropolis algorithm in the following way. First choose an initial configuration. Then:

1. Select a random monomer and place it somewhere else (within restrictions of the polymeric bounds), creating a configuration X'.
2. Calculate the energy difference ΔE that this change in configuration would cause.
3. If $\Delta E < 0$, make X' the new distribution. Otherwise, make X' the new distribution with probability $e^{-\beta \Delta E}$.
4. Repeat.

By using this algorithm, we find a stationary solution that is energetically favourable to the system. We want to study under which circumstance (parameters β, chain length, ...) the proteins fold.

4.4.1 Notes on implementation

To generate an initial polymer configuration, a random chain on a lattice is generated. Every point on the lattice is represented by either a 0 (an empty space) or a letter (a certain monomer). Two examples of starting configurations are plotted in Figure 4.4. On the left is a 50-monomer-long polymer, on the right the polymer consists of 100 monomers. Both images show a 50×50 lattice, blue represents an open spot, yellow a monomer.
We chose for a lattice in which the upper row is connected to the lower row and the left column to the right column. In this way, the two dimensional space represents the spherical outer surface of a torus. We see this effect in the Figure on the right in Figure 4.4. In later simulations, the model could be extended to three dimensions as well.
The random heteropolymers were created by the program as shown in the Appendix. From a starting point on the lattice a random walk is generated, creating a random polymer.

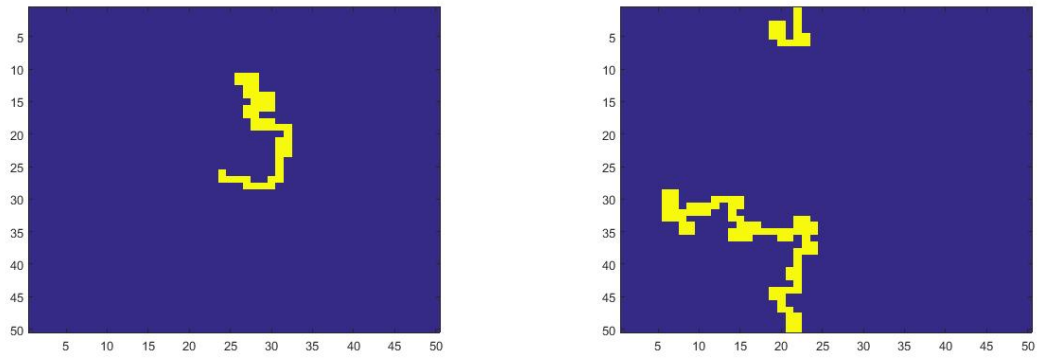

Figure 4.4: Two starting configurations on a 50×50 lattice. Blue lattice points represent an empty spot, yellow stands for a certain monomer.

Once an initial configuration is obtained, we can start using the Metropolis algorithm. A certain monomer is selected and placed to another position. The positions that are allowed by the polymeric bounds are the neighbouring lattice points, as long as they are not occupied by another monomer yet. Acceptance of the new structure is determined by using the energy expressions described in step 2 and 3 of the algorithm.
From some preliminary studies[26] the parameters were taken to be $\mu = -2$, $J = 1$, with units in quantities of $k_B T$. The used program can be found in the Appendix.

4.4.2 Simulation results

After running the algorithm for different values of β, sequences were unable to fold to a compact conformation within 10^8 Monte Carlo steps. Results give conformations of polymers that are very similar to the arbitrarily chosen initial configurations: the configuration are not observably more compact. As an example, the result of a simulation for $\beta = 0.01$ is shown below. If the value of β is increased, the effects are even smaller. This means our model is

not yet good enough to describe the folding process. In the next sections, we will investigate a better approach.

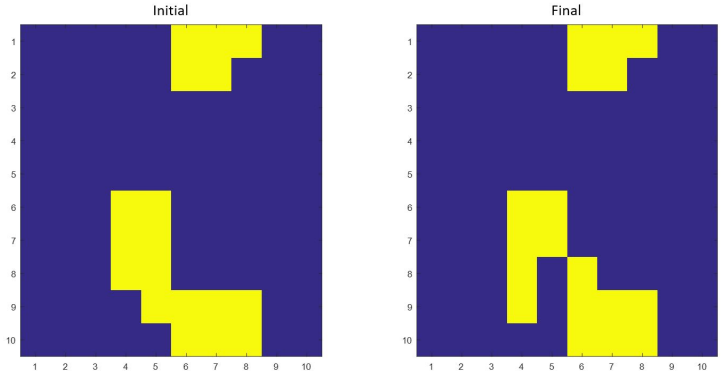

Figure 4.5: Simulation result for $\beta = 0.01$. The initial configuration on the left has barely changed compared to the final configuration after 10^8 steps.

4.5 Validity of the REM for heteropolymer folding

At first sight the REM can only be true for a system that actually behaves as if the interactions act independently[2]. It was shown that the REM is a good model for these kind of systems (see Shakhnovich and Gutin[23]) for small configurations. However, it is not always true that two polymer configurations have interactions that are independent from each other. Pande, Yu et al. [24] give a good example of that: take two polymer configurations that are very much alike except for a small rearrangement (let us call them 'similar packings'). These two configurations are of course close in energy. Therefore, the REM approach can only give an approximation to the compact folding problem. Luckily these kind of 'similar packings' are very rare[3] , which makes the REM a very good approximation in this setting.

However, the REM has some disadvantages in modelling protein folding. One of these is a problem known by biophysicists as the speed-stability paradox[27]: for a protein, the lowest possible energy state occurs significantly more frequently at a temperature below T_c. Thus, the folding of the polymer happens at a very low temperature, which means the dynamics of the system are slow as the number of states in the system decreases drastically and (as mentioned above) the different energy configurations do not have much in common. So, to change to the lowest energy state, the polymer has to rearrange many of its monomers. This contradicts the observation that most proteins fold easily and fast. Of course, the REM is created to describe random heteropolymers and naturally occurring polymers (such as DNA) are a special kind of polymers. It has been suggested that biological heteropolymers were

[2]The interactions at different sites are uncorrelated.

[3]This is due to the heavy constraints that the required compactness lies upon the configuration.

modified to more easily foldable versions over time due to evolution. We want to describe naturally occurring proteins with the REM, so in the next section we will extend our model.

4.6 Designed REM

To design a model that describes the folding of naturally occurring proteins in a better way, we add a single state with low energy. We call this energy the native energy E_n and it is now the lowest energy state possible ($E_n < E_c$). In the figure below, the new system is drawn in the same way as in Figure 3.1.
The probability to find the system in the native state now is $\frac{e^{-\beta E_n}}{Z_n(\beta)}$ with $Z_n(\beta) = e^{-\beta E_n} + Z(\beta)$ and $Z(\beta)$ the partition function of the normal REM.

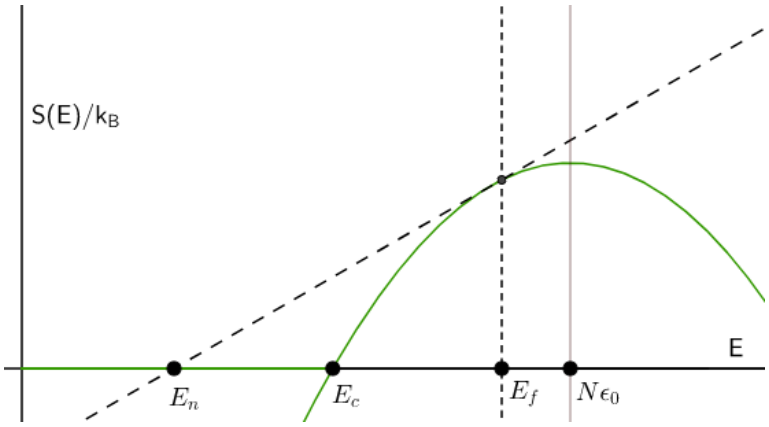

Figure 4.6: Shape of the entropy function for the designed REM. Again, only the first half positive part of the parabola is physical

What are the properties of this new system? First of all, there will be a new folding temperature T_f, which we want to compute. There will also be a new folding energy E_f, a new folding β_f, etc. Using a "tangent construction" shown in the above Figure, we see that $\frac{dS}{dE}|_{E_f} \simeq \frac{S(E_f)/k_B}{E_f - E_n}$. Now we can derive an expression for the folding temperature by using equation (3.23) for $E = E_f$:

$$\beta_f = \frac{1}{k_B T_f} = \frac{1}{k_B} \frac{dS}{dE}|_{E_f} = \frac{N \ln 2 - \frac{(E_f - N\mu)^2}{2NJ^2}}{E_f - E_n} \tag{4.9}$$

We now notice that $E = N\mu - NJ^2\beta$ (so $E_f = N\mu - NJ^2\beta_f$ and $E_n = N\mu - NJ^2\beta_n$) and $\frac{1}{2}(\beta_c\sigma)^2 = \ln 2$. Plug this into equation (4.9) to get:

$$\beta_f = \frac{N\frac{1}{2}(\beta_c\sigma)^2 - \frac{(N\mu - NJ^2\beta_f - N\mu)^2}{NJ^2}}{N\mu - NJ^2\beta_f - N\mu + NJ^2\beta_n} = \frac{\beta_c^2 - \beta_f^2}{2(\beta_n - \beta_f)}$$

This equation can be solved for β_f:

$$\beta_f^2 - 2\beta_n\beta_f + \beta_c^2 = 0$$

$$\beta_f = \beta_n \pm \sqrt{\beta_n^2 - \beta_c^2}$$

Where we should use the minus sign in the last equation, as the other one is not physical. From this expression we see that we can make the model fold at higher temperature levels by increasing the energy difference between E_n and E_c.

Chapter 5

Conclusions

Theoretically, it is possible to use the Random Energy Model for studying the folding of random heteropolymers. We proved that a version of the model in which the Hamiltonians satisfy a Gaussian probability distribution around a non-zero mean still leads to a phase transition at a critical point and found an expression for the free energy and entropy.
We wrote a simple model that simulates the behaviour of folding polymers. However, this model is not very accurate in describing the actual process. Other versions of the REM, such as the designed REM, may give a better result, although this was not proven in this thesis. Even without a solid simulation of the difficult protein folding process, the application of the REM in the field of random heteropolymers is still useful to gain insight in the different phases and the transition temperature of a polymer.
Properties of the REM heteropolymer model that are left to explore consist of a simulation of the designed REM as it was suggested in this thesis, the effect of not making the nearest-neighbour approximation or the extension of the simulation to 3D. Some mathematical consequences of the REM were intentionally left out of this thesis. For example, the distinction between annealed and quenched averages was not made, as was the whole concept of replica theory.

Bibliography

[1] Branden C and Tooze J. *Introduction to Protein Structure*. Garland Publishing, New York. 1999.

[2] Alberts, Bray et al. *Protein Structure and Function*. Essential cell biology. Garland Science, New York. 2010.

[3] B. Derrida. *Random-energy model; an exactly solvable model of disordered systems*. Phys. Rev. B (3), 24(5): 2613–2626. 1981.

[4] Nicola Kistler. *Derrida's random energy models. From spin glasses to the extremes of correlated random fields*. Frankfurt University. 2014.

[5] Anton Bovier. *Statistical Mechanics of Disordered Systems*.

[6] Marc Mézard and Andrea Montanari. *Information, Physics, and Computation*. Oxford University press, part B. 2009.

[7] Anton Bovier and Irina Kurkova. *Local energy statistics in disordered systems: a proof of the local REM conjecture*.

[8] Christian Borgs, Jennifer Chayes et al. *Proof of the local REM conjecture for number partitioning I*. 2008.

[9] M. Kardar. IITS course on Statistical Physics in Biology, lecture 4. KU Leuven. 2013. http://www.mit.edu/ kardar/teaching/IITS/lectures/lec4/lec4.pdf

[10] J. Cook and B. Derrida. *Finite-Size Effects in Random Energy Models and in the Problem of Polymers in a Random Medium* . J. Stat. Phys. 63, 505-539. 1991.

[11] Nabin Kumar Jana. *Contributions to Random Energy Models*. 2007.

[12] Daniel V. Schroeder. *An introduction to thermal phyics*. 2014.

[13] William Feller. *An Introduction to Probability Theory and Its Applications*, Vol. 1, 3rd Edition. Wiley.

[14] Frank den Hollander. *Large Deviations*. Vol 14. American Mathematical Society

[15] Jordan Bell. *Varadhan's lemma for large deviations*. 2015.

[16] Prof. Dr. Nina Gantert. Large Deviations, lecture notes. Winter 14/15. https://www-m14.ma.tum.de/fileadmin/w00biy/www/Lehre/ws14_15/Large_Deviations/largedevMar23.pdf

[17] Peter Morters. *Large deviation theory and applications*. November 10 2008.

[18] T.C. Dorlas and J.R. Wedagedera. *Large Deviations and the Random Energy Model*.

[19] H. McKean. *Stochastic Integrals*. Acad. Press. 1969.

[20] V. I. Abkevich, A. M. Gutin, and E. I. Shakhnovich. *Specific Nucleus as the Transition State for Protein Folding: Evidence from the Lattice Model?*. 1994.

[21] Hue Sun Chan and Ken A. Dill. *The protein folding problem*. Physics Today. 1993.

[22] Grosberg, A. Yu and A.R. Khokhlov. *Statistical Physics of Macromolecules*. American Institute of Physics, New York. 1994.

[23] Alexander Gutin. *Enumeration of all compact conformations of copolymers with random sequence of links*. The Journal of Chemical Physics 93, 5967. 1990.

[24] V.S. Prande, A. Yu, Grosberg, T. Tanaka. *Statistical Mechanics of Simple Models of Protein Folding and Design*. Biophysical Journal, Vol.73, 3192-3210. 1997.

[25] Jos Thijssen. *Computational Physics*. Cambridge University Press. 2007.

[26] E. Shakhnovich, G. Fartztdinov, A.M. Gutin. *Protein Folding Bottlenecks: A Lattice Monte Carlo Simulation*. Physical Review Letters. Volume 67, number 12. 1665-1668. 1991.

[27] Michael Slutsky and A. Mirny. *Kinetics of Protein-DNA Interaction: Facilitated Target Location in Sequence-Dependent Potential*. Biophysical Journal Volume 87, Issue 6, 4021-4035. 2004.

Chapter 6

Appendix

6.1 Matlab code for creating a random heteropolymer

```
function [state_space, positions] = create_polymer( polymer_length,
    state_space, numSpinsPerDim )
%Given a square lattice and a polymer length generates a random
    chain on
%the lattice
Alphabet=char('a'+(1:26)-1)';
[I,J]=ndgrid(1:26,1:26);
I=I'; J=J';
XX=[Alphabet(I(:)), Alphabet(J(:))];
XX=strvcat(Alphabet,XX);                    %Is now a random chain of
    letter, but can represent any set of monomers
polymer = XX(1: polymer_length);
start_x = round(numSpinsPerDim*rand)+1; %Random starting point on
    lattice
start_y = round(numSpinsPerDim*rand)+1;
state_space(start_x,start_y) = polymer(1);
row = start_x;
col = start_y;
positions = [row, col];                     %Quick localisation of
    monomers
for i = 1:(polymer_length-1)                %Generate random chain
    above = mod(row - 1 - 1, size(state_space,1)) + 1;
    below = mod(row + 1 - 1, size(state_space,1)) + 1;
    left  = mod(col - 1 - 1, size(state_space,2)) + 1;
    right = mod(col + 1 - 1, size(state_space,2)) + 1;
    neighbors = [above, below, left, right];
    check = 0;
```

```
    counter = 0;
    while check == 0;
        new = randi(4);
        if new == 1
            if state_space(above, col) == 0
                row = above;
            end
        elseif new == 2
            if state_space(below, col) == 0
                row = below;
            end
        elseif new == 3
            if state_space(row, left) == 0
                col = left;
            end
        elseif new == 4
            if state_space(row, right) == 0
                col = right;
            end
        end
    if state_space(row,col) == 0
    check = 1;
    state_space(row,col) = polymer(i+1);
    positions = [positions, row, col];
    end
    counter = counter +1;
    if counter > 100
        check =1;
    end
     end
end
positions
end
```

6.2 Metropolis algorithm

```
clear all; close all; clc;
numSpinsPerDim = 10;

%create initial random polymer
state_space = zeros(numSpinsPerDim, numSpinsPerDim);
polymer_length = 20;
[state_space, positions] = create_polymer(polymer_length,
    state_space, numSpinsPerDim);
```

```
8
9  %model parameters
10 N = numSpinsPerDim*numSpinsPerDim;
11 mu = -2;
12 J = 1;
13 initial = state_space;
14 initial_positions = positions;
15 new_state_space = state_space;
16 new_positions = positions;
17 beta = 0.1;
18 E = 0;
19 E_new = 0;
20 possib =[];
21 E_isolation = 0;
22
23 for i=1:100000
24     %pick a random monomer
25     pick_col = 2*randi(numel(positions)/2);
26     col = positions(pick_col);
27     row = positions(pick_col-1);
28     %find its neighbours
29     ab = mod(row - 1 - 1, size(state_space,1)) + 1;
30     be = mod(row + 1 - 1, size(state_space,1)) + 1;
31     le  = mod(col - 1 - 1, size(state_space,2)) + 1;
32     ri = mod(col + 1 - 1, size(state_space,2)) + 1;
33     neighbors = [ab, be, le, ri];
34     if state_space(ab,col) == 0;
35         possib =  [possib, ab, col];
36     end
37     if state_space(be,col) == 0;
38         possib =  [possib, be, col];
39     end
40     if state_space(row,le) == 0;
41         possib =  [possib, row, le];
42     end
43     if state_space(row, ri) == 0;
44         possib =  [possib, row, ri];
45     end
46     if numel(possib) ~=0
47         number = randi(numel(possib));
48         if mod(number,2) ==0
49             new_col = possib(number);
50             new_row = possib(number-1);
51         end
```

```
52          if mod(number,2) ==1
53              new_row = possib(number);
54              new_col = possib(number+1);
55          end
56          new_state_space(new_row,new_col) = state_space(row,col);
57          new_state_space(row,col) =0;
58          new_positions(pick_col) = new_col;
59          new_positions(pick_col -1) = new_row;
60      end
61
62      %check energy
63      for row = 1:numSpinsPerDim
64              for col = 1:numSpinsPerDim
65                  n1 = normrnd(N*mu,sqrt(N)*J);
66                  n2 = normrnd(N*mu,sqrt(N)*J);
67                  n3 = normrnd(N*mu,sqrt(N)*J);
68                  n4 = normrnd(N*mu,sqrt(N)*J);
69                  if state_space(row, col) ~= 0
70                      E_above = state_space(mod(row - 1 - 1, size(
                            state_space,1)) + 1,col)*n1;
71                      E_below = state_space(mod(row + 1 - 1, size(
                            state_space,1)) + 1,col)*n2;
72                      E_left  = state_space(row, mod(col - 1 - 1,
                            size(state_space,2)) + 1)*n3;
73                      E_right = state_space(row, mod(col + 1 - 1,
                            size(state_space,2)) + 1)*n4;
74                      E = E + E_above + E_below + E_left + E_right;
75                  end
76                  if new_state_space(row, col) ~= 0
77                      E_above_new = new_state_space(mod(row - 1 - 1,
                            size(new_state_space,1)) + 1,col)*n1;
78                      E_below_new = new_state_space(mod(row + 1 - 1,
                            size(new_state_space,1)) + 1,col)*n2;
79                      E_left_new  = new_state_space(row, mod(col - 1
                            - 1, size(new_state_space,2)) + 1)*n3;
80                      E_right_new = new_state_space(row, mod(col + 1
                            - 1, size(new_state_space,2)) + 1)*n4;
81                      E_new = E_new + E_above_new + E_below_new +
                            E_left_new + E_right_new;
82                  end
83              end
84          end
85      %better? If not: probability
86      if E_new < E
```

```
87          state_space = new_state_space;
88          positions = new_positions;
89      else
90          prob = exp(-beta*(E_new-E));
91          if rand <= prob
92              state_space = new_state_space;
93              positions = new_positions;
94          end
95      end
96      new_state_space = state_space;
97      new_positions = positions;
98      possib =[];
99      E_isolation =0;
100 end
101
102 figure(1)
103 subplot(1,2,1)
104 title('Initial')
105 image(initial)
106 subplot(1,2,2)
107 title('Final')
108 image(state_space)
```