Statistical Physics for Communications, Signal Processing, and Computer Science

EPFL

Nicolas Macris and Rüdiger Urbanke

Contents

Foreword

Statistical physics, over more than a century, has developed powerful techniques to analyze systems consisting of many interacting "particles." In the last twenty years it has become increasingly clear that the very same techniques can be applied successfully to problems in engineering such as communications, signal processing, or computer science.

However, there are several hurdles which one encounters when one tries to make use of these methods.

First, there is the language. Statistical physics has developed over the last 150 years with the aim of providing models and deriving predictions for various physical phenomena, such as magnetism or the behavior of gases. This long history, together with the specific areas of their original application, has resulted in a rich language whose origins and meaning are not always clear to someone just starting in the field. It therefore takes a considerable effort to learn this language.

Second, except for extremely simple models, the "calculations" which are necessary are often long and daunting and frequently use little tricks and conventions somewhat outside the realm what one usually picks up in a calculus class. A good way of overcoming this difficulty is to start with a familiar example, casting it in terms of statistical physics notation, and by then going through some basic calculations.

Third, and connected to the second point, not all methods and tricks used in the calculations are mathematically rigorous. Some of the most powerful techniques, such as the cavity method, currently do not have a mathematical justification. In the "right hands" they can do miracles and give predictions which are currently not possible to derive with any classical method. But a newcomer to the field might quickly despair in trying to figure out what parts are mathematical rigorous and what parts are "most likely correct" but cannot currently be justified. Both worlds are valuable. The cavity or replica method give predictions which would be very difficult to guess. These predictions can then be used as a starting point for a rigorous proof. But it is important to cleanly separate the two worlds.

Our aim in writing these notes is not to give an exhaustive account of all there is to know about statistical mechanics ideas applied to engineering problems.

Several excellent books already exist. We in particular recommend (Nishimori 2001, MacKay 2003, Mézard & Montanari 2009).

Our aim was to write the simplest non-trivial account of the most useful statistical mechanics methods so as to ease the transition for anyone interested in this strange put powerful world. Therefore, whenever we were faced with an option between completeness and simplicity, we chose simplicity. On purpose our language changes progressively throughout the text. Whereas at the beginning we try to avoid as much jargon as possible, we progressively start talking like a physicist. Most of the literature uses this language, so you better get used to it.

We decided to structure our notes around three important problems, namely error correcting codes, compressive sensing, and the random K-SAT problem. Although we will introduce basic versions of each of these problems, we only introduce what is necessary for our purpose. It goes without saying that there are countless versions and extensions that we do not discuss. In fact, we hope that the reader is already somewhat familiar with these topics and accepts that these are important problems worth while studying. Using the basic versions of these problems we explain how they can be cast in a statistical physics framework and how standard concepts and techniques from statistical physics can be used to study these problems. This allows us to introduce the necessary terminology step by step, just when it is needed.

The notes are further partitioned into three parts. In the first part, comprised of Chapters 1-4, we introduce the problems, some of the language, and we rewrite these problems in the language of statistical physics. In the first chapter of the second part, namely Chapter 5, we then introduce the main protagonist, a message-passing algorithm which is also know as the *belief-propagation* algorithm. The remaining chapters of the second part, namely Chapters 6-9.3, contain the analysis of the performance of our three problems under this low-complexity algorithm. We will see that, in many cases, even this simple combination yields excellent performance. Finally, in the third part, consisting of Chapters 11-13, we get to the perhaps most surprising part of our story. Our aim will be to study the fundamental behavior of these three problems without the restriction to low complexity algorithms. I.e., how well would these systems work under optimal processing. The surprise is that the same quantities which appeared in our study of low-complexity suboptimal message-passing algorithms will play center stage also for this seemingly completely unrelated question.

Although we follow essentially the same pattern for each of the three problems, we will see that they are not all equally difficult.

Error correcting coding is perhaps easiest, and in principle most of the question one might be interested in can be answered rigorously. In this case we are dealing with large graphically models which are locally "tree like." It is therefore perhaps not so surprising that message-passing algorithms work well in this setting and that the performance can be analyzed.

Compressive sensing follows a similar pattern but introduces a few more wrinkles. In particular, the story of compressive sensing is leading to the so-called

AMP algorithm. The surprising fact here is that message-passing works very well, and that its performance can be predicted, despite that the relevant graphical model is not sparse at all but rather is a complete tree. The key observation is that every single edge contributes very little to the global performance. AMP can still be analyzed rigorously but the required computations are quite lengthy. We will give an outline of the whole story, but we will not discuss every single step in detail. Once the basic idea is clear, the interested reader should be able to fill in missing details by studying the pointers to the literature.

The hardest problem is without doubt the random K-SAT problem. We will only be able to present a partial picture. Many interesting and very basic questions remain open.

Many people have helped us in creating these notes. In the Spring of 2011 we gave a series of lectures on these topics at EPFL to mostly a graduate student population. We would like to thank Marc Vuffray, Mahdi Jafari, Amin Karbasi, Masoud Alipour, Marc Desgroseilliers, Vahid Aref, Andrei Giurgiui, Amir Hesam Salavati for typing up initial notes for some lectures. In addition we would like to thank Mike Bardet who typed up further material as well as Hamed Hassani who has since contributed material to several of the chapters.

Nicolas Macris, Lausanne, 2017
Rüdiger Urbanke

Part I

Models and their Statistical Physics Formulations

1 Models, Questions, and Overview

This chapter introduces three problems: error correcting *coding*, *compressive sensing*, and *random constraint satisfaction*. All three problems play a fundamental role and have emerged as important paradigms in their respective disciplines: communications, signal processing, and theoretical computer science. Although the three problems are quite different, it turns out that essentially the same concepts and tools from statistical physics can be used to analyze and make quantitative predictions. It is no coincidence that statistical physics provides a unifying point of view on these problems. Indeed, they all involve the study of probability distributions, over a large number of random variables, with a structure akin to the probability distributions studied in the framework of statistical physics.

The connections between statistical physics, computation and information are not new, and were already recognised a decade after Shannon's 1948 foundational article on information theory (a few historical pointers are given in the Notes at the end of the chapter). Such connections were pushed towards engineering problems in communications and signal processing during the late 1990's and flourished since then, and it seems already impossible to review the whole subject. We concentrate on three essential models which form our guiding line, and should provide the reader with the necessary basis to understand the literature and solve further problems. In each domain of interest the models are paradigmatic, have important practical applications, and beautifully connect to some of their cousin statistical physical models. As is often the case with paradigms the models are simple enough that they can be formulated from scratch, and almost no prior knowledge in any of the above mentioned disciplines is needed.

In the next three sections we define each problem and outline a few fundamental questions that will be addressed in the next chapters. We then present a high level overview of the developments in the next chapters. If the discussion sounds difficult to follow at first, for one that is not familiar with the subject, we hope that refering back to it will help to grasp the general picture.

1.1 Coding

Error correcting codes

Codes are used to reliably transmit information across a noisy channel. The basic idea is to add some further bits to the information to be transmitted and to use these extra *redundant* bits to reconstruct the original information from the "noisy" observation.

A *binary block code* $\mathcal{C}$ of length n is a collection of binary n-tuples, $\mathcal{C} = \{\underline{x}^{(1)}, \ldots, \underline{x}^{(\mathcal{M})}\}$, where $\underline{x}^{(i)}$, $1 \leq i \leq \mathcal{M}$, is called a codeword, and where the components of each codeword are elements of $\mathbb{F}_2 = (\{0,1\}, \oplus, \times)$, the binary field. The number of codewords is hence $\mathcal{M} = |\mathcal{C}|$ the cardinality of $\mathcal{C}$. The *rate* of the code is defined as $\frac{\log_2 |\mathcal{C}|}{n}$. This is the ratio of the number of *information* bits per transmitted bit.

We will soon talk about various channel models, in other words, various mathematical models that describe how information is "perturbed" during the transmission process. In this respect it is good to know that for a large class of such models we can achieve optimal performance (in terms of the rate we can reliably transmit) by limiting ourselves to a simple class of codes, namely *linear* codes.

A *linear* binary block code is a subspace of $\mathbb{F}_2^n$, the vector space of dimension n over the field $\mathbb{F}_2$. Equivalently, a binary block code $\mathcal{C}$ is linear iff for any two codewords $\underline{x}^{(i)}$ and $\underline{x}^{(j)}$, we have $\underline{x}^{(i)} - \underline{x}^{(j)} \in \mathcal{C}$. In particular $\underline{x}^{(i)} - \underline{x}^{(i)} = \underline{0}$ the vector with all-zero components always belongs to a linear code. Since $\mathcal{C}$ is a subspace, it has a dimension, call it k, $0 \leq k \leq n$. Hence $|\mathcal{C}| = 2^k$, and the rate of $\mathcal{C}$ is equal to $\frac{k}{n}$. All codes which we consider in this course are binary and linear. Therefore, in the sequel we sometimes omit these qualifiers. We will view vectors $\underline{x}$ as *column* vectors. For *row* vectors we write $\underline{x}^T$.

It will be convenient to represent a linear binary code $\mathcal{C}$ of length n and dimension k as the kernel (or null space) of an $(n-k) \times n$ binary matrix of rank $n-k$. Such a matrix is called a *parity-check* matrix and is usually denoted by H. Every binary linear code has such a representation (because any linear subspace is the null space of some matrix). So equivalently, we may write

$$\mathcal{C} = \{\underline{x} \in \mathbb{F}_2^n : H\underline{x} = \underline{0}\}$$

for some suitably chosen $(n-k) \times n$ binary matrix H of rank $(n-k)$.

A few remarks might be in order. First, once we have convinced ourselves that there is at least one such matrix, it is easy to see that there are exponentially many (in $n-k$) such matrices since elementary row operations do not change the row space and hence the code defined by the matrix. All these matrices define the same code, and are equivalent in this sense. But the representation of the code in terms of a bipartite graph, which we will introduce shortly, and the related decoding algorithm, do depend on the specific matrix we choose and so our choice of matrix is important.

Second, and somewhat connected to the first point, rather than first defining

a code $\mathcal{C}$ and then finding a suitable parity-check matrix H, we typically specify directly the matrix H and hence indirectly the code $\mathcal{C}$.

It can then happen that this matrix does not have full row rank, i.e., that its rank is strictly less than $n-k$. What this means is that the code $\mathcal{C}$ contains more codewords than 2^k. Since this will happen rarely, and since having more codewords than "initially planned" is in fact a good thing, we will ignore this possibility and only count on having 2^k codewords at our disposal.

The factor graph associated to the parity-check matrix H (of a code $\mathcal{C}$)

Assume that we have a code $\mathcal{C}$ defined by the $(n-k) \times n$ binary parity-check matrix H. We can associate to H the following bipartite graph G. The graph G has vertices $V \cup C$, where $V = \{x_1, \ldots, x_n\}$ is the set of n *variable* nodes corresponding to the n bits (and hence to the n columns of H), and where $C = \{c_1, \ldots, c_{n-k}\}$ is the set of $n-k$ *check* nodes, each node corresponding to one row of H. There is an edge between x_i and c_j if and only if $H_{ji} = 1$. The parity-check matrix H and the graph G encode the same information. But we will see that G is the natural starting point for introducing the low-complexity decoding algorithms.

EXAMPLE 1 (Factor Graph) Consider the following parity-check matrix,

$$H = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

The factor graph corresponding to H is shown in Fig. 1.1. □

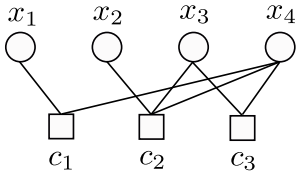


Figure 1.1 The factor graph corresponding to the parity-check matrix of Example 1.

Gallager's ensemble and the configuration model

A common theme in these notes is that instead of studying specific instances of a problem we define an *ensemble* of instances, in other words a set of instances endowed with a probability distribution. We then study the average "performance" of this ensemble, and once the average is determined, we know that there must be at least one element of the ensemble with a performance at least as good as this average. In fact, in many cases, with extra effort one can show that most elements in the ensemble behave almost as good as the ensemble average.

For coding, we focus on a specific ensemble of codes called the (d_v, d_c)-regular *Gallager* ensemble introduced by Gallager in the 1960's. Rather than specifying the codes directly, we specify their factor graphs. The ensemble is characterized by the triple of integers (n, d_v, d_c) such that $m = n\frac{d_v}{d_c}$ is also an integer. The parameter n is the length of the code, d_v is the variable node degree, and d_c is the check node degree.

We describe the ensemble by explaining how to sample from it. Pick n variable nodes and $m = n\frac{d_v}{d_c}$ check nodes. Each variable node has d_v *sockets* and each check node has d_c *sockets*. Number the $d_v n$ variable sockets in an arbitrary but fixed way from 1 till $d_v n$. Do the same with the $d_c m$ check node sockets. Pick a permutation π uniformly at random from the set of permutations on $d_v n$ letters. For $s \in \{1, \ldots, d_v n\}$ insert an edge which connects variable node socket s to check node socket $\pi(s) \in \{1, \ldots d_c m\}$.

If, after construction, we delete sockets (and retain the connections between variable and check nodes) then we get a bipartite graph which is the factor graph representing our code. To this bipartite graph we can of course associate a parity-check matrix H. But note that in this model there can be multiple edges between nodes. Can such a graph be meaningfully interpreted as representing a code?

A moments thought shows that the natural interpretation of such a graph in terms of a parity-check matrix H is to say that H has a 1 at row j and column i if there are an odd number of connections between constraint j and variable i. Otherwise it has a 0 at this position. In practice multiple connections are not desirable and more sophisticated graph generation algorithms are employed. But for our purpose the typically small number of multiple connections will not play a role. In particular, it does not play a role if we are interested in the behavior of such codes for very large instances.

The above way of specifying the Gallager ensemble is inspired by the *configuration model* of random graphs. This particular ensemble is a special case of what is called a *low-density parity-check* (LDPC) ensemble. This name is easily explained. The ensemble is *low-density* since the number of edges grows linearly in the block length. This is distinct from what is typically called the Fano random ensemble where each entry of the parity-check matrix is chosen uniformly at random from $\{0, 1\}$, so that the number of edges grows like the square of the block length. It is further a parity-check ensemble since it is defined by describing the parity-check matrix. We will see that a reasonable decoding algorithm consists of sending messages along the edges of the graph. So few edges means low complexity and, even more importantly, we will see that the algorithm works better if the graph is *sparse*.

For many real systems, LDPC codes are the codes of choice. They have a very good trade-off between complexity and performance and they are well suited for implementations. "Real" LDPC codes are often further optimized. For example, instead of using regular degrees we might want to choose nodes of different degrees and the connections are often chosen with care in order to minimize

complexity and to maximize performance. We will ignore these refinements in the sequel. The most important trade-offs are already apparent for the relatively simple regular Gallager ensemble.

Encoding, Transmission, and Decoding

The three operations involved in the coding problem are *encoding*, *transmission over a channel*, and *decoding*. Let us briefly discuss each of them.

Encoding: Given a binary linear block code $\mathcal{C}$ of dimension k, we can *encode* k bits of information by our choice of codeword, i.e., by choosing one out of the 2^k possibilities. More precisely, we have an information word $\underline{u} \in \mathbb{F}_2^k$, and an encoding function $g : \mathbb{F}_2^k \to \mathcal{C}$, which maps each information word into a codeword.

Although this function is of crucial importance for real systems, it only plays a minor role for our purposes. This is true since, as we will discuss in more detail later on, for "typical" channels, by symmetry the performance of the system is independent of the transmitted codeword. We therefore typically assume that the all-zero codeword (which is always contained in a binary linear code) was transmitted. Also, in terms of complexity, the encoding operation is not a difficult task. One possible option is to write the linear binary code in the form $\mathcal{C} = \{G\underline{u} : \underline{u} \in \mathbb{F}_2^k\}$, where G is the so-called *generator* matrix and where $\underline{u}$ is a binary column vector of length k which contains the information bits. In this form, encoding corresponds to a multiplication of a vector of length k with a $n \times k$ binary matrix G and can hence be implemented in $O(k \times n)$ binary operations. In practice the code is often chosen to have some additional structure so that this operation can even be performed in $O(n)$ operations. We will hence ignore the issue of encoding in the sequel.

Transmission over a Channel: We assume that we pick a codeword $\underline{x}$ uniformly at random from the code $\mathcal{C}$. We transmit $\underline{x}$ over a *channel*. This channel is a physical device that takes bits as inputs, converts them into a physical quantity, such as an electric or optical signal, transmits this signal over a suitable medium, such as a cable or optical fiber, and then converts the physical signal back into a number that can be processed, perhaps a voltage which is measured or the number of photons that were detected.

During the transmission the signal is distorted. This distortion is either due to imperfections of the system or due to unpredictable processes such as thermal noise. Instead of considering this potentially very complicated process in all its detail we use a simple mathematical model that summarizes the end-to-end effect of all these physical processes. We call this mathematical model the "channel model."

Formally, the channel has an *input alphabet*. We will always assume that the input alphabet is binary, $\mathcal{X} = \{0, 1\}$. The channel also has an *output alphabet* $\mathcal{Y}$. Two common cases are $\mathcal{Y} = \{0, 1\}$ and $\mathcal{Y} = \mathbb{R}$. We assume that the channel

is *memoryless*, which means that it acts on each bit independently. We further assume that there is no *feedback* from the output of the channel back to the input. In this case the channel is characterized by a transition probability $p(\underline{y} \mid \underline{x})$ where $\underline{y} \in \mathcal{Y}^n$ is the output and where

$$p(\underline{y} \mid \underline{x}) = \prod_{i=1}^{n} p(y_i \mid x_i). \tag{1.1}$$

This product form accurately models the relationship between channel input and output only if the channel is memoryless and there is no feedback.

The following three channels are the most important examples, both from a theoretical perspective, but also because they form the basis of real-world channels. These are the *binary erasure channel* (BEC), the *binary symmetric* channel (BSC) and the *binary additive white Gaussian noise channel* (BAWGNC).

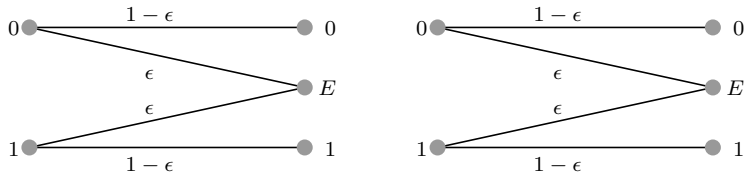


Figure 1.2 Binary erasure and symmetric channels with parameter ϵ. Both channels have binary inputs. The output alphabet for the BEC is $\mathcal{Y} = \{0, E, 1\}$ and for the BSC $\mathcal{Y} = \{0, 1\}$

BEC. The BEC is a very special channel with output alphabet $\mathcal{Y} = \{0, E, 1\}$. As depicted in Fig. 1.2, the transmitted bit is either correctly received at the channel output with probability $1-\epsilon$ or erased by the channel with probability ϵ and thus, nothing is received at the channel output. The erased bits are denoted by "E". For example, if $x = 1$ is transmitted then the set of possible channel observation is $\{1, E\}$. We can write the transition probability in the somewhat formal way

$$p(y|x) = (1-\epsilon)\delta(y-x) + \epsilon\delta(y-E).$$

BSC. The output of the BSC is binary $\mathcal{Y} = \{0, 1\}$. As seen in Fig. 1.2 the bit is transmitted correctly with probability $1-\epsilon$ or flipped with probability ϵ. The transition probability is

$$p(y|x) = (1-\epsilon)\delta(y-x) + \epsilon\delta(y-(1-x)).$$

BAWGNC. The output is a real number $\mathcal{Y} = \mathbb{R}$. When $x \in \{0, 1\}$ is sent the received signal is $y = x + z$ with z a Gaussian random number with zero mean and variance σ^2. With these conventions the "signal to noise ratio" is σ^{-2} and the transition probability

$$p(y|x) = (\sqrt{2\pi}\sigma)^{-1} e^{-\frac{(y-x)^2}{2\sigma^2}}.$$

One might wonder if these three simple models even scratch the surface of the rich class of channels that one encounters in practice. Fortunately, the answer is *yes*. Communications theory has built up a rich theory of how more complicated scenarios can be dealt with assuming that we know how to deal with these three simple models.

Decoding: Given the output $\underline{y}$ we want to map it back to a codeword $\underline{x}$. Let $\widehat{\underline{x}}(\underline{y})$ denote the function which corresponds to this *decoding* operation. What decoding function shall we use?

One option is to first pick a suitable criterion by which we can measure the performance of a particular decoding function and then to find decoding functions which optimize this criterion. The most common such criteria are the *block error probability* $\mathbb{P}\left[\widehat{\underline{x}}(\underline{Y}) \neq \underline{X}\right]$, and the *bit error probability* $\frac{1}{n}\sum_{i=1}^{n}\mathbb{P}\left[\widehat{x}_i(\underline{Y}) \neq X_i\right]$ (capital letters $\underline{X}$ and $\underline{Y}$ denote *random variables* associated to the channel input and output). The decoding functions that minimize these two criteria are the *block MAP* and the *bit MAP* decoder. For generic codes, no algorithms are known that implement these decoders in less than exponential time. We will discuss this in more detail in Chapter 3.

In practice, due to complexity constraints, it is typically not possible to implement an optimal decoding function but we have to be content with a low-complexity alternative. Of course, the closer we can pick it to optimal the better.

Shannon Capacity

So far we have defined codes, we have discussed the encoding problem, the process of transmission, the decoding problem, and the two most standard criteria to judge the performance of a particular decoder, namely the block and the bit error probability.

It is now natural to ask what is the *maximum rate* at which we can hope to transmit reliably, assuming that we pick the best possible codes and the best possible decoder. Reliably here means that we can make the block or bit probability of error as small as we desire. In fact, it turns out that the answer is the same whether we use the block error probability or the bit error probability.

In 1948 Shannon gave the answer and he called this maximum rate the *capacity* of the channel. For binary-input memoryless output-symmetric channels the capacity has a very simple form. If the input alphabet is binary and the output alphabet discrete, and if $p(y|x)$, $x \in \mathcal{X} = \{0,1\}$ and $y \in \mathcal{Y}$, denotes the transition probabilities, then the capacity of the associated channel can be expressed (in bits per channel use) as

$$C_{\text{channel}} = H\big(\frac{1}{2}p(\cdot|0) + \frac{1}{2}p(\cdot|1)\big) - \big\{\frac{1}{2}H(p(\cdot|0)) + \frac{1}{2}H(p(\cdot|1)\big\} \tag{1.2}$$

where $H(q(\cdot))$ denotes the entropy associated to a discrete distribution $q(y)$,

$y \in \mathcal{Y}$. By definition

$$H(q(\cdot)) = -\sum_{y\in\mathcal{Y}} q(y)\log_2 q(y). \tag{1.3}$$

Let us illustrate Shannon's formula for the three important channels introduced above.[1]

For the BEC the distribution entering in the first entropy on the right hand side of (1.2) is $q(0) = q(1) = \frac{1}{2}(1-\epsilon)$ and $q(E) = \epsilon$. This gives $H(q(\cdot)) = (1-\epsilon)+h_2(\epsilon)$ where $h_2(\epsilon) = -\epsilon\log_2\epsilon - (1-\epsilon)\log_2(1-\epsilon)$ is called binary entropy function. We still have to subtract the bracket which is the average of two entropies. For the first of these $q(x = 0) = p(0|0) = 1-\epsilon$, $q(1) = p(y = 1|0) = 0$, and $q(E) = p(E|0) = \epsilon$, so $H(p(\cdot|0)) = h_2(\epsilon)$. Similarly $H(p(\cdot|1)) = h_2(\epsilon)$. The average is $h_2(\epsilon)$ and we conclude that the Shannon capacity of the BEC is

$$C_{\mathrm{BEC}} = 1-\epsilon$$

That the capacity is at most $1-\epsilon$ for the BEC is intuitive. For large blocklengths with high probability the fraction of non-erased positions is very close to $1-\epsilon$. So even if we knew a priori which positions will be erased and which will be left unperturbed, we could not hope to transmit more than $n(1-\epsilon)$ bits over such a channel. What is perhaps a little bit surprising is that this quantity is achievable: we do not need to know a priori what positions will be erased and still can transmit reliably at this rate.

The capacities of the BSC and BAWGNC are given by the formulas,

$$C_{\mathrm{BSC}} = 1-h_2(\epsilon), \qquad C_{\mathrm{BAWGNC}} = 1-\int_{-\infty}^{+\infty} dy\, \frac{e^{\frac{-y^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma}\log_2\big(1+e^{\frac{y}{\sigma^2}-\frac{1}{2\sigma^2}}\big).$$

Their derivation is left as an exercise for the reader.

Questions

For the purpose of communication we are interested in codes that allow transmission close to capacity using only low-complexity encoding and decoding algorithms and have a very low probability of error. So perhaps the most natural question is if the types of codes we discussed are suitable for this purpose. The answer is a resounding *yes*, and indeed these are the codes that are used in all practical systems. In order to arrive at this answer we will have to answer a sequence of related questions: What (low-complexity) decoding algorithm should we use? How can we analyse the performance of such codes and how does the performance depend on the code parameters?

We will be able to derive a fairly complete and satisfying picture. We will in

[1] These three channels belong to the important class of *symmetric* binary input memoryless channels. For such channels the capacity formula can be simplified down to $C_{\mathrm{channel}} = 1-\sum_{y\in\mathcal{Y}} p(y|0)\log_2(1+\frac{p(y|1)}{p(y|0)})$. The sum stands for an integral when the output is continuous. See exercises.

particular see that, as the code length tends to infinity, the performance exhibits a "threshold" behavior as shown in Fig. 1.3. I.e., we will be able to decode with high probability if the channel quality is above this threshold, but will fail with high probability if the channel quality is below this threshold. We will further be able to determine this threshol for various decoding algorithm and so be able to compare it to the Shannon limit.

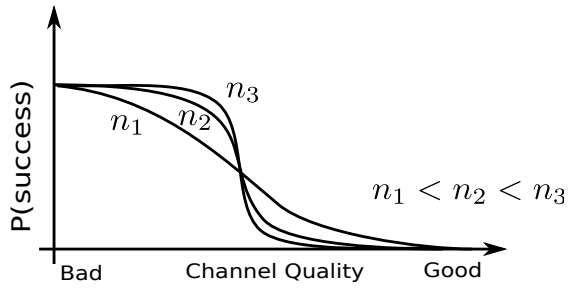


Figure 1.3 The probability of decoding error for a transmitted message versus the channel quality. As the blocklength of the code gets larger, we expect to see a sharper and sharper transition between range of the channel parameters where the system "works" and where it "breaks down."

1.2 Compressive sensing

Traditionally "sensing" and "compression" are two separated operations. E.g., if you take a photo, you first measure (sense) the light that impinges on the image sensor of your camera. Modern cameras have image sensors with millions of pixels and each pixel value measures various spectral components at a high resolution (at least 8 bits). As a result, the "raw" image of a modern camera is a large file. But you also know that you can compress an image significantly without a noticeable loss of quality. This is so since the various pixel values are not independent quantities but are highly correlated. This begs the question whether it is really necessary to collect a large amounts of data if at the end a much smaller amount suffices. In other words, can we combine the sensing with the compression step? Indeed, we can, and this idea is known as *compressive sensing*.

Basic problem

Here is perhaps the simplest version of compressive sensing.

Let $\underline{x}^{\text{in}} \in \mathbb{R}^n$ representing an "input signal" that we want to capture. We assume that the number of *non-zero* components[2]

$$\|\underline{x}^{\text{in}}\|_0 = |\{i|x_i^{\text{in}} \neq 0, i = 1, \ldots, n\}| \leq k$$

[2] This is not a norm (it is not homogenous, i.e., scaling the signal does not scale the

of the signal is at most a (fixed) fraction of n, so $k = \kappa n$ with $\kappa < 1$ (and usually much smaller than one). Such signals are called *k-sparse*. The signal is captured or measured thanks to an $m \times n$ "measurement matrix" A with real entries, $1 \leq m < n$, $m = \mu n$ with $0 < \mu < 1$,

$$\underline{y} = A\underline{x}^{\text{in}}.$$

We think of $\underline{y} \in \mathbb{R}^m$ as the result of m linear measurements, one corresponding to each row of A. Our basic aim is to reconstruct the k-sparse signal $\underline{x}^{\text{in}}$ from the least possible measurements $\underline{y}$.

We know that at least one solution exists, namely $\underline{x}^{\text{in}}$, because the measurements $\underline{y}$ have been produced by this input signal. But since $m < n$, and in fact m is typically *much smaller*, we cannot simply solve the undetermined linear system of equations since the solution will not be unique. But we know in addition that $\underline{x}$ has at most k non-zero entries entries with $k < n$ (we do not know which of these entries are non-zero), and it could conceivably be the case that among k-sparse vectors the solution is unique. Therefore, we determine if the set of possible signals, namely

$$\{\underline{x} : A\underline{x} = \underline{y} \text{ and } \|\underline{x}\|_0 \leq k\}. \tag{1.4}$$

has cardinality one. If this is the case we may in principle be able to reconstruct our signal unambiguously.

One way to ensure the unicity of the solution is to take a measurement matrix A satisfying a *Restricted Isometry Property* (RIP). We say that A satisfies the RIP$(2k, \delta_{2k})$ condition if one can find $0 \leq \delta_{2k} < 1$ such that

$$(1 - \delta_{2k})\|\underline{x}\|_2 \leq \|A\underline{x}\|_2 \leq (1 + \delta_{2k})\|\underline{x}\|_2, \quad \text{for all } 2k\text{-sparse vectors } \underline{x} \in \mathbb{R}^n. \tag{1.5}$$

It is not difficult to see that when this condition is met, then the set (1.4) contains a *unique* element given by

$$\widehat{\underline{x}}_0(y) = \text{argmin}_{\underline{x}: A\underline{x} = y}\|\underline{x}\|_0. \tag{1.6}$$

Indeed, first notice that evidently $A\widehat{\underline{x}}_0(y) = y$ so we only have to prove unicity. Suppose $\underline{x}'(\underline{y})$ is another element in (1.4). Then, since both $\underline{x}'(\underline{y})$ and $\underline{x}_0(y)$ are k-sparse, their difference is $2k$-sparse. The left hand inequality of the RIP$(2k, \delta)$ condition states $(1 - \delta)\|\underline{x}'(\underline{y}) - \widehat{\underline{x}}_0(y)\|_2 \leq \|A\underline{x}'(\underline{y}) - A\widehat{\underline{x}}_0(y)\|_2 = \|\underline{y} - \underline{y}\|_2 = 0$, which of course implies $\underline{x}'(\underline{y}) = \widehat{\underline{x}}_0(y)$.

Solving the optimization problem (1.6) essentially requires an exhaustive search over $\binom{n}{k}$ possible supports of the sparse vectors, which is intractable in practice. One avenue for simplifying this problem is to replace the "ℓ_0 norm" in (1.6) with the ℓ_1 norm. In other words we solve the *convex* optimization problem,

$$\widehat{\underline{x}}_1(y) = \text{argmin}_{\underline{x}: A\underline{x} = y}\|\underline{x}\|_1. \tag{1.7}$$

"norm") but it is common to use the notation $\|\cdot\|_0$ for reasons that will become clear shortly. It is also common to abuse language and call it a "ℓ_0-norm".

Note that this problem is convex since the function $\|\underline{x}\|_1$ is convex on $\mathbb{R}^n$ and the domain $\{\underline{x} : A\underline{x} = y\}$ is convex also. The following is one of a series of fundamental results.

THEOREM 1.1 (Candès and Tao 2006) *If A satisfies the* RIP$(2k, \delta_{2k})$ *condition with $\delta_{2k} < \sqrt{2} - 1$, then the solution of* (1.7) *is unique and identical to* (1.6).

This important result says that, for suitable measurement matrices, instead of solving the ℓ_0 problem, it suffices to solve the ℓ_1 problem which is a convex optimization problem, instead of the combinatorial search ℓ_0 problem. There is a prize we pay for going from ℓ_0 to ℓ_1, at least in terms of what the theorem guarantees. For the ℓ_0 problem we only need $\delta_{2k} < 1$, whereas for the ℓ_1 problem we need $\delta_{2k} \leq \sqrt{2} - 1$.

We will not prove Theorem 1.1 here but only offer some intuition through a simple toy example. Suppose that $n = 3$, so $\underline{x} = (x_1, x_2, x_3)^T$, and that we perform a single measurement $y = a_1x_1 + a_2x_2 + a_3x_3$. This equation corresponds

Figure 1.4 The ℓ_p balls

to the plane in figure 1.4. We seek to find a point on this plane, which minimizes $(x_1^p + x_2^p + x_3^p)^{1/p}$, $p \geq 0$. The case $p = 0$ is to be understood as the number of non-zero components of (x_1, x_2, x_3). As shown in figure 1.4 the solution is found by "inflating" the "ℓ_p-balls" around the origin until the plane is touched. It is clear that for a plane in generic position the solution is the same for all $0 \leq p \leq 1$. In particular it is the same for $p = 0$ and $p = 1$. Note also that for $0 \leq p \leq 1$ the solution only has a single non-zero component, so is "sparse". In contrast, for $p > 1$ the solution changes with p and all components are non-zero.

Note when $p = 1$ there are non-generic measurement matrices corresponding to planes parallel to the faces of the ℓ_1-ball for which the solution is not unique; it is therefore clear that the matrix should satisfy some conditions that excludes these non-generic cases.

But what matrices satisfy the RIP condition (1.5)? It should come as no surprise that a matrix satisfying the RIP condition should have a number of rows m at least as large as k. In fact one can show that necessarily $m \geq C_\delta k \log \frac{n}{k}$ for a suitable constant $C_\delta > 0$. It is not easy to make deterministic constructions of "good" measurement matrices approaching such bounds. The RIP condition is not the only possible condition that allows to replace the ℓ_0 by the ℓ_1 norm, but again they are not easily handled.

However the toy example naively suggests that in fact all we migth need are *random* measurement matrices. This is indeed a fruitful idea, at least in the asymptotic setting $n, m \to +\infty$ with $\kappa = \frac{k}{n}, \mu = \frac{m}{n}$ fixed. This is the route we will follow in the sequel, very much in the spirit of random coding constructions.

Ensembles of Measurement Matrices

The $m \times n$ matrix A will be taken from *the Gaussian ensemble* where the matrix entries are independent identically distributed Gaussian variables of zero mean and variance $1/m$. As in coding we will consider the asymptotic regime of a large system size. This corresponds to $n, m, k \to +\infty$ with *sparsity parameter* $\kappa = \frac{k}{n}$ and *measurement fraction* $\mu = \frac{m}{n}$ fixed. Note that each *row* of A has an expected ℓ_2 norm $O(1)$. This amounts to say that the average energy consumed per measurement is normalized to $O(1)$.[3]

One can show that there exists positive numerical constants c_1, c_2 such that for $m \geq c_1 \delta^{-2} k \log(\frac{en}{k})$ matrices from the Gaussian ensemble satisfy the RIP condition with overwhelming probability $1 - \exp(-c_2 \delta^2 m)$. We therefore do not attempt to construct specific matrices but are content with typical random Gaussian matrices. More general ensembles are also possible.

We also extend the ensemble formulation to the signal model. The simplest signal distributions assume that the components x_i are independently identically distributed according to a law of the form

$$p_0(x) = (1 - \kappa)\delta(x) + \kappa \phi_0(x), \qquad x \in \mathbb{R} \tag{1.8}$$

where $\phi_0(x)$ is a continuous probability density. Depending on the model or the application $\phi_0(x)$ is known or unknown. The most realistic assumption for applications is to consider that $\phi_0(x)$ is unknown, and in that case we call $\mathcal{S}_\kappa$ this class of sparse signals.

Noisy measurements and LASSO

A somewhat more realistic version of the measurement model takes noise into account,

$$\underline{y} = A\underline{x} + \underline{z}.$$

[3] It is perhaps more natural to choose a scaling $1/n$ for the variance of A but the present equivalent choice makes our life simpler in Chapter 8.

Here $\underline{z}$ is a noise vector typically assumed to consist of m identical independently distributed zero-mean Gaussian random variables with variance of σ^2. Because measurements have unit average power, the "signal to noise ratio" of the measurement model is $1/\sigma^2$.

Again our aim is to reconstruct a k-sparse signal with as few measurements as possible. The matrix A is chosen from the random Gaussian ensemble and the signal from the class $\mathcal{S}_\kappa$.

If we ignored the sparsity constraint then it would be natural to pick an estimate $\widehat{\underline{x}}(\underline{y})$ which solves the least-squares problem $\min_{\underline{x}} \|A\underline{x} - \underline{y}\|_2^2$. Solutions are well known,

$$\widehat{\underline{x}}(\underline{y}) = A^+\underline{y} + (I - A^+A)\underline{u}, \qquad \underline{u} \in \mathbb{R}^m,$$

where A^+ is the Moore-Penrose pseudo inverse,[4] but in general these solutions are not k-sparse.

To enforce the sparsity constraint, we can add a second term to our objective function, and attempt to solve the following minimization problem,

$$\widehat{\underline{x}}_0(\underline{y}) = \text{argmin}_{\underline{x}}(\|A\underline{x} - \underline{y}\|_2^2 + \lambda\|\underline{x}\|_0), \tag{1.9}$$

for a properly tuned parameter λ. Unfortunately this minimization problem is intractable, again because it requires an exhaustive search over the $\binom{n}{k}$ possible supports of the sparse vectors.

We saw in the noiseless case that replacing the "ℓ_0 norm" by the ℓ_1 norm is a fruitful idea. We follow the same route here and consider the following minimization problem

$$\widehat{\underline{x}}_1(\underline{y}) = \text{argmin}_{\underline{x}}(\|A\underline{x} - \underline{y}\|_2^2 + \lambda\|\underline{x}\|_1). \tag{1.10}$$

This estimator is called the *Least Absolute Shrinkage and Selection Operator* (LASSO). Again λ has to be chosen appropriately. This estimator can in principle be calculated by standard convex optimizaton techniques, which is already a big improvement over exhaustive search.

Although the LASSO estimator is popular, its a priori justification is not so straightforward. Our discussion suggests that in the noiseless limit it reduces to the pure ℓ_1 estimator which we know gives for a certain range of parameters the correct solution of the ℓ_0 problem. This is one possible justification. In Chapter 3 we also discuss a more or less "Bayesian justification" of the LASSO in a setting where the signal distribution is not known, but only the parameter κ is assumed to be known. Interestingly, the analysis of the LASSO in Chapter 8 yields an exact region for the ℓ_0-ℓ_1 equivalence in the (κ, μ) plane. The frontier of this region is known as the Donoho-Tanner curve who originally derived by

[4] In the standard setting of linear estimation where $m \geq n$ and if A has linearly independent columns then A^TA is invertible and $A^+ = (A^TA)^{-1}A$. The solution of least squares $\widehat{\underline{x}}(\underline{y}) = (A^TA)^{-1}A\underline{y}$ is then unique. See the exercises for the general definition of the Moore-Penrose pseudoinverse, which *always* exists.

methods of combinatorial geometry of polytopes. All this is ample justification for studying the LASSO in detail.

Graphical representation

Analogously to coding one can set up a graphical representation for the measurement matrix. We associate to A a bipartite graph G with vertices $V \cup C$, where $V = \{x_1, \ldots, x_n\}$ is the set of *variable* nodes corresponding to the n signal components and $C = \{c_1, \ldots, c_m\}$ is the set of *measurement* nodes each node corresponding to a row (a measurement) of A. There is an edge between x_i and c_j if an only if $A_{ji} \neq 0$. For the random measurement matrices discussed above this will essentially always be the case and therefore the graph is simply the *complete bipartite* graph depicted in figure 1.5.

FIGURE

Figure 1.5 The factor graph corresponding to a random Gaussian 2×4 measurement matrix.

We could attribute a "random weight" to the edges, but we will seldom need to do so. Therefore, unlike coding, here the graph is always the same. At this point the graphical construction may seem slightly trivial and arbitrary, but it will turn out to be a very useful way of thinking. The reason is that, much as in coding theory, we will develop iterative algorithms exchanging messages along the edges in order to reconstruct the signal.

Questions

We will consider the asymptotic regime where the total number of signal components n tends to infinity while the fractions of non-zero components $\kappa = k/n$ and of measurements $\mu = m/n$ are kept constant.

For given sparsity κ, we want to determine the smallest fraction μ of measurements so that with high probability we can recover $\underline{x}^{in}$ from the measurements $\underline{y}$, assuming we have no limitations on the computational complexity. This sets a theoretical limit on the minimal amount of measurements.

We then want to answer the same question given that we restrict ourselves to *low-complexity* algorithms.

Finally, we would like to design, if possible, compressive sensing schemes

which achieve the theoretical limits on the fraction of measurements under low-complexity algorithms.

1.3 Satisfiability

SAT problem

Suppose that we are given a set of n Boolean variables $\{x_1, \ldots, x_n\}$. Each variable x_i can take on the values 0 and 1, where 0 means "false" and 1 means "true". We define a *literal* to be either a variable x_i or its negation $\bar{x}_i$. A *clause* is a disjunction of literals, e.g.,

$$c = x_1 \vee x_2 \vee \bar{x}_3$$

where the operation "$\vee$" denotes the Boolean "or" operation. An *assignment* is an assignment of values to the Boolean variables, e.g., $x_1 = 0$, $x_2 = 1$, and $x_3 = 0$. Such an assignment will either make a clause to be *satisfied* or *not satisfied*. For example the clause $x_1 \vee x_2 \vee \bar{x}_3$ with assignment $x_1 = 0$, $x_2 = 1$, and $x_3 = 0$ evaluates to 1, i.e., the clause is satisfied. A SAT formula, call it F, is a conjunction of a set of clauses. For example, consider the SAT formula

$$F = (x_1 \vee x_2 \vee \bar{x}_3) \wedge (x_2 \vee \bar{x}_4) \wedge x_3.$$

where "$\wedge$" is the Boolean "and" operation.

The basic SAT problem is defined as follows. Given a SAT formula F, determine the satisfiability of F, i.e., determine if there exists an assignment on $\{x_1, \ldots, x_n\}$ so that F is satisfied. This is the SAT *decision* problem. If such an assignment exists we might also want to find an explicit solution.

Why would anyone be interested in studying this question? Perhaps surprisingly, many real-world problems map naturally into a SAT problem. For example designing circuits, optimizing compilers, verifying programs, or scheduling can be phrased in this way. The bad news is that Cook proved in 1973 that it is unlikely that there exists an algorithm which solves all instances of this problem in polynomial time (in n). More precisely, the SAT decision problem is NP-complete.

We say that a formula F is a K-SAT formula if every clause involves exactly K literals. E.g., $(x_1 \vee x_2 \vee \bar{x}_3) \wedge (x_2 \vee x_3 \vee \bar{x}_4)$ is a 3-SAT formula. The following facts are known. The 2-SAT decision problem is easily solved in a polynomial number of steps. Problem 1.6 discusses a simple algorithm called unit-clause propagation which solves a 2-SAT decision problem in at most $2n$ steps and produces a satisfying assignment if one exists. On the other hand for $K \geq 3$ the K-SAT decision problem is NP-complete.

Graphical representation of SAT formulas

Given a SAT formula F, we associate to it a bipartite graph G. The vertices of the graph are $V \cup C$, where $V = \{x_1, \ldots, x_n\}$ are the Boolean variables and

$C = \{c_1, \ldots, c_m\}$ are the m clauses. There is an edge between x_i and c_j if and only if x_i or $\bar{x}_i$ is contained in the clause c_j. Further we draw a "solid line" if c_j contains x_i and a "dashed line" if c_j contains $\bar{x}_i$.

EXAMPLE 2 (Factor Graph of SAT Formula) As an example, the graphical representation of $F = (x_1 \vee x_2 \vee \bar{x}_3) \wedge (x_2 \vee x_3 \vee \bar{x}_4)$ is shown in Fig. 1.6. □

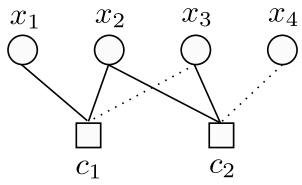


Figure 1.6 The factor graph corresponding to the SAT formula of Example 2.

Ensemble of random K-SAT Formulas

Just like in the coding and compressed sensing problems, rather than looking at individual SAT formulas, we will define an *ensemble* of such formulas and we will then study the probability that a formula from this ensemble is satisfiable. In particular, we will stick to the behaviour of random K-SAT formulas.

The ensemble $\mathcal{F}(n, m, K)$ is characterized by 3 parameters: K is the number of literals per clause, n is the number of Boolean variables, and m is the number of clauses. Notice that with K variables we can form $\binom{n}{K}2^K$ clauses by taking K variables among $x_1, \ldots, x_n$ and then negating them or not. We define $\mathcal{F}(n, m, K)$ by showing how to sample from it. To this end, pick m clauses $c_1, \ldots, c_m$ independently, where each clause is chosen uniformly at random[5] from the $\binom{n}{K}2^K$ possible clauses. Then form F as the conjunction of these m clauses. In other words, the ensemble $\mathcal{F}(n, m, K)$ is the uniform probability distribution over the set of all possible formulas F constructed out of n Boolean variables by choosing m clauses.

Threshold behavior

Now let us consider the following experiment. Fix $K \geq 2$ (e.g., $K = 3$) and draw a formula F from the $\mathcal{F}(n, m, K)$ ensemble. Is such a formula satisfiable with high probability? It turns out that the most important parameter that affects the answer is $\alpha = \frac{m}{n}$. This ratio is called the *clause density*. Like in coding and compressed sensing we are interested in the asymptotic regime where $n, m \to +\infty$ and α is fixed.

Fig. 1.7 shows the probability of satisfiability of F as a function of both n and

[5] Choosing a clause with or without replacement yields two different enesembles which are for all pratical purposes equivalent in the large size limit

α. As we see from this figure, as n becomes larger the transition of the probability of satisfiability becomes sharper and sharper. This is a strong indication that there exists a *sharp* threshold behavior, that is, there exists a real number $\alpha_s(K)$ such that

$$\lim_{n\to\infty} \mathbb{P}\left[F \text{ is satisfied}\right] = \begin{cases} 1, & \alpha < \alpha_s(K), \\ 0, & \alpha > \alpha_s(K). \end{cases} \tag{1.11}$$

Here $\mathbb{P}\left[-\right]$ is the uniform probability distribution of the ensemble $\mathcal{F}(n, m, K)$ with $\frac{m}{n} = \alpha$ fixed.

As the density α increases one has more and more clauses to satisfy, so it intuitively quite clear that the probability of satisfaction decreases as a function of α. However the existence of a sharp threshold is much less evident, let alone its computation. Such a threshold behavior was conjectured based on experiments in 1992. For many years this was proved only for $K = 2$ for which $\alpha_s(2) = 1$. For $K \geq 3$ Friedgut proved that there exists a sequence $\alpha_s(K, n)$, $n \in \mathbb{N}$, such that for all $\epsilon > 0$

$$\lim_{n\to\infty} \mathbb{P}\left[F \text{ is satisfied}\right] = \begin{cases} 1, & \alpha < (1-\epsilon)\alpha_s(K, n), \\ 0, & \alpha > (1+\epsilon)\alpha_s(K, n). \end{cases} \tag{1.12}$$

This result leaves open the possibility that the sequence of thresholds $\alpha_s(K, n)$ does not converge to a definite value as $n \to +\infty$. Even though this result does not settle the story completely, it is of considerable importance if we want to find bounds on the threshold. E.g., suppose that we have an algorithm that allows to find solutions of random K-SAT formulas with uniformly positive probability (uniformly with respect to the size n of the formulas) for some range of densities, say $\alpha < \alpha_{\text{alg}}(K)$. Then invoking the threshold behavior (1.12) guaranteed by Friedgut's theorem we conclude that the formula is almost surely satisfiable for $\alpha < \alpha_{\text{alg}}(K)$.

The proof of a sharp threshold behavior (1.11) was proved recently for K large enough (very large but finite), but for small K's (except $K = 2$) a proof is still a challenging problem and might require a new set of ideas.

The underpinnings of this proof for large K's rest on the statistical mechanics methods which also give the means to compute $\alpha_s(K)$ (for example it is known that $\alpha_s(3) \approx 4.259$ to three decimal places). As we will see these methods yield much more information than just the threshold value. We will uncover various other threshold behaviors, related not only to the satisfiability of random formulas, but also to the "nature" and "organisation" of the solution space. Understanding the nature of these threshold behaviors in K-SAT is an order of magnitude more difficult than in coding theory and compressed sensing.

Random max-K-SAT

In the K-SAT decision problem, one is given a formula and is asked to determine if this formula is satisfiable or not. An important variation on this theme is

the *max-K*-SAT problem. In this problem one is interested in determining the *maximum* possible number of *satisfied* clauses where the maximum is taken over all possible 2^n assignments of variables $x_1, \ldots, x_n \in \{0,1\}^n$. Of course it is equivalent to determine the *minimum* possible number of *violated* clauses where the minimum is taken over all assignments of variables. We adopt this perspective in the sequel because it makes the contact with traditional statistical mechanics questions clearer.

We will be interested in the random version of max-K-SAT which we know formulate more precisely. Take a formula at random from the ensemble $\mathcal{F}(n, m, K)$. This formula contains m clauses labelled $c_1, \ldots, c_m$. If we let $\mathbb{1}_c(\underline{x})$ be the indicator function over assignments that satisfy clause c (the function evaluates to 1 if $\underline{x}$ satisfies c and 0 if $\underline{x}$ does not satisfy c) then the maximum possible number of satisfied clauses is

$$\max_{\underline{x}\in\{0,1\}^n} \sum_{i=1}^{m} \mathbb{1}_{c_i}(\underline{x}) \tag{1.13}$$

In the random max-K-SAT problem we want to compute

$$\lim_{m\to+\infty} \frac{1}{m} \mathbb{E}\Big[\max_{\underline{x}\in\{0,1\}^n} \sum_{i=1}^{m} \mathbb{1}_{c_i}(\underline{x})\Big] \tag{1.14}$$

where the expectation is taken over the ensemble $\mathcal{F}(n, m, K)$ (the existence of the limit has been proven by methods that we will study in Chapter 12). Equivalently we want to compute the average of the minimum possible number of violated clauses

$$e(\alpha) \equiv \lim_{m\to+\infty} \frac{1}{m} \mathbb{E}\Big[\min_{\underline{x}\in\{0,1\}^n} \sum_{i=1}^{m} (1 - \mathbb{1}_{c_i}(\underline{x}))\Big] \tag{1.15}$$

We define the *max-K-sat threshold* as

$$\alpha_{s,\max}(K) = \sup\{\alpha | e(\alpha) = 0\} \tag{1.16}$$

We will give a non-rigorous computation of (1.15) and (1.16) in chapters 16 and 17. In fact, the recent proof for the sharp threshold behavior (1.11) (for very large K) has its origin in such statistical mechanics computations.

Intuitively one expects that $\alpha_{s,\max}(K) = \alpha_s(K)$. It is an exercise to show that one must have $\alpha_s(K) \leq \alpha_{s,\max}(K)$. However the converse bound is not immediate because one could conceivably have a finite interval $]\alpha_s(K), \alpha_{s,\max}(K)[$ where $e(\alpha) = 0$ and at the same time a sublinear fraction of unsatisfied clauses. Nevertheless it is widely believed this does not happen and that $\alpha_s(K) = \alpha_{s,\max}(K)$. At least we know that this is true for $K = 2$ and for large enough K.

Questions

One would like to determine if the random K-SAT problem exhibits a threshold behaviour, and if so, determine this threshold α_s.

One wants to find low-complexity algorithms which are capable of finding satisfying assignments, assuming such assignments exist, and determine up to what clause density they work with high probability.

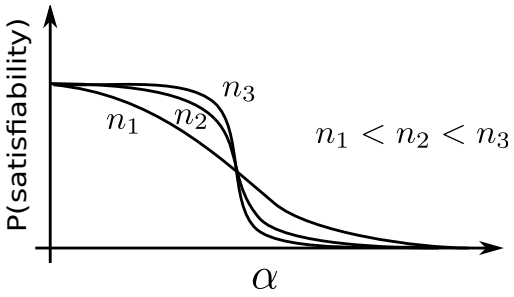


Figure 1.7 The probability that a formula generated from the random K-SAT ensemble is satisfied versus the clause density α.

Perhaps surprisingly, many of the above questions do not yet have a rigorous answer and the satisfiability problem is by far the hardest of our three examples. Nevertheless we will derive non-trivial statements about this problem and if one admits non-rigorous methods, the problem is fairly well understood.

1.4 Notes

The theory of error correcting codes is a very large subject which dates back to the 1940's. General references are (Berlekamp 1984, Blahut 2003, Lin & Costello 2004). Although Shannon's channel coding theorem used the concept of random codes, the first "practical" code constructions were deterministic and used algebraic tools. Random Low-Density Parity-Check codes were first proposed by Gallager (Gallager 1962, Gallager 1963) who also proposed efficient decoding algorithms and methods to analyse them. This theory did not find immediate applications due to limitation in computing power. A revival of Gallager's ideas occurred after the invention of Turbo Codes (Berrou, Glavieux & Thitimajshima 1993, Goff, Glavieux & Berrou 1994) and the rediscovery of LDPC codes (MacKay & Neal 1996, MacKay 1999). We have only presented the construction of regular Gallager ensembles. Such constructions are inspired by the configuration model of random graphs (Bollabás 1998). Ensembles of bipartite graphs with irregular degrees were introduced in (Luby, Mitzenmacher, Shokrollahi, Spielman & Stemann 1997, Luby, Mitzenmacher, Shokrollahi & Spielman 2001) and by choosing appropriate degree distribution one gets excellent codes with capacity close to Shannon's limit. The factor graph representations were introduced by (Tanner 1981) and have become ubiquitous.

The traditional basis of digital signal processing is the Whittaker, Nyquist, Kotelnikov, Shannon sampling theorem which states that band limited signals

can be reconstructed when sampled at a rate at least twice the bandwidth A short history of this theorem whose various aspects were discovered many times can be found in (Lukke 1999). While this theorem applies to any band limited signal, it was already realised long ago, notably in (Caratheodory 1907), that much fewer samples are needed when the number of Fourier coefficients is limited and the frequency range large. The use of the ℓ_1 norm for reconstructing "impulsive signals" from a limited number of observations goes back at least to (Beurling 1938). These and other works predated the modern compressive sensing paradigm that emerged from a series of important recent papers (e.g Candès 2006*a*, Candès 2006*b*, Candès, Romberg & Tao 2006, Candès & Tao 2006, Donoho 2006). These works introduced the "restricted isometry property" and proved fundamental theorems in the spirit of Theorem 1.1 in noiseless as well as noisy settings. Variants of the restricted isometry property as well as constructions of suitable measurement matrices have been intensively explored. We refer to the reviews in the book (Eldar & Kutyniok 2012) for further information. The LASSO dates back to the 1990's and was first introduced in the context of linear regression models (Tibshirani 1996, Chen & Donoho 1995). It is also referred to as "Basis Pursuit Denoising".

The satisfiability problem is one of the most important paradigms of complexity theory (Garey & Johnson 1979, Papadimitriou & Steiglitz 1982). The 3-SAT problem was the first of a long list which was shown to be NP complete (Cook 1971). Traditionally computer scientists have been interested in the worst case analysis of this and similar problems. Interest in the random version of such problems is more recent and was largely motivated by the "experimental" discovery that typical random K-SAT formulas exhibit a phase transition (Mitchell, Selman & Levesque 1992). A threshold behaviour was first proven in the important work by (Friedgut 1999), and the existence of a sharp threshold separating a satisfiable from an unsatisfiable phase in the limit of an infinite number of variables has been proved for large enough K (Ding, Sly & Sun 2014). An excellent reference discussing various aspects of computational complexity is (Moore & Mertens 2011).

As stated in the introduction, the connections between statistical physics, information theory and computer science have been recognised early on. Already one decade after Shannon laid the foundations of information theory (Shannon 1948), Jaynes discussed connections between this theory and some of the basic principles of statistical mechanics (Jaynes 1957), and Brillouin wrote a classic book applying information theory concepts to a wide array of physical problems, e.g., in thermodynamics, measurements and the physical limits of observations (Brillouin 1956). Ever since there have been numerous fundamental investigations on the physical aspects of computation and information. One line of thought led to the concept of a "quantum computer", see (Loyd 2000) for a review. Here we are concerned with a completely different thread that originated in the 1980's, namely analogies between "combinatorial optimisation problems" and "spin glasses". Such connections were explicitly put for-

ward in (Hopfield 1982) and (Fu & Anderson 1986) and a set of early references can be found in (Mézard, Parisi & Virasoro 1987*a*). This school of thought has flourished since then in many areas (e.g, neural networks) and since the late 1990's some of the most successful developments have occurred in communications, signal processing and random constraint satisfaction problems (e.g Nishimori 2001, MacKay 2003, Mézard & Montanari 2009).

Problems

1.1 Shannon Capacity for Binary Input Symmetric Memoryless Channels. Derive the capacities of the BSC and BAWGNC from Shannon's general formula (1.2).

A binary input memoryless channel is said to be *symmetric* if one can find a bijection $\pi : \mathcal{Y} \to \mathcal{Y}$ acting on the output alphabet such that $p(y|1) = p(\pi(y)|0)$. For the BSC we have $\pi(0,1) = (1,0)$, for the BEC $\pi(0,E,1) = (1,E,0)$, and for the BAWGNC $\pi(y) = -y$. So these three channels are symmetric.

Show that for general symmetric channels Shannon's capacity formula can be written as

$$C_{\text{channel}} = 1 - \sum_{y\in\mathcal{Y}} p(y|0)\log_2\left(1+\frac{p(y|1)}{p(y|0)}\right)$$

If the output alphabet is $\mathbb{R}$ (e.g., the BAWGNC) the sum is interpreted as an integral. Check the special cases of the BEC, BSC, BAWGNC.

1.2 Configuration Model. The aim of this problem is to write a program that can sample a random graph from the configuration model introduced in section 1.1. Your program should take as input the parameters n, m, d_v, and d_c, it should then check that the input is valid, and finally return a bipartite graph according to the configuration model. Think about the data structure. If we run algorithms on such a graph it is necessary to loop over all nodes, refer to edges of each node, be able to address the neighbour of a node via a particular edge and store values associated to nodes and edges.

1.3 Norms and pseudo-norms. Let $\|\underline{x}\|_p = (\sum_{i=1}^n |x_i|^p)^{1/p}$ for $p > 0$. Let also $\|\underline{x}\|_0 = \sharp(\text{non zero } x_1,\dots,x_n)$ and $\|\underline{x}\|_\infty = \max_i |x_i|$. Show first that $\|\underline{x}\|_0 = \lim_{p\to 0}\|\underline{x}\|_p$ and $\|\underline{x}\|_\infty = \lim_{p\to+\infty}\|\underline{x}\|_p$. Explain why $\|\cdot\|_p$ is a norm for $1 \le p \le +\infty$ and is *not* a norm for $0 \le p < 1$ (this is why for $0 \le p < 1$ we call it a pseudo-norm). *Hint*: refer to the figure 1.4.

1.4 Least square estimator. Show that the minimizer of $\|\underline{y}-A\underline{x}\|_2^2+\alpha\|\underline{x}\|^2$ where A is a real matrix and $\alpha > 0$ fixed, is equal to $\widehat{\underline{x}}^{(\alpha)}(\underline{y}) = (A^TA+\alpha)^{-1}A^T\underline{y}$ and is unique. When the matrix A has linearly independent columns (this is only possible if $m \ge n$) then A^TA is invertible and $\lim_{\alpha\to 0}(A^TA+\alpha)^{-1} = (A^TA)^{-1}$. Thus when A has full column rank the minimizer of $\|\underline{y}-A\underline{x}\|_2^2$ is the least square estimator $\widehat{\underline{x}}^{\text{LS}}(\underline{y}) = (A^TA)^{-1}A^T\underline{y}$.

It is possible to prove, and we will admit it here, that $\lim_{\alpha\to 0}(A^TA+\alpha)^{-1}A^T \equiv A^+$ always exists for any real matrix A. It is also possible to show that this

is the unique matrix which satisfies the four conditions: (i) $AA^+A = A$, (ii) $A^+AA^+ = A^+$, (iii) $(AA^+)^T = AA^+$, (iv) $(A^+A)^T = A^+A$. The matrix A^+ is called the *Moore-Penrose pseudoinverse* of A.

Prove that the Moore-Penrose pseudoinverse always solves the least square problem in the sense that

$$\|\underline{y} - A\underline{x}\|_2^2 \geq \|\underline{y} - A\underline{\widehat{x}}^{\mathrm{LS}}(\underline{y})\|_2^2, \qquad \underline{\widehat{x}}^{\mathrm{LS}}(\underline{y}) = A^+\underline{y}.$$

Note that equality is satisfied for $\underline{x} = A^+\underline{y} + (I - A^+A)\underline{u}$ for any $\underline{u}$. Thus in general the least square minimiser is not unique. Show that if A is has full column rank the minimizer is unique and equal to $\underline{\widehat{x}}^{\mathrm{LS}}(\underline{y}) = (A^TA)^{-1}A^T\underline{y}$.

1.5 Poisson Model. An important model of bipartite random graphs is the *Poisson model*. For example the random K-SAT problem is often formulated on this graph ensemble. Pick two integers, n and m. As before, there are n variable nodes and m check nodes. Further, let K be the degree of a check node. For each check node pick K variables uniformly at random either with or without repetition and connect this check node to these variable nodes. For each edge store in addition a binary value chosen according to a Bernoulli(1/2) random variable.

This is called the Poisson model because the node degree distribution on the variable nodes converges to a Poisson distribution for large n. This is also the case for the formulation in Section 1.3 which is equivalent in the asymptotic limit.

Write a program that takes n, m, K as input parameters and outputs a graph instance from the Poisson model. Again, think of the data structure.

1.6 Unit Clause Propagation for Random 3-SAT Instances. The aim of this problem is to test a simple algorithm for soving SAT instances. Generate random instances of the Poisson model. Pick $n = 10^5$ and let $K = 3$. Let α be a non-negative real number. It will be somewhere in the range $[0, 5]$. Let $m = \lfloor \alpha n \rfloor$ the largest integer smaller than αn. For a given α generate many random bipartite graphs according to the Poisson model. Interpret such bipartite graphs as random instances of a 3-SAT problem. This means, the variables nodes are the Boolean variables and the check nodes represent each a clause involving 3 variables. Associate to each edge a Boolean variable indicating whether in this clause we have the variable itself or its negation.

For each instance you generate, try to find a satisfying assignment in the following greedy manner. This is called the *unit clause propagation* algorithm:

(i) If there is a check node of degree one in the graph (this corresponds to a *unit*-clause), then choose one among such check nodes uniformly at random. Set the variable to satisfy it. Remove the clause from the graph together with the connected variable and remove or shorten other clauses connected to this variable (if the variable satisfies other clauses they are removed while if not they are shortened).

(ii) If no such check exists, pick a variable node uniformly at random from the graph and set the corresponding litteral to 0 or 1 uniformly at random. Remove this variable node from the graph. For each constraint node connected by an edge to this litteral do the following. If the clause is satisfied then remove the edge and the clause (or constraint). If not, then remove only the edge.

Continue the above procedure until there are no variable nodes left. If, at the end of the procedure, there are no check nodes left in the graph (by definition all variable nodes are gone) then we have found a satisfying assignment and we declare success. If not, then the algorithm failed (although the instance itself might very well be satisfiable).

Plot the empirical probability of success for this algorithm as a function of α. Roughly at what value of α does the probability of success becomes close to 0?

2 Basic Notions of Statistical Mechanics

There is a special class of probability distributions called *Gibbs distributions* which plays a prominent role in the analysis of our models. We show in the next chapter how Gibbs distributions arise quite naturally in the context of coding, compressed sensing and satisfiability. However, many insights and useful analogies can be gained by understanding why Gibbs distributions play a prominent role in the description macroscopic *physical* systems. It is the goal of this chapter to expound on the second point. This also gives us the opportunity to introduce some of the language and standard notions and settings of statistical mechanics.

Statistical mechanics describes the *macroscopic* (large scale) behavior of systems that are composed of a very large number of "elementary" degrees of freedom. For example condensed matter systems have a huge[1] number of atoms, molecules, magnetic moments or spins, etc. Similarly, we are interested in the behavior of our models when the number of transmitted bits, of signal components or literals is very large.[2]

In macroscopic physical systems a precise knowledge and description of the microscopic dynamics of each degree of freedom, say by solving Newton's differential equations for the positions and velocities of all molecules, is just impossible. Fortunately this is usually not required in order to derive macroscopic properties of the system. The general approach of statistical mechanics is to replace the microscopic dynamical description by a statistical one, based on appropriate probability distributions. A *universal* probabilistic description - given by Gibbs distributions - is known for systems that have reached the state of "thermodynamic equilibrium". It is not easy to precisely define what thermodynamic equilibrium is; it is enough to think of it as a state of matter where the temperature, pressure and chemical potential are homogeneous so that heat currents, mechanical stresses, and particle currents are all absent. It turns out that the precise nature of the underlying microscopic dynamics is largely irrelevant, e.g., whether it is deterministic or random, except for the existence of quantities that are conserved under the dynamics (a typical example of a conserved quantity is the energy of the system). In fact even the existence of a dynamics is not needed, or at least it is not explicitly needed. This is noteworthy because in our models

[1] For an order of magnitude, $1\,\mathrm{cm}^3$ of helium at normal conditions of $1\,\mathrm{atm}$ and $0^o\,C$ contains 2.7×10^{19} atoms.

[2] Here "large" depends on the the technology used. For example ...

no natural dynamics is a priori given, and if for some reason we choose one, this choice is not unique.

We warn the reader that Gibbs distributions do not describe systems that are not in thermal equilibrium; such systems are said to be "out of equilibrium" and their fundamental probabilistic descriptions, assuming they exist, are not yet elucidated. Such systems range from the simplest stationary heat or electric flows all the way to living systems.

Thermodynamic equilibrium can be characterized as a state of "maximal disorder" compatible with whatever "conserved quantities" are relevant. This gives us a clue into the nature of the Gibbs distributions: these are the distributions that maximize an entropy functional (in fact Shannon's entropy) under the constraints provided by the conserved quantities. The notion of conserved quantity might not be familiar to the reader, but this should not be a problem because the most important one (and the one that is relevant to us) is the energy or *Hamiltonian* of the system. One can just think of this quantity as some sort of *cost function*. We already encountered one such cost function in the satisfiablity problem, namely the minimum possible number of violated clauses. In compressed sensing the mean square errors (penalized by the ℓ_0 or ℓ_1 norms) are also cost functions.

To lay the foundations on a concrete footing we will first describe "toy" models of statistical mechanics, which as it turns out, belong to its most fundamental paradigms. Based on these models we illustrate a simple derivation of Gibbs distributions from a *maximum entropy principle*. We then define standard notions of free energy, marginals, correlation functions, thermodynamic limit and provide a first introduction to the concept of phase transition. There is no unique way to introduce Gibbs distributions and the main body of this chapter goes along a short path. But one should note this path uses the notion of Shannon entropy which itself is not an obvious primary object for physical systems. The founding fathers of statistical mechanics deduced Gibbs distributions from more primary principles. The interested reader will find a derivation along such lines in the last section; the impatient reader can skip it without harm.

2.1 Lattice gas and Ising models

The *lattice gas* and *Ising models* are very simple to formulate and have taught us surprisingly much about statistical mechanics; their importance cannot be understated. There is an immense body of theory that is known about such models which we completely omit here (some of it is briefly reviewed in Chapter 4, Section 4.9). These models will serve us well to get a rapid and concrete derivation of the Gibbs distribution. This section introduces their Hamiltonians or cost functions, first in the traditional language of statistical mechanics, and then with a "factor graph" representation.

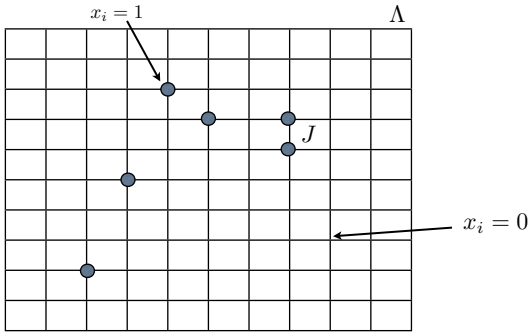


Figure 2.1 Left: a particle configuration in the lattice gas model. Full circles represent occupied sites $x_i = 1$ and empty circles unoccupied sites $x_i = 0$. There can be at most one particle per lattice site. Right: a magnetic configuration of the Ising model. Positive signs indicate "up spins" $s_i = +1$ and negative signs "down spins" $s_i = -1$.

Lattice gas model

Consider a discrete d-dimensional grid (see Figure 2.1; naturally, $d = 3$ is an important case but other values of d are of also of great relevance both theoretically and practically). Particles occupy the vertices of this grid with at most one particle occupying any single vertex. We call $V = \{x_1, \ldots, x_n\}$ the set of vertices and E the set of edges formed by pairs of nearest neighbord vertices. The configuration of the system is described by a vector $\underline{x} = (x_1, \cdots, x_n)$ where $x_i = 1$ if an atom is present at vertex i and $x_i = 0$ if vertex i is empty. Let us introduce a cost function usually called the *Hamiltonian*. Physically this function gives the energy cost associated to a configuration $\underline{x}$. We define

$$\mathcal{H}(\underline{x}) = -\sum_{\{i,j\}\in E} J_{ij} x_i x_j - \sum_{i\in V} \mu_i x_i. \tag{2.1}$$

where J_{ij} and μ_i are in $\mathbb{R}$. Each edge $\{i, j\}$ is counted once in the sum (so $\{i, j\} = \{j, i\}$). Only neighboring atoms "interact" and their "interaction energy" is $-J_{ij}$ (because when vertices i and j are occupied $x_i = x_j = 1$, and their intercation energy is $-J_{ij}x_i x_j = -J_{ij}$).

In the canonical model $J_{ij} = J$ and $\mu_i = \mu$ are constant, with $J < 0$ corresponding to a repulsive interaction and $J > 0$ to an attractive interaction between neighboring atoms. The real number μ is an energy cost associated to the presence or absence of a particle. For example in two dimensions the grid may model the surface of some material which absorbs some vapour and one

may think of μ as a binding energy between the atoms of the vapour and the surface. This kind of energy is called a "chemical potential".

Canonical Ising model

The canonical Ising model introduced by Lenz and Ising in 1928, is one of the oldest and best studied models of statistical mechanics. We will refer to it in many occasions. In this model the degrees of freedom describe "magnetic moments" localized at the sites of a crystal. For our case these sites are the vertices of a d-diemensional grid with vertex set V and edge set E. In the Ising model we retain only "up or down directions" for the magnetic moments. These are modeled by *Ising spins* which are binary variables $s_i = \pm 1$, $i \in V$. The Hamiltonian is

$$\mathcal{H}(\underline{s}) = -\sum_{\{i,j\}\in E} J_{ij} s_i s_j - \sum_{i\in V} h_i s_i. \tag{2.2}$$

where $\underline{s} = (s_1, \ldots, s_n)$ and J_{ij}, h_i are in $\mathbb{R}$. In the canonical Ising model $J_{ij} = J$ and $h_i = h$ are constant throughout the lattice. For $J > 0$ (ferromagnetic interaction) neighboring spins lower their energy when they "align" in the same direction" ($s_i = s_j$) while for $J < 0$ (anti-ferromagnetic interaction) they lower their energy by "anti-aligning" ($s_i = -s_j$). Here h has the interpretation of a "magnetic field" applied on the system and biases the "direction" of the magnetic moments.

Mathematically speaking the lattice-gas and Ising models are equivalent. One can go from one to the other simply by performing the change of variable

$$x_i = \frac{1}{2}(1 - s_i), \qquad \text{or} \qquad s_i = 1 - 2x_i = (-1)^{x_i}$$

and redefining the various "interaction constants" J_{ij}, μ_i, h_i.

General Ising models

It is common to formulate the Ising model on general graphs $G = (V, E)$ with vertex set $V = \{1, \ldots, n\}$ and edge set $E \subset V \times V$. Motivations for such a generalisation are diverse. In statistical or condensed matter physics the graph may be a regular grid or lattice representing an underlying crystalline structure. It may also represent an approximation of continuous space in various dimensions. But there are also important applications of the model in other disciplines, e.g., image processing, neural networks, learning, social networks. For such applications the graphs do not necessarily have a spatial structure and may just be arbitrary. A general Ising model has the Hamiltonian (2.2) where now the vertex and edge sets refer to an arbitrary graph $G = (V, E)$.

EXAMPLE 3 The canonical *Ising model* has $V = \mathbb{Z}^d \cap [-\frac{L}{2}, \frac{L}{2}]^d$, where d is the "spatial dimension" and L is an odd integer. The box $[-\frac{L}{2}, \frac{L}{2}]^d$ is centered at the origin and encloses L^d vertices. Vertices $i \in V$ are vectors with integer

components and $\{i,j\} \in E$ consist of all nearest neighbord pairs, $|i-j|=1$. Further, $h_i = h$ for $i \in V$ and $J_{ij} = J$ for $\{i,j\} \in E$. The model is called ferromagnetic when $J > 0$ and anti-ferromagnetic when $J < 0$. □

EXAMPLE 4 In the *Curie-Weiss model* G is the complete graph on n vertices. There are $n(n-1)/2$ edges with interaction constant $J_{ij} = J/n$, $J > 0$. In addition the magnetic field is taken constant $h_i = h$. This an important "exactly solvable" model which we treat in detail in Chapter 4. □

EXAMPLE 5 In the *Ising model on a tree* G is a (finite) tree, in other words a graph without loops. An important "exactly solvable" consists The case of a regular tree of degree k (except for the leaf nodes which have degree one) and $J_{ij} = J > 0$, $h_i = h$ constitutes an important fully solvable model which we analyze in Chapter 4. □

General binary spin systems

So far all examples have involved "pairwise interactions" between spins s_i and s_j linked to by and $\{i,j\}$. We can consider more general models with "multi-spin interactions" (and this occurs all the time in coding and satisfiability for example). For example on a grid the four spins of elementary loops (called "plaquettes") may interact through a term

$$-\sum_{(i,j,k,l)\in P} J_{ijkl} s_i s_j s_k s_l$$

in the Hamiltonian, where P is the set of all elementary plaquettes of the grid and $J_{ijkl} \in \mathbb{R}$.

The *most general binary spin model* has a Hamiltonian of the form

$$\mathcal{H}(\underline{s}) = -\sum_{A \subset V} J_A \prod_{i \in A} s_i \tag{2.3}$$

where $J_A \in \mathbb{R}$ and the sum over $A \subset V = \{1,\dots,n\}$ carries over all possible subsets of V (the power set with $2^{|V|}$ elements). The most general lattice gas has a similar Hamiltonian. The pairwise Ising models correspond to the choice $J_A = h \neq 0$ for $A = \{i\}$, $i \in V$ and $J_A = J_{ij} \neq 0$ for all $A = \{i,j\} \in E$ the set of edges, and $J_A = 0$ otherwise. If we add a plaquette interaction we also have $J_A = J_{ijkl} \neq 0$ for all $A = \{i,j,k,l\} \in P$ the set of all plaquettes.

The factor graph representation is a convenient representation of general hamiltonians (2.3). Here the factor graph is a bipartite graph with "variable nodes" associated to spin variables $s_1,\dots,s_n$ (or lattice gas variables $x_1,\dots,x_n$) and "function nodes" associated to subsets $A \subset V$ with $J_A \neq 0$. The factor graphs associated to the Ising and lattice gas models on a grid, as well as the one with plaquette interactions added, are illustrated on Fig. 2.2. Note that the factor graph itself does not represent the underlying physical lattice but rather is a convenient summary of the various interactions present in the model. We will come back to this representation is more details in Chapter **??**.

FIGURE

Figure 2.2 Left: factor graph of the canonical Ising model. Right: factor graph of a spin system with pair and plaquette interactions.

In Chapter 3 it will become clear that the LDPC codes and K-SAT models have cost functions that are of the form 2.3. For compressed sensing the "spins" are real numbers and one talks about "continuous" or "scalar" spins.

2.2 Gibbs distribution from maximum entropy

The Gibbs distributions date back at least to the beginning of the 20th century, and presenting the historical derivations of Maxwell, Boltzmann, Gibbs, Einstein and others would lead us much too far (see the notes for references). In the decade following Shannon's 1948 paper, Jaynes in 1957 showed that one can derive Gibbs distributions from a "maximum entropy principle". This is the route we take here because it is economical and serves our purpose well.

Let $p(\underline{x})$ be a probability distribution which is supposed to describe the thermal equilibrium state of a macroscopic system with degrees of freedom $\underline{x} = (x_1, \ldots, x_n)$. Here we keep in mind the lattice gas, Ising or generalized spin systems for concreteness (with $|V| = n$), but it will soon be clear that the development here is very generic. The question is: how should we choose this probability distribution?

This probability distribution must describe typical configurations of the degrees of freedom. If the system were to be completely isolated from the rest of the universe then certainly its energy would be conserved. There could also be other relevant conserved quantities depending on the nature of the system but for our purposes we ignore these more general cases. In reality the system has reached thermal equilibrium through its interactions with the environment, so it is not isolated and the energy is not strictly conserved. However in thermal equilibrium there are no macroscopic fluxes of energy between the system and its environnement, and we assume that the *average energy* is fixed. Thus $p(\underline{x})$

should satisfy the constraint

$$\sum_{\underline{x}} p(\underline{x})\mathcal{H}(\underline{x}) = E \tag{2.4}$$

where E is the average total energy, and the sum carries over all possible configurations of degrees of freedom. Of course there remains energy fluctuations due to random exchanges between the system and the environnement but these are expected to be of order of the surface separating the system from the environnement, i.e., $O(n^{(d-1)/d})$.

The *maximum entropy principle* postulates that the state of thermal equilibrium[3] maximizes the entropy but still satisfies the constraint (2.4). For the entropy we take Shannon's functional

$$S(p(\cdot)) = -\sum_{\underline{x}} p(\underline{x}) \ln p(\underline{x}) \tag{2.5}$$

(here we use the letter S instead of H because the logarithm is Neperian). This "guess work" leads us to the following prescription: the distribution that describes the thermodynamic equilibrium state is the one that maximizes

$$S(p(\cdot)) - \beta \sum_{\underline{x}} p(\underline{x})\mathcal{H}(\underline{x}) \tag{2.6}$$

where β is a Lagrange multipier enforcing the constraint (2.4).

Shannon's entropy is a strictly concave functional of $p(\cdot)$ and the second term in (2.6) is linear, therefore the functional (2.6) is strictly concave and has a unique maximizer. To find it we must recall that $\sum_{\underline{x}} p(\underline{x}) = 1$, so we introduce one more Lagrange multplier γ, and maximize

$$S(p(\cdot)) - \beta \sum_{\underline{x}} p(\underline{x})\mathcal{H}(\underline{x}) + \gamma \sum_{\underline{x}} p(\underline{x})$$

Setting the derivative with respect to $p(\underline{x}')$ (for any fixed $\underline{x}'$) to zero we find

$$p(\underline{x}) = e^{\gamma - 1} e^{-\beta \mathcal{H}(\underline{x})}.$$

The constant γ is fixed by the normalization condition and we find for the maximizer of (2.6)

$$p_G(\underline{x}) = \frac{e^{-\beta \mathcal{H}(\underline{x})}}{Z} \tag{2.7}$$

where

$$Z = \sum_{\underline{x}} e^{-\beta \mathcal{H}(\underline{x})}. \tag{2.8}$$

The distribution (2.7) is called the *Gibbs distribution* and Z the *partition function* (or sometimes the sum over states).

[3] This is a state with no gradients of temperature, pressure, or chemical potential, on a macroscopic scale. As such, it is "maximally disordered" and "structureless".

What is the interpretation of the Lagrange multiplier β? For physical systems $\beta^{-1} = k_B T$ where T is the temperature of the system and k_B a constant - the Boltzmann constant - such that $k_B T$ has units of energy. Of course in coding, compressed sensing, or satisfiability, there is no physical temperature and the interpretation may vary according to the problem at hand.

Let us now justify the physical interpretation $\beta = 1/k_B T$ where T is the temperature of the system. In the process we also introduce important quantities. We define the *Gibbs entropy*

$$S(\beta) \equiv S(p_G(\cdot)) = -\sum_{\underline{x}} p_G(\underline{x}) \ln p_G(\underline{x}) \tag{2.9}$$

and the *internal energy*

$$\mathcal{E}(\beta) \equiv -\sum_{\underline{x}} p_G(\underline{x}) \mathcal{H}(\underline{x}). \tag{2.10}$$

as functions of β. Replacing (2.7) in (2.9) we get

$$S(\beta) = \beta \mathcal{E}(\beta) + \ln Z\,. \tag{2.11}$$

To make contact with the temperature we have to look at the entropy as a function of the average energy E,[4]

$$S(E) = \beta(E) E + \ln Z(\beta(E)) \tag{2.12}$$

where $\beta(E)$ is computed by inverting the relation $\mathcal{E}(\beta) = E$. Differentiating (2.12) with respect to E,

$$\begin{aligned} \frac{dS(E)}{dE} =& \beta(E) + \frac{d\beta(E)}{dE} E + \frac{d \ln Z}{d\beta} \frac{d\beta(E)}{dE} \\ &= \beta(E) + \frac{d\beta(E)}{dE} E - \mathcal{E}(\beta(E)) \frac{d\beta(E)}{dE} \\ &= \beta(E) \end{aligned} \tag{2.13}$$

In thermodynamics the inverse temperature $1/T$ is equal to the derivative of the (experimentally measurable) "thermodynamic entropy" with respect to the internal energy of the system. Thus, if we identify the Gibbs and thermodynamic entropies,[5] the identity (2.13) suggests the interpretation $\beta = 1/k_B T$.[6] One commonly says that β is the "inverse temperature".

[4] It is common in statistical mechanics and thermodynamics to use the same letter S for the entropy as a function of β, E or even $p(\cdot)$.

[5] This identification is not at all obvious and can here be viewed as a postulate. Alternatively it can be deduced from Boltzmann's postulate briefly discussed in Section 2.7

[6] T the temperature in degree Kelvin and k_B Boltzmann's constant in Joules per degree Kelvin which enters here because Gibbs entropy has no physical unit whereas thermodynamic entropy has units of Joules per Kelvin

2.3 Free energy and variational principle

On the way of our derivation of the Gibbs distribution we encountered a few important facts that we highlight in this section. But first we introduce a notation that is standard in statistical mechanics.

Bracket notation

Let $A(\underline{x})$ be any function of the configurations $\underline{x}$ of the system. These functions are often called *observables*. The average with respect to $p_G(\underline{x})$ is denoted by the bracket $\langle - \rangle$,

$$\langle A(\underline{x}) \rangle \equiv \frac{1}{Z} \sum_{\underline{x}} A(\underline{x}) e^{-\beta \mathcal{H}(\underline{x})} \,. \tag{2.14}$$

The normalization factor in such averages is always given by the partition function (2.8). It will become apparent in the next Chapter how convenient it is to have a reserved notation for the Gibbs average $\langle - \rangle$, and distinguish it from expectations $\mathbb{E}$ over other random objects.

Free energy

A notion of paramount importance is the *free energy* defined by

$$F(\beta) = -\frac{1}{\beta} \ln Z \,. \tag{2.15}$$

The important relationship (2.11), namely

$$F(\beta) = \mathcal{E}(\beta) - \beta^{-1} S(\beta) \tag{2.16}$$

suggests the thermodynamic interpretation of the free energy. This is the amount of energy in the system that is not in "disordered form" and can be extracted in the form of mechanical work, hence the adjective "free".

Computing, exactly or approximately, the free energy is often a major goal and when this is possible we learn a great deal about the model or system at hand. In particular from the free energy we can calculate the *internal energy* by differentiating $\beta F(\beta)$ with respect to β. Indeed (2.8) and (2.10) implie

$$\mathcal{E}(\beta) = \langle \mathcal{H}(\underline{x}) \rangle = -\frac{d}{d\beta} \ln Z = \frac{d}{d\beta}(\beta F(\beta)). \tag{2.17}$$

We can also compute the Gibbs entropy by differentiating $F(\beta)$ with respect to $1/\beta$. From (2.7) and (2.9)

$$\begin{aligned} S(\beta) &= -\langle \ln p_G(\underline{x}) \rangle \\ &= \ln Z - \beta \langle \mathcal{H}(\underline{x}) \rangle = \beta F(\beta) - \beta \frac{d}{d\beta}(\beta F(\beta)) \\ &= -\beta^2 \frac{d}{d\beta} F(\beta) = \frac{d}{d(1/\beta)} F(\beta) \end{aligned} \tag{2.18}$$

Finally, the *energy fluctuations* are obtained by differentiating twice $\ln Z$,

$$\langle \mathcal{H}(\underline{x})^2 \rangle - \langle \mathcal{H}(\underline{x}) \rangle^2 = -\frac{d^2}{d\beta^2}(\beta F(\beta)) \,. \tag{2.19}$$

Gibbs variationnal principle

Recall that we deduced the Gibbs distribution as the one which maximizes the functional (2.6). This can be formalized as follows. Define the *Gibbs free energy functional* as

$$\mathcal{F}(p(\cdot)) \equiv \sum_{\underline{x}} p(\underline{x}) \mathcal{H}(\underline{x}) - \beta^{-1} S(p(\cdot)) \tag{2.20}$$

This is a convex functional which satisfies the lower bound

$$F(\beta) \leq \mathcal{F}(p(\cdot)) \tag{2.21}$$

with equality attained for $p(\cdot) = p_G(\cdot)$. This general inequality is called the *Gibbs variational principle*. In practice it is often used to compute upper bounds to the free energy $F(\beta)$ by wisely choosing "trial distributions" for $p(\cdot)$. These upper bounds sometimes turn out to be useful approximations to the free energy or may even be sharp.

It is instructive to cast the variational principle in a language that is familiar to information theorists and statisticians. The *Kulback-Leibler divergence* between two distributions $p(\cdot)$ and $q(\cdot)$ is defined as

$$D_{KL}(p||q) \equiv \sum_{\underline{x}} p(\underline{x}) \ln\left(\frac{p(\underline{x})}{q(\underline{x})}\right) \tag{2.22}$$

and satisfies $D_{KL}(p||q) \geq 0$ with equality when $p = q$ (see exercises). Now, note that for $q = p_G$ we have (using (2.7), (2.15) and (2.20))

$$\begin{aligned} D_{KL}(p||p_G) &= \sum_{\underline{x}} p(\underline{x}) \ln\big(\frac{p(\underline{x})}{p_G(\underline{x})}\big) \\ &= -S(p) - \sum_{\underline{x}} p(\underline{x}) \ln p_G(\underline{x}) \\ &= -S(p) + \beta \sum_{\underline{x}} p(\underline{x}) \mathcal{H}(\underline{x}) + \ln Z \sum_{\underline{x}} p(\underline{x}) \\ &= \beta \mathcal{F}(p(\cdot)) - \beta F(\beta) \end{aligned} \tag{2.23}$$

The "free energy difference" between a trial distribution and the Gibbs distribution is equal (up to a factor β) to the Kullback-Leibler divergence. Also,

$$F(\beta) \leq \mathcal{F}(p(\cdot)) \quad \text{and} \quad D_{KL}(p||p_G) \geq 0$$

are one and the same inequality. It is fitting that $D_{KL}(p||q) \geq 0$ is often called the "Gibbs inequality".

2.4 Marginals, correlation functions and magnetization

Assume that a system is described by a Gibbs distribution. In practice, in order to answer many basic questions, it is often sufficient to compute (exactly or approximately) the first few marginals or even only the averages of a few important observables. In this section we collect a few related definitions and remarks.

Marginals

The definition of marginals is just the usual probabilistic one. More precisely the "first order" marginal, is defined as

$$\nu_i(x_i) = \sum_{\sim x_i} p_G(\underline{x}) \tag{2.24}$$

where $\sum_{\sim x_i}$ means that we sum over all x_j for $j = 1, \dots i-1, i+1, \dots n$. In other words we sum over all variables *except* x_i. The "second order" marginal is

$$\nu_{i,j}(x_i, x_j) = \sum_{\sim x_i, x_j} p_G(\underline{x}). \tag{2.25}$$

where we sum over all variables *except* x_i, x_j. Note that the marginals are normalized probability distributions.[7]

To illustrate the use of marginals, suppose we want to compute the averages of the observables, total number of particles $\sum_{i\in V} x_i$ and energy $\mathcal{H}(\underline{x})$, for the lattice gas model. By linearity of the Gibbs bracket

$$\langle \sum_{i\in V} x_i \rangle = \sum_{i\in V} \langle x_i \rangle \quad \text{and} \quad \langle \mathcal{H}(\underline{x}) \rangle = \sum_{\{i,j\}\in E} J_{ij} \langle x_i x_j \rangle - \sum_{i\in V} h_i \langle x_i \rangle$$

If the marginals are known we then can use

$$\langle x_i \rangle = \sum_{x_i} x_i \nu_i(x_i) \quad \text{and} \quad \langle x_i x_j \rangle = \sum_{x_i, x_j} x_i x_j \nu_{i,j}(x_i, x_j) \tag{2.26}$$

The reader should check these two identities.

Correlation functions

In the previous section we saw that the internal energy, energy fluctuations and entropy can be computed by differentiating the free energy. Something similar is also true for the averages (2.26). Consider the following perturbation of the Hamiltonian where we add "source terms"

$$\mathcal{H}(\underline{x}) \to \mathcal{H}(\underline{x}) + \sum_{i=1}^{n} \lambda_i x_i \tag{2.27}$$

[7] Marginals (2.24), (2.24) are often called one-point and two-point functions in the statistical mechanics literatture

where λ_i are real numbers. It is sometimes the case that if we know how to compute the free energy for the unperturbed Hamiltonian then we can also compute it for small values of λ_i's. When this optimistic situation is met, such perturbations may be turned into a useful theoretical tool. Indeed, suppose we have access to $\ln Z(\underline{\lambda})$, for $\underline{\lambda} = (\lambda_1, \ldots, \lambda_n)$. Then we can compute the following Gibbs brackets

$$\langle x_i \rangle = \frac{\partial}{\partial \lambda_i} \ln Z(\underline{\lambda})|_{\underline{\lambda}=0}, \qquad \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle = \frac{\partial^2}{\partial \lambda_i \partial \lambda_j} \ln Z(\underline{\lambda})|_{\underline{\lambda}=0}. \quad (2.28)$$

It is a general fact that higher order derivatives yield higher order cumulants. In statistical mechanics these cumulants are called "truncated correlation functions". For example the covariance $\langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle$ is the two-point truncated correlation function and the average $\langle x_i \rangle$ is the one-point function. It is a good exercise to compute the third order derivative (with respect to $\lambda_i, \lambda_j, \lambda_k$) to see what kind of truncated correlation function is obtained.

Finally we note that for binary variables (e.g., $x_i \in \{0, 1\}$ or $s_i \in \{+1, -1\}$ as is the case for a lattice gas, an Ising spin system, coding or SAT) the marginals $\nu_i(x_i)$ can be recovered from the averages $\langle x_i \rangle$. For example, for $x_i \in \{0, 1\}$ we have $\langle x_i \rangle = 0.\nu_i(0) + 1.\nu_i(1) = \nu_i(1)$ and from the normalization condition $\nu_i(0) = 1 - \langle x_i \rangle$. For $s_i \in \{+1, -1\}$ we have $\langle s_i \rangle = \nu_i(1) - \nu_i(-1)$ and $1 = \nu_i(1) + \nu_i(-1)$, thus $\nu_i(1) = \frac{1}{2}(1 + \langle s_i \rangle)$, $\nu_i(-1) = \frac{1}{2}(1 - \langle s_i \rangle)$. Similarly one can reconstruct $\nu_{i,j}(x_i, x_j)$ from one and two-point correlation functions (see exercises).

Magnetization

An observable that plays a specially important role in Ising spin systems is the magnetization of a spin configuration $m_n(\underline{s}) = \frac{1}{n} \sum_{i \in V} s_i$. The *average magnetization* (also simply called magnetization) is the expectation with respect to the Gibbs distribution.

$$m(\beta) \equiv \langle m(\underline{s}) \rangle = \frac{1}{n} \sum_{i \in V} \langle s_i \rangle. \quad (2.29)$$

According to the remarks of the previous paragraph, when the Hamiltonian contains a term $h \sum_{i \in V} s_i$ the average magnetization can be obtained as a derivative of the free energy with respect to the magnetic field,

$$m(\beta) = -\frac{1}{\beta} \frac{\partial}{\partial h} \ln Z = -\frac{\partial}{\partial h} f(\beta) \quad (2.30)$$

In general one can always add an infinitesimal magnetic field to the Hamiltonian, differentiate the free energy, and finally set this additional field to zero.

As a last remark we note that for certain models with a symmetry between sites it is often the case that $\langle s_i \rangle$ is independent of i, so that $\langle m(\underline{s}) \rangle = \langle s_i \rangle$. For example if we replace the square grid by a complete graph in the Ising model and take interaction constants independent of edges and vertices we have

a permutation symetry between sites, so $\langle s_i \rangle$ is obviously independent of i. This is the Curie-Weiss model treated in chapter 4.

2.5 Thermodynamic limit and notion of phase transition

The regime of validity of statistical mechanics is the *thermodynamic limit*, namely the asymptotic limit of large systems where the number of degrees of freedom tends to infinity, $n \to +\infty$. This is also the limit in which sharp *phase transitions* are well defined. Here a provide a first rather informal discussion of these concepts, which we will encounter again in coding, compressive sensing and constraint satisfaction, and which will be defined more precisely on a case by case basis.

Thermodynamic limit

For reasonably well defined models we expect that $\ln Z$, $S(\beta)$ and $\langle \mathcal{H}(\underline{x}) \rangle$ all scale like n, for large n. Such quantities are called *extensive*. Their thermodynamic limit, if it exists, is defined as

$$f(\beta) \equiv \lim_{n\to+\infty} \frac{1}{n} \ln Z, \qquad s(\beta) \equiv \lim_{n\to+\infty} \frac{1}{n} S(\beta), \qquad e(\beta) \equiv \lim_{n\to+\infty} \langle \mathcal{H}(\underline{x}) \rangle \tag{2.31}$$

Taking the limit of (2.11) we obtain

$$f(\beta) = e(\beta) - \beta^{-1} s(\beta)\,. \tag{2.32}$$

Relations (2.17), (2.18), (2.19) are also true for the limiting quantities scaled by $1/n$, *provided* one can permute $d/d\beta$ and $\lim_{n\to+\infty}$. This is the case as long as $f(\beta)$, $s(\beta)$ and $e(\beta)$ are "sufficiently smooth" functions of β. The issue here is a real one and is connected to the subject of phase transitions.

The thermodynamic limit for the correlation functions and the Gibbs distribution itself is not a simple matter. One cannot simply use the definition (2.7) and naively take the limit $n \to +\infty$ since the numerator and denominator both tend to infinity (generically exponentially fast). So what is the meaning of the Gibbs distribution in the thermodynamic limit? One way to proceed would be to compute all marginals for finite n and only then take their limits,

$$\lim_{n\to+\infty} \nu_i(x_i), \quad \lim_{n\to+\infty} \nu_{i,j}(x_i, x_j), \quad \lim_{n\to+\infty} \nu_{i,j,k}(x_i, x_j, x_k), \quad \ldots \tag{2.33}$$

The limiting Gibbs distribution can then be defined as the distribution with has this set of marginals. Because of phase transition phenomena such limits are *not* always defined in a unique way.

Say more here?

Phase transitions

Let us now say a few words about phase transitions, a subject to which we will come back in due course. The free energy $f(\beta)$ is always a *continuous* and *convex* function of β. To see this note that for finite n, $F(\beta)/n$ is an analytic and convex function of β. Convexity can be seen from the positivity of the variance of the Hamiltonian in (2.19). Moreover the limit of a sequence (indexed by n) of continuous convex functions is continuous and convex, thus $f(\beta)$ is continuous and convex (provided this limit exists!). However such a general argument cannot guarantee more than continuity, and indeed the limiting free energy $f(\beta)$ may develop non-analyticities. Such non-analytic points on the β axis are called *phase transition points* or *thresholds*. Typically these points are isolated and correspond to lack of differentiability at some order.[8]

Points where the first derivative of $f(\beta)$ has a jump are called *first order* phase transition points; those where the first derivative is continuous but the second derivative is discontinuous are called *second order* phase transition points.[9]. Phase transitions of higher order are also possible: a phase transition of n-th order is one where the first $n-1$ derivatives of $f(\beta)$ are all continuous but the n-th has a jump discontinuity. This classification of phase transitions is due to Ehrenfest and dates back to the early days of statistical physics. We stress that this is not the only possible classification, nor the most modern one, however it is one that will suit our needs quite well.

Temperature is not the only parameter with respect to which the free energy can be non-differentiable. For example in the canonical Ising model there are first order phase transitions with respect to the magnetic field h. This helps us understand the statement made above about the non-unicity of the Gibbs distribution in thermodynamic limit. Indeed we saw that the magnetization is obtained as the derivative of the free energy with respect to h; thus since at a first order phase transition point this derivative is discontinuousm the magnetization can take two distinct values, which means that one should define two one-point marginals and hence two limiting Gibbs distributions. Hence typically at a first order phase transition point we expect non-unicity of the limiting marginals and Gibbs distribution.

In Chapter 4 we solve explicitly useful toy models - the Curie-Weiss model and the Ising model on a tree - which will allow us to discuss phase transitions more concretely. Furthermore a mini-review of the phase transitions in the canonical Ising and lattice gas models is found as an aside at the end of that Chapter 4.

[8] More exotic such as for example an essential singularity, where all derivatives exist but the Taylor expansion does not converge, are possible, but will not be encountered in our problems.

[9] Such points necessarily form a set of measure zero by a theorem of Alexandrov on limits of convex functions

2.6 Spin glass models - random Gibbs distributions

In the next chapter we will see that our three problems, coding, compressive sensing and satisfiability, can be formulated as a particular class of statistical mechanics models, the so-called *spin glass models*. In this paragraph we provide a first glimpse on the very large and subtle topic of spin-glasses.

One of the ambitions of statistical mechanics is to describe the rich variety of "phases" of condensed matter, e.g, gases, liquids, crystalline solids, metals, insulators, semi-conductors, superconductors, superfluids, magnetism, liquid crystals, polymers, glasses, emulsions etc. In this list, "ordinary glass", although empirically known and manufactured since very ancien times, is scientifically ill-understood and is arguably one of the most intriguing phases of matter. Ordinary glass is an amorphous material where the geometrical arrangement of atoms is frozen as in a solid, but at the same time is irregular as in a liquid. It is believed that in a sense ordinary glass is a "frozen liquid" with a viscosity so huge that it does not flow for all practical purposes. There also exist magnetic materials which have a glassy behaviour with their magnetization responding very slowly to external magnetic fields. Here we will not dwell on the various physical types and properties of glassy materials and we only limit ourselves to simple models.

Spin glass *models* are Ising or generalized spin systems (e.g., (2.2), (2.3)) with *random* interaction constants.[10]

EXAMPLE 6 The usual *Edwards-Anderson model* has Hamiltonian (2.2) with *random* i.i.d Bernoulli coupling constants, $\mathbb{P}(J_{ij} = \pm J) = 1/2$, and $h_i = h$ is constant. In another variant one can take iid Gaussian coupling constants. The analysis of this model is still far from being rigorously understood. □

EXAMPLE 7 The *random field Ising model* also has Hamiltonian (2.2), but now the interaction is constant $J_{ij} = J > 0$ and the magnetic field is i.i.d Bernoulli, $\mathbb{P}(h_i = \pm h) = 1/2$ This is also a very non-trivial and open questions remain. □

EXAMPLE 8 In the *Sherrington-Kirkpatrick model* G is the complete graph on n vertices with $n(n-1)/2$ edges (like in the Curie-Weiss model). The coupling constants J_{ij} are iid Bernoulli with $\mathbb{P}(J_{ij} = \pm J/\sqrt{n}) = 1/2$ or i.i.d Gaussian $\mathcal{N}(0, \frac{J^2}{n})$. The magnetic field is generally taken constant. The scaling of the coupling constants by $1\sqrt{n}$ is necessary in order to have fluctuations of $O(\sqrt{n})$ for the Hamiltonian on the complete graph. The analysis of this model in Chapter 7 will serve us well as a steping stone towards compressed sensing. □

Variants of these models use other distributions for the interaction constants, for example Gaussians. One can also take more complicated models with more general interactions, e.g. J_A's in (2.3) may be random variables, or the underlying

[10] Such models where first introduced in the 1970's in an attempt to capture glassy properties of magnetic materials, the idea being that their glassy behavior is related to physical interactions of varying intensity and sign between magnetic moments, and it was proposed that these interactions be modelled as random variables.

graph may be random. The study of spin glass models has turned out be very non-trivial and has been a source of many fundamental concepts in statistical mechanics of so-called *disordered systems*. Fortunately, the spin glass models that will be relevant for our three problems are defined on complete or locally tree-like graphs and as we will see the absence of "low dimensional geometry" makes them somehow much easier to study than the Edwards-Anderson and random field Ising model. easier to study.

The Gibbs distribution associated to a spin glass Hamiltonian has two levels of randomness. First we have the randomness of the Hamiltonian itself, i.e. the interaction constants or the underlying graph. Once the Hamiltonian is sampled from a specified ensemble we have a fixed instance of a Gibbs distribution which is a probability distribution over the spin (or lattice gas) variables. So the study of spin glass models is the study of *ensembles of random Gibbs distributions*.

A word about some standard terminology is in order here.[11] The random interaction constants J_A of the Hamiltonian are called *quenched variables* because once the instance is specified they are fixed or "frozen" once for all. The spin or lattice gas degrees of freedom $s_1, \ldots, s_n$ or $x_1, \ldots, x_n$ are called *annealed variables* because they "adapt" themselves into typical configurations of the Gibbs measure.

A word about notation is also in order. It is very convenient to have two separate notations to distinguish averages with respect to quenched and annealed variables. The expectations with respect to the Gibbs distribution are always denoted by the bracket $\langle - \rangle$ and those with respect to the quenched variables by $\mathbb{E}$ with possible subscripts describing the ensemble. Thus if $A(\underline{x})$ is an observable (say the magnetization) the total average over annealed and quenched variables is denoted $\mathbb{E}[\langle A(\underline{x}) \rangle]$. Let us insist on two elementary facts. First, $\langle A(\underline{x}) \rangle$ is a random object depending on all quenched variables. Second, it would be meaningless to permute the two expectations $\mathbb{E}$ and $\langle - \rangle$.

Quenched randomness is ubiquitous in many engineering optimization problems where one has to deal with particular instances that belong to a model ensemble. This is the point of view that we took in the definition of the coding, compressive sensing and satisfiability problems. As we will see in the next Chapter, once an instance of the ensemble is specified, the Gibbs distribution appears quite naturally in the mathematical formulation. So in a sense the connections between our models and the statistical mechanics of spin glasses is not surprising; in fact they are very natural. Such connections have been recognized since the 1970's for various computer science problems, e.g., the travelling salesman, graph partionning, and neural networks (see the notes for references).

[11] This terminology comes from the manufacturing process of ordinary glass.

2.7 Gibbs distribution from Boltzmann's principle[12]

We already pointed out that here is no unique way to introduce the Gibbs distribution. Rather, as with any physical law, it has to be guessed from experiments, plausible assumptioms, and modeling, which all lead to conclusions that are experimentally validated. From a physical point of view the maximum entropy principle is perhaps not very satisfying because Shannon's entropy is not a "primary" physical or at least "simple" quantity. In this section we provide a derivation, based on two basic principles, and which is perhaps closer in spirit to the original ones by Maxwell, Boltzmann, Gibbs, Einstein and others in the early 20-th century.

For concreteness the reader may keep in mind the lattice gas model throughout the arguments of this section. We suppose that the particles have a dynamics with "trajectories on the lattice," $x_i(t)$, $i = 1, \dots, n$, parametrized by time t.

Uniform microcanonical measure

Let $[0, T]$ be the time interval over which we measure an observable quantity $A(\underline{x}(t))$ and let τ be a characteristic microscopic time scale, for example the time scale on which a single particle jumps from a position to a neighboring one. In practice we have $T \gg \tau$, so we think as T being very large ($T/\tau \to +\infty$). For an isolated system the energy is conserved. Thus during the measurement interval the state of the system $\underline{x}(t)$ will wander across the energy surface $\Gamma_E \subset \{0,1\}^{|V|} = \{\underline{x} \mid \mathcal{H}(\underline{x}) = E\}$. Let $t(\underline{x})/T$ be the fraction of time it spends in state $\underline{x}$.

Our first principle states that when $T \gg \tau$, the fraction of time $t(\underline{x})/T$ spent in state $\underline{x}$, is given by the uniform distribution on the energy surface Γ_E. In other words for $t(\underline{x})/T$ we take,

$$\mu_E(\underline{x}) = \frac{\mathbb{1}(\underline{x} \in \Gamma_E)}{W(E)} \tag{2.34}$$

where the normalization factor is

$$W(E) = \sum_{\underline{x} \in \{0,1\}^{|V|}} \mathbb{1}(\underline{x} \in \Gamma_E). \tag{2.35}$$

The measure (2.34) is called the *microcanonical distribution*. In words the first principle states that *an isolated system spends an equal fraction of time in all states of belonging to the energy surface.*

A fundamental consequence is that we can replace time averages of observables by configurational averages,

$$\frac{1}{T} \int_0^T dt\, A(\underline{x}(t)) \approx \sum_{\underline{x} \in \{0,1\}^{|V|}} \mu_E(\underline{x}) A(\underline{x}), \qquad T \gg \tau \tag{2.36}$$

[12] This section is not needed for the main development and can be skipped in a first reading.

The idea here is that an experiment (or a measurement) returns a time average (the left hand side) and that as theoreticians we can compute this average from the microcanonical average (the right hand side). In particular we can esssentially ignore the underlying microscopic dynamics.

Often equ. (2.36) is formalized and called the *ergodic hypothesis*. The ergodic hypothesis states that the dynamics exactly satisfies this identity in the limit $T \to +\infty$, for almost all initial conditions $\underline{x}(0)$ (note that the right hand side does not depend on the initial condition). Although the ergodic hypothesis has played a very important historical and foundational role, it has never been proved for macroscopic systems of interest in statistical mechanics. Its physical relevance is also not completely clear, and as such the hypothesis does not explicitly specify the relevant class of observables, initial conditions, and relation between system size and time scales. Nevertheless, the ergodic hypothesis has developped into a beautiful branch of mathematics (ergodic theory) for dynamical systems with *few* degrees of freedom.

Boltzmann's principle

Consider the normalization $W(E)$ of the microcanonical measure. Generically this has an exponential behavior in the number of degrees of freedom. The *Boltzmann entropy* is defined as

$$S_B(E) = \ln W(E). \tag{2.37}$$

This is a purely *combinatorial* object in the sense that $W(E$ is the number of microscopic states belonging to the energy surface Γ_E.

EXAMPLE 9 Let us consider the lattice gas model introduced in the previous example for the non-interacting case $J = 0$. Since the energy surface is the set $\Gamma_E = \{\underline{x} \mid \sum_{i\in 1}^{n} x_i = E/\mu\}$ there must be E/μ lattice nodes with $x_i = 1$ among a total of n sites. Hence

$$W(E) = \binom{n}{E/\mu} \simeq \exp\left(n h_2\Big(\frac{E}{\mu n}\Big)\right), \tag{2.38}$$

where $h(u) = -u\ln u - (1-u)\ln(1-u)$ is the binary entropy function expressed with the natural logarithm. In the thermodynamic limit we obtain

$$s(e) = \lim_{\substack{n\to\infty \\ E/n=e}} \frac{1}{n} S_B(E) = h\left(\frac{e}{\mu}\right), \tag{2.39}$$

where $e = E/n$ is the energy per particle. In this simple example the entropy $s(e)$ is a concave function of e. For physically sensible Hamiltonians the Boltzmann entropy must be concave; but this is not always the case in computer science and coding problems where the cost functions are not necessarily physical. □

There is a purely thermodynamic and experimentally measurable notion of entropy elucidated in the 19-th century, along with the notions of heat and

work, by Carnot, Clausius, Joule, Helmholtz, Kelvin and others. For a system in thermodynamic equilibrium with homogeneous temperature T and pressure p, the thermodynamic entropy $S_{\text{ther}}(E, V)$ is a function of the total energy E and volume V satisfying

$$\frac{\partial S_{\text{ther}}}{\partial E} = \frac{1}{T}, \qquad \frac{\partial S_{\text{ther}}}{\partial V} = \frac{p}{T}\,. \tag{2.40}$$

From T and p one can in principle recover S_{ther}. The units of S_{th} are Joules per degree Kelvin.

Boltzmann's principle postulates equality of the thermodynamic and Boltzmann entropies,

$$S_{\text{ther}} = k_B \ln W(E)\,. \tag{2.41}$$

Here, k_B is Boltzmann's constant with units of Joules per degree Kelvin (this constant is needed because we defined S_{B} as a pure number). Boltzmann's principle is one of the most far reaching laws of physics: it identifies a thermodynamic measurable quantity with a purely combinatorial counting object.[13] Combining this identity with the first equation in (2.40) we get

$$\frac{\partial \mathcal{S}_{\text{B}}}{\partial E} = \frac{1}{k_B T}. \tag{2.42}$$

In the next paragraph we will see that this identification is a crucial ingredient in the derivation of the Gibbs distribution.

Derivation of the Gibbs distribution

The microcanonical distribution (2.34) only characterizes an isolated system with fixed energy E. However, real macroscopic systems are not isolated. One should also notice that in practice, in order to reach thermal equilibrium it is necessary to put systems in contact with a "thermal bath," i.e., an infinite reservoir at a constant temperature.

For simplicity, we take the lattice gas as our big reservoir and suppose it is isolated with total energy E. The real system of interest is *a much smaller* but still *macroscopic* system $\Sigma \subset V$ (see Figure 2.3). We label the degrees of freedom in Σ as $(x_1, \ldots, x_m)$ and those outside Σ by $(x_{m+1}, \ldots, x_n)$. The regime of interest is $1 \ll m \ll n$. We are interested in computing *only* averages of observables $A(x_1, \ldots, x_m)$ which depend on the degrees of freedom of the smaller system Σ. Of course we can compute them with the microcanonical distribution

$$\mu_E(x_1, \ldots, x_n) = \frac{\mathbb{1}((x_1, \ldots, x_n) \in \Gamma_E)}{W(E)}, \tag{2.43}$$

but clearly, since the observable depends only on $x_1, \ldots, x_m$, we only need the marginal of this distribution over the degrees of freedom of Σ.

[13] Even that matter is constituted of discrete entities that can be counted, atoms and molecules, was far from universally accepted in Boltzmann's days.

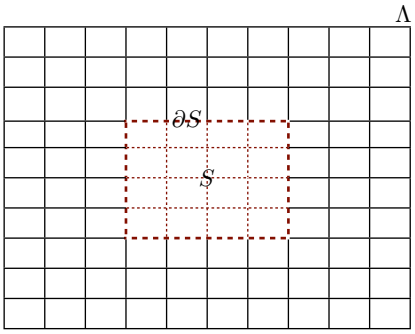


Figure 2.3 The system S is embedded in a thermal bath V. The total system V is considered as an isolated system and its total energy E is conserved. We compute the induced measure on S.

We now show that the marginal of (2.34) is the Gibbs distribution with inverse temperature $\frac{1}{k_B T} = \frac{\partial S_B(E)}{\partial E}$. This is the temperature of the thermal bath.

The marginal distribution for Σ reads

$$\begin{aligned}\mu_\Sigma(x_1,\dots,x_m) &= \sum_{x_{m+1},\dots,x_n} \mu_E(x_1,\dots,x_n) \\ &= \frac{\sum_{x_{m+1},\dots,x_n} \mathbb{1}((x_1,\dots,x_n)\in\Gamma_E)}{\sum_{x_1,\dots,x_n} \mathbb{1}((x_1,\dots,x_n)\in\Gamma_E)}.\end{aligned} \tag{2.44}$$

The total energy E is a sum of the energy inside Σ, the energy outside Σ and an interaction part between the inside and the outside,

$$E = \mathcal{H}(x_1,\dots,x_n) = \mathcal{H}_\Sigma(x_1,\dots,x_m) + \mathcal{H}_{V\setminus\Sigma}(x_{m+1},\dots,x_n) + \mathcal{H}_{\text{int}}. \tag{2.45}$$

Generically $\mathcal{H}_\Sigma$ is of order m (the volume of Σ), $\mathcal{H}_{V\setminus\Sigma}$ is of order $n-m$ (the volume of the outside of Σ) and $\mathcal{H}_{int}$ is of order the surface of Σ. In d dimensions the surface of Σ is of order $m^{(d-1)/d} << m << n-m$, thus neglecting the interaction in (2.45) we conclude that: if $(x_1,\dots,x_n)$ belongs to the energy surface Γ_E then $(x_{m+1},\dots,x_n)$ belongs to the energy surface $\Gamma_{E-\mathcal{H}_\Sigma(x_1,\dots,x_m)}$.

With these remarks (2.44) becomes

$$\begin{aligned}\mu_\Sigma(x_1,\dots,x_m) &= \frac{\sum_{x_{m+1},\dots,x_n} \mathbb{1}((x_{m+1},\dots,x_n)\in\Gamma_{E-\mathcal{H}_\Sigma(x_1,\dots,x_m)})}{\sum_{x_1,\dots,x_m}\sum_{x_{m+1},\dots,x_n}\mathbb{1}((x_{m+1},\dots,x_n)\in\Gamma_{E-\mathcal{H}_\Sigma(x_1,\dots,x_m)})}\\ &= \frac{\exp(S_B(E-\mathcal{H}_S(x_1,\dots,x_m))}{\sum_{x_1,\dots,x_m}\exp(S_B(E-\mathcal{H}_\Sigma(x_1,\dots,x_m))}\\ &= \frac{\exp(S_B(E)-\mathcal{H}_\Sigma(x_1,\dots,x_m)\frac{\partial}{\partial E}S_B+\dots)}{\sum_{x_1,\dots,x_m}\exp(S_B(E)-\mathcal{H}_S(x_1,\dots,x_m)\frac{\partial}{\partial E}S_B+\dots)}\\ &= \frac{\exp\left(-\mathcal{H}_\Sigma(x_1,\dots,x_m)/k_BT\right)}{\sum_{x_1,\dots,x_m}\exp\left(-\mathcal{H}_\Sigma(x_1,\dots,x_m)/k_BT\right)},\end{aligned}$$

The second equality uses the definition of the Boltzmann entropy. The third equality uses a Taylor expansion to first order (since $n \gg m$ implies $E \gg \mathcal{H}_\Sigma(x_1,\dots,x_m)$). The last equality is the point where Boltzmann's principle is used. The final result is exactly the Gibbs distribution for the small system Σ with a temperature equal to the one of the thermal bath.

This derivation shows that the Gibbs distribution and the microcanonical distributions are "equivalent" in the sense that we can compute the average of the observable $A(x_1,\cdots,x_m)$ directly from one or the other distribution. This is related to the subject of "equivalence of ensembles" in statistical mechanics: the microcanonical distribution (also called the microcanonical ensemble) and the Gibbs distribution (also called the canonical ensemble) are equivalent. For this equivalence to hold an important assumption was that the interaction $\mathcal{H}_{\text{int}}$ between the system and its complement can be neglected. For physical finite dimensional systems with local interactions (finite range or fast decaying with distance) between particles this is not a problem.[14] However if one deals with "infinite dimensional" systems (meaning that $d \to +\infty$ or that the graph G cannot be metrically embedded in a finite dimensional space) this assumption breaks down. In our coding, compressed sensing and satisfiability problems the underlying graphs are in a sense infinite dimensional and the equivalence of ensembles does not hold.

2.8 Notes

The ambition of statistical mechanics is to explain the great variety of phases of condensed matter, their properties and the transitions among the phases. One can easily imagine this is a rich and huge discipline just by contemplating a far from exhaustive list of examples of phases of matter: gases, liquids, solids, metals, plasmas, magnets, insulators, superfluids, superconductors, Bose-Einstein condensates, crystals, quasi-crystals, emulsions, glasses etc. Experimentalists keep on finding new phases, e.g., by lowering the temperature and/or increasing the

[14] Physical systems where this breaks down are gravitational systems.

pressure. The subject is more than a hundred years old and the present chapter has of course not even scratched its surface. We only give some pointers to a small subset of classic texts.

For the reader interested by the rich and fascinating history of statistical mechanics, as well as the debates on foundational issues, we recommend (Brush 1983). The derivation based on the maximum entropy principle goes back to (Jaynes 1957). Good introductions to statistical physics are (e.g Schroeder 2000, Thompson 1988); a more advanced and classic graduate level text is (Huang 1987). For treatises with emphasis on foundational issues see (e.g. Penrose 2005, Gallavotti 1999). The Ising model, which was introduced in 1928 by Lenz and Ising, has a long and distinguished history reviewed in (Brush 1967). More generally, the class of spin models on regular lattices forms an unavoidable portion of the discipline and their study has led to a large body of mathematically rigorous results precisely defining and characterizing the notions of "phases" and "phase transition". A comprehensive treatise is (Simon 1993). Statistical mechanics of disordered systems is a subject of its own; and "glass" is just one example of disordered system. In the seventies spin glass models where put forward to investigate the special types of behavior and phase transitions that occur in this area, see (Fisher & Hertz 1991). Many of the techniques developed in this course have their origin in the replica and cavity theories of so-called "mean field" spin glass models such as the SK model. The classic reference covering the body of work spanning the seventies and eighties is (Mézard et al. 1987*a*). Connections to neural networks, graph partionning and other optimization problems are also found in this reference. In recent years there has been a lot of progress in the mathematically rigorous aspects of the theory (Talagrand 2011).

Problems

2.1 Gibbs distribution. Give the details of the derivation leading to (2.7) and (2.8).

2.2 Energy fluctuations. Derive the formula (2.19) for energy fluctuations.

2.3 Positivity of Kullback-Leibler divergence. Prove in two different ways that $D_{KL}(p||q) \geq 0$ with equality if and only if $p(\underline{x}) = q(\underline{x})$ for all $\underline{x}$. Hint: use $\ln u \leq u - 1$ for $u > 0$ or the convexity of $f(u) = u \ln u$.

2.4 Correlation functions from derivatives of partition function. Check the formulas (2.28) and also

$$\frac{\partial^3}{\partial\lambda_i\lambda_j\lambda_k} \ln Z(\underline{\lambda})|_{\underline{\lambda}=0} = \langle x_i x_j x_k\rangle - \langle x_i x_j\rangle\langle x_k\rangle - \langle x_j x_k\rangle\langle x_i\rangle - \langle x_i x_k\rangle\langle x_j\rangle + 2\langle x_i\rangle\langle x_j\rangle\langle x_k\rangle$$

2.5 Marginals for binary spins. Consider any spin system with binary variables $s_i \in \pm 1$. Express the marginals $\nu_i(s_i)$ and $\nu_{i,j}(s_i, s_j)$ in terms of the averages $\langle s_i\rangle$, $\langle s_j\rangle$ and $\langle s_i s_j\rangle$.

2.6 Ising model in one dimension: transfer matrix method. The aim

of this problem is to solve the one-dimensional Ising model by the transfer matrix method. The Hamiltonian of the one-dimensional Ising model on a ring (circle) with n sites

$$\mathcal{H} = -J \sum_{i=1}^{n-1} s_i s_{i+1} - J s_n s_1 - h \sum_{i=1}^{n} s_i$$

Consider the "transfer matrix"

$$T = \begin{pmatrix} e^{\beta J + \beta h} & e^{-\beta J} \\ e^{-\beta J} & e^{\beta J - \beta h} \end{pmatrix}$$

(i) Show that the partition function can be expressed as $Z_N = \mathrm{tr}\,(T^n)$ where tr is the sum over eigenvalues (the trace).
(ii) Find the eigenvalues of T and show that the free energy per spin is in the thermodynamic limit

$$f(\beta, h) = -\beta^{-1} \ln[e^{\beta J} \cosh(\beta h) + (e^{2\beta J} \sinh^2(\beta h) + e^{-2\beta J})^{1/2}].$$

(iii) Compute the magnetization (the easiest way is to use (2.30)) and plot it m as a function of h for various values of β. Convince yourself both on the plot and from the analytic formula that this curve does not display a phase transition for any temperature $T > 0$.

3 Formulation of Problems as Spin Glass Models

The three problems introduced in Chapter 1 can be reformulated in a statistical physics language. Both coding and compressive sensing are inference problems, and from a Bayesian point of view Gibbs distributions appear quite naturally. For the random satisfiability problem, which is not an inference problem, the Gibbs distribution may seem less natural. The simplest and perhaps most natural distribution that one would settle to study is the uniform one over the set of all satisfying assignments.[1] However, given a formula, the set of satisfying assignments is not known and even worse, we y do not know if it is empty. So it is difficult to get a handle (or even define) the uniform distribution, and instead we introduce a Gibbs distribution which has the advantage of being well defined for *all* formulas. One hopes to get a good approximation of the uniform distribution when the inverse temperature β tends to infinity.

In all cases we end up with *spin glass models*. What do we mean by this? Take for example the coding or satifiability examples. We can think of the bits which are to be transmitted, or the Boolean variables which take one of two possible thruth values, as *spins*. This explains why we talk about *spin* systems. In compressed sensing the signal components are continuous and this model falls in the class of systems with "continuous spins". Thus the microscopic "spin" degrees of freedom are bits, Boolean variables or signal components. Moreover each problem has its own cost function or Hamiltonian over the spin assignments. We already encountered two such cost functions in compressive sensing and satisfiability (see Equs. (1.10) and (1.13)). But where is the "glass"? In coding the way we have defined our code ensemble, a parity check constrains a random subset of the bits so the graph and interactions are random. The same is true for satisfiability. In compressed sensing the measurement matrices are random which results in random interaction constants between the continuous spins (note that the graph itself is not random but bipartite complete). In all our models the randomness is quenched: once we pick an instance from the appropriate ensemble we have a fixed Gibbs distribution. In this sense our models fall in the general category of "spin glass models".

To summarize, our reformulations will lead us to *random* Gibbs distributions or spin glass models. For each problem we will identify a Hamiltonian function

[1] In a sense this distribution is a microcanonical measure introduced in Sec. 2.7.

over "spins" with underlying graphs and interaction constants belonging to a random ensemble.

3.1 Coding as a spin glass model

Let $\mathcal{C}$ be a code from Gallager's (d_v, d_c) ensemble of block length n. Recall that d_v is the degree of variable nodes, d_c is the degree of check nodes, and n is the number of variable nodes. We have $nd_v = md_c$ where m is the number of parity checks.

Assume that we transmit the codeword $\underline{x} = (x_1, \ldots, x_n)$ through a binary, memoryless symmetric channel without feedback, and let $\underline{y} = (y_1, \ldots, y_n)$ be the received word. We will use the spin variable notation for the codebits. This means that we write $s_i = (-1)^{x_i}$ (or $s_i = 1 - 2x_i$). The channel is described by transition probabilities

$$p(\underline{y}|\underline{s}) = \prod_{i=1}^{n} p(y_i|s_i) \tag{3.1}$$

The three examples of channels to which we will refer most often are the BEC, BSC, and BAWGNC introduced in Sect. 1.1. While (for us) the input alphabet is always binary, the received symbols live in discrete or continuus alphabets depending on the channel. To have a unified discussion we sometimes abuse notation and use the generic summation symbol $\sum_{\underline{y}}$ to mean "sum" or "integral" over the alphabet of received bits.

We will assume that the transmitted (input) codeword $\underline{s}^{\text{in}}$ is selected uniformly at random from $\mathcal{C}$. Thus the joint distribution for $(\underline{s}, \underline{y})$ is $p(\underline{y}|\underline{s})\mathbb{1}(\underline{s} \in \mathcal{C})/|\mathcal{C}|$ and the posterior probability distribution of $\underline{s}$ given the received word $\underline{y}$ is

$$p(\underline{s} \mid \underline{y}) = \frac{p(\underline{y}|\underline{s})\mathbb{1}(\underline{s} \in \mathcal{C})}{\sum_{\underline{s}} p(\underline{y}|\underline{s})\mathbb{1}(\underline{s} \in \mathcal{C})}. \tag{3.2}$$

MAP decoding

Let $\widehat{s}_i(\underline{y})$ be a decoding estimate (based on all channel outputs) for the i-th bit. Since $\underline{s}^{\text{in}}$ is picked uniformly at random from the code, the probability that bit i is wrongly decoded, called *bit probability of error*, is

$$\frac{1}{|\mathcal{C}|} \sum_{\underline{s}^{\text{in}} \in \mathcal{C}} \mathbb{P}[\widehat{s}_i(\underline{Y}) \neq s_i^{\text{in}}]. \tag{3.3}$$

Thus we define *average bit probability of error* as

$$P_b = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{|\mathcal{C}|} \sum_{\underline{s}^{\text{in}} \in \mathcal{C}} \mathbb{P}[\widehat{s}_i(\underline{Y}) \neq s_i^{\text{in}}] \tag{3.4}$$

As explained in Chapter 1, a central task is to compute the average bit probability of error for various codes and decoders, and satisfy various performance and complexity requirements.

Among all possible estimators the *bit-MAP estimate* (MAP stands for "maximum a posteriori") wil play an important role. By definition

$$\widehat{s}_i^{\mathrm{MAP}}(\underline{y}) = \mathrm{argmax}_{s_i}\ \nu_i(s_i|\underline{y}) \tag{3.5}$$

where $\nu_i(s_i|\underline{y})$ is the marginal of the posterior $p(\underline{s}|\underline{y})$. This estimator is *optimal* in the sense that it minimizes the "bit probability of error" (see below for a proof). However, in general, we do not have low complexity algorithms to compute it efficiently. Nevertheless we will see that it has a very natural interpretation in terms of the "magnetization" of a spin glass model. Even more interestingly, in Chapter 10 we will discover through such connections that it is intimately related to other low complexity estimators.

Although we will not deal much with it, we mention the *block-MAP estimate* and the associated block probability of error

$$\underline{\widehat{s}}^{\mathrm{MAP}}(\underline{y}) = \mathrm{argmax}_{\underline{s}}\ p(\underline{s} \mid \underline{y}), \quad P_{\mathrm{B}} = \frac{1}{|\mathcal{C}|} \sum_{\underline{s}^{\mathrm{in}} \in \mathcal{C}} \mathbb{P}[\underline{\widehat{s}}(\underline{Y}) \neq \underline{s}^{\mathrm{in}}].$$

The block-MAP estimator is also optimal in the sense that among all possible block estimators it minimizes the block probability of error P_B. We will see that block-MAP decoding is equivalent to finding the minimum energy states of a Hamiltonian. We will also see that there is a natural "finite temperature" decoder which interpolates between the bit-MAP and block-MAP decoders.

Before proceeding let us prove an important fact, namely that the bit-MAP estimator is optimal. This is most easily seen from the following representation for the bit probability of error,[2]

$$\begin{aligned}
\frac{1}{|\mathcal{C}|} \sum_{\underline{s}^{\mathrm{in}} \in \mathcal{C}} \mathbb{P}[\widehat{s}_i(\underline{Y}) \neq s_i^{\mathrm{in}}] &= \frac{1}{|\mathcal{C}|} \sum_{\underline{s}^{\mathrm{in}} \in \mathcal{C}} \sum_{\underline{y}} p(\underline{y}|\underline{s}^{\mathrm{in}})\, \mathbb{1}(\widehat{s}_i(\underline{y}) \neq s_i^{\mathrm{in}}) \\
&= \sum_{\underline{y}} p(\underline{y}) \sum_{\underline{s}} p(\underline{s}|\underline{y})(1 - \mathbb{1}(\widehat{s}_i(\underline{y}) = s_i)) \\
&= 1 - \sum_{\underline{y}} p(\underline{y}) \sum_{s_i = \pm 1} \nu_i(s_i|\underline{y})\, \mathbb{1}(s_i = \widehat{s}_i(\underline{y})) \\
&= 1 - \sum_{\underline{y}} p(\underline{y})\, \nu_i(\widehat{s}_i(\underline{y})|\underline{y}).
\end{aligned}$$

Clearly, the last expression shows that the choice (3.5) for $\widehat{s}_i(\underline{y})$ minimizes the bit probability of error (3.3). The optimality of the block-MAP estimate is left as an exercise for the reader.

[2] Here $p(\underline{y}) \equiv \sum_{\underline{s}} p(\underline{y}|\underline{s})\mathbb{1}(\underline{s} \underline{\in} \mathcal{C})/|\mathcal{C}|$.

The posterior distribution as a spin glass model

We now show that the posterior distribution $p(\underline{s} \mid \underline{y})$ is a random Gibbs distribution. Recall that a code is represented by a bipartite factor graph with variable nodes $i = 1, \ldots, n$ and checks[3] $a = 1, \ldots, m$; like in Fig. 1.1. We call ∂a the set of variable nodes connected to check a. A code word $\underline{x}$ has to satisfy all parity check constraints $\sum_{i \in \partial a} x_i = 0$, $a = 1, \ldots, m$. In spin language this is equivalent to $\prod_{i \in \partial a} s_i = 1$ for all checks. Thus the prior distribution over codewords can be written as

$$\frac{\mathbb{1}(\underline{s} \in \mathcal{C})}{|\mathcal{C}|} = \frac{1}{|\,\mathcal{C}\,|} \prod_{a=1}^{m} \frac{1}{2}(1 + \prod_{i \in \partial a} s_i). \tag{3.6}$$

Replacing the channel law (3.1) and (3.6) in the posterior (3.2) we get

$$p(\underline{s}|\underline{y}) = \frac{\prod_{i=1}^{n} p(y_i|s_i) \prod_{a=1}^{m} \frac{1}{2}(1 + \prod_{i \in \partial a} s_i)}{\sum_{\underline{s}} \prod_{i=1}^{n} p(y_i|s_i) \prod_{a=1}^{m} \frac{1}{2}(1 + \prod_{i \in \partial a} s_i)} \tag{3.7}$$

Now we divide the numerator and denominator by $\prod_{i=1}^{n} p(y_i| - 1)$ and use

$$\frac{p(y_i|s_i)}{p(y_i| - 1)} = e^{h_i s_i + h_i} \tag{3.8}$$

where we have introduced the *loglikelihood variable*[4] associated to channel observation y_i

$$h_i = \frac{1}{2} \ln \frac{p(y_i| + 1)}{p(y_i| - 1)}. \tag{3.9}$$

This yields our final representation

$$p(\underline{s}|\underline{y}) = \frac{1}{Z} \prod_{a=1}^{m} \frac{1}{2}(1 + \prod_{i \in \partial a} s_i) \prod_{i=1}^{n} e^{h_i s_i} \tag{3.10}$$

where the normalizing factor in the denominator is

$$Z = \sum_{\underline{s}} \prod_{a=1}^{m} \frac{1}{2}(1 + \prod_{i \in \partial a} s_i) \prod_{i=1}^{n} e^{h_i s_i}. \tag{3.11}$$

It is equivalent to describe the channel outputs by $\underline{h} = (h_1, \ldots, h_n)$ or $\underline{y} = (y_1, \ldots, y_n)$. We will often interchange them in our notations when this does not lead to ambiguities. For example we can write $p(\underline{s}|\underline{y}) = p(\underline{s}|\underline{h})$ for the posterior. For the transition probability of the memoryless channel we have to be more careful. In terms of loglikelihood variable we denote it $\mathrm{c}(h_i|s_i)$, and formally

$$p(y_i|s_i)dy_i = \mathrm{c}(h_i|s_i)dh_i \tag{3.12}$$

[3] We will usually denote variable nodes by letters $i, j, k, \ldots$ and parity checks by $a, b, c, \ldots$

[4] In the coding theory literature (or in statistics) it is usual to define likelihood variable without the factor 1/2. Here the "1/2" yields a somewhat nicer connection to statistical mechanics quantities. The loglikelihood is analogous to a "magnetic field".

(by conservation of mass). The explicit expressions of $\mathrm{c}(h_i|s_i)$ for the BEC, BSC and BAWGNC are found in example 10.

The posterior (3.10) is a *random Gibbs distribution*, or in other words a *spin glass model*. Here the word "random" relates to the randomness of the channel outputs as well as the choice of code or factor graph. For each channel realization $\underline{h}$ and each code $\mathcal{C}$ picked from an ensemble (say Gallager's ensemble) we have a distribution over the spins $\underline{s} \in \{-1, +1\}^n$. In the terminology of physics the randomness associated with the code (or factor graph) and channel realisations is "quenched". This is because in a *given* experiment (here the transmission and reception of a message) the code and channel realisations are *fixed*, or *frozen*. The spins on the other hand are "annealed" degrees of freedom which fluctuate and adapt themselves in typical configurations.

What are the distributions of the quenched randomness? The distribution over the codes is the uniform distribution over Gallager's ensemble. For the configuration model introduced in Chapter 1 this is the uniform distribution over all permutations among nd_v sockets. Expectations with respect to codes of the ensemble are denoted $\mathbb{E}_{\mathcal{C}}$. The channel outputs are distributed according to $\mathrm{c}(\underline{h}|\underline{s}^{\mathrm{in}})$ when $\underline{s}_{\mathrm{in}}$ is the input codeword, and the correspondings expectations are denoted $\mathbb{E}_{\underline{h}|\underline{s}^{\mathrm{in}}}$. Averages over all quenched variables (both the code and channel outputs) are generically denoted by $\mathbb{E}$.

This is a good point to recall (see Sect. 2.6) that averages with respect to the Gibbs distribution are denoted by the bracket $\langle - \rangle$, and are distinguished from expectations $\mathbb{E}$ over quenched variables. These two averages *cannot* be exchanged.

What is the Hamiltonian corresponding to the Gibbs distribution (3.10)? To answer this question it is enough rewrite this expression as $e^{-\beta \mathcal{H}(\underline{s})}/Z_\beta$. If we set $\beta = 1$ we have

$$\mathcal{H}(\underline{s}) = \sum_{a=1}^{m} \ln \big\{ \frac{1}{2}(1 + \prod_{i \in \partial a} s_i) \big\} - \sum_{i=1}^{n} h_i s_i \tag{3.13}$$

Setting β to a different value would simply amount to scale the Hamiltonian by the inverse of that value. The Hamiltonian has two parts. The second part is a magnetic field term, where the magnetic fields h_i are i.i.d random variables distributed according to $\mathrm{c}(h_i|s_i^{\mathrm{in}})$. The first part is a sum of interactions between spins involved in a parity check. If a parity check is satisfied $\ln\{\dots\} = 0$ and there is no energy cost; if a parity check is violated $\ln\{\dots\} = +\infty$ and there is infinite energy cost. Equivalently, the interaction terms can also be represented as $J_a(1 - \prod_{i \in \partial a} s_i)$ with J_a formally equal to $+\infty$. Summarizing, the posterior distribution used in bit-wise MAP decoding can be thought of as a Gibbs distribution with inverse temperature set to the special value $\beta = 1$ and Hamiltonian of the general form (2.3).

Bit-MAP decoder and magnetization

The bit-MAP decoder has a natural relation to the magnetization of the spin glass. Indeed (3.5) is equivalent to

$$\widehat{s}_i^{\mathrm{MAP}}(\underline{h}) = \mathrm{sign}\big(\nu_i(s_i = 1|\underline{h}) - \nu_i(s_i = -1|\underline{h})\big) = \mathrm{sign}\langle s_i\rangle\,. \tag{3.14}$$

So the bit-MAP estimate for the i-th bit is given by the *sign* of the (local) magnetisation $\langle s_i\rangle$,

$$\begin{aligned}\langle s_i\rangle &= \frac{1}{Z}\sum_{\underline{s}} s_i \prod_{a=1}^{m}\frac{1}{2}(1+\prod_{i\in\partial a} s_i)\prod_{i=1}^{n} e^{h_i s_i}\\ &= \frac{\partial}{\partial h_i}\ln Z\end{aligned} \tag{3.15}$$

Using $\mathbb{P}[\widehat{s}_i^{\mathrm{MAP}}(\underline{h}) \neq s_i^{\mathrm{in}}] = \mathbb{E}_{\underline{h}|\underline{s}^{\mathrm{in}}}[\mathbb{1}(\widehat{s}_i(\underline{h}) \neq s_i^{\mathrm{in}})$ the average bit probability of error (3.4) becomes

$$P_b = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|\mathcal{C}|}\sum_{\underline{s}^{\mathrm{in}}\in\mathcal{C}}\frac{1}{2}\big(1-\mathbb{E}_{\underline{h}|\underline{s}^{\mathrm{in}}}\big[s_i^{\mathrm{in}}\mathrm{sign}(\langle s_i\rangle)\big]\big). \tag{3.16}$$

The BEC, BSC and BAWGNC have a special symmetry property which allows to simplify this expression. In the next section we show that for a general class of *symmetric channels* the terms in the sum (3.16) are independent of the input word. For such channels there is no loss in generality to assume that the transmitted word is $s_i^{\mathrm{in}} = 1$ for all $i = 1,\ldots,n$, or equivalently $\underline{x}^{\mathrm{in}} = 0$ the "all-zero codeword". In conclusion for the class of symmetric channels (and a given code) the average bit error probability is given by

$$P_b = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}\big(1-\mathbb{E}_{\underline{h}|\underline{1}}\big[\mathrm{sign}(\langle s_i\rangle)\big]\big). \tag{3.17}$$

where we recall that $\mathbb{E}_{\underline{h}|\underline{1}}$ is the expectation with respect to $\mathrm{c}(\underline{h}|\underline{s}^{\mathrm{in}} = \underline{1})$.

Interpolating between bit-MAP and block-MAP decoders

With Gibbs ddistributions in mind, it is natural to generalize the bit-MAP decoder by taking $\beta \neq 1$. More precisely consider replacing the posterior with $p_\beta(\underline{s}|\underline{h}) = e^{-\beta\mathcal{H}(\underline{s})}/Z_\beta$ where the Hamiltonian is still (3.13) but β is *general*,

$$p_\beta(\underline{s}|\underline{h}) = \frac{1}{Z_\beta}e^{-\beta\mathcal{H}(\underline{s})} = \frac{1}{Z_\beta}\prod_{a=1}^{m}\frac{1}{2}(1+\prod_{i\in\partial a} s_i)\prod_{i=1}^{n} e^{\beta h_i s_i} \tag{3.18}$$

with the partition function Z_β the sum over all $\underline{s}\in\{-1,+1\}^n$ of the numerator. The *general temperature decoder* is then defined from the marginals of (3.18),

$$\widehat{s}_i(\underline{h};\beta) = \mathrm{argmax}\,\nu_{i,\beta}(s_i|\underline{h}) = \mathrm{sgn}\langle s_i\rangle_\beta, \tag{3.19}$$

where the bracket $\langle - \rangle_\beta$ is the average with respect to (3.18). Obviously for $\beta = 1$ this is the bit-MAP decoder. Taking the limit $\beta \to +\infty$ it is not difficult to see that $\mathrm{sgn}\langle s_i \rangle_\beta \to \mathrm{argmin}\, \mathcal{H}(\underline{s})$ or equivalently $\mathrm{argmax}\, p(\underline{s}|\underline{h})$. Thus in the zero temperature limit we recover the block-MAP decoder. For $1 \leq \beta \leq +\infty$ the general temperature decoder interpolates between the bit-wise and block MAP decoders.

3.2 Channel symmetry and gauge transformations

A binary input channel is said to be *symmetric* when the transition probability satisfies $p(y_i|s_i) = p(-y_i| - s_i)$. From (3.9) and (3.12) we note

$$\frac{c(h|1)}{c(-h|1)} = \frac{p(y|1)}{p(-y|1)} = \frac{p(y|1)}{p(y|-1)} = e^{2h}$$

so that a symmetric channel can also be defined by the identity

$$\mathrm{c}(-h_i|1) = \mathrm{c}(h_i|1)e^{-2h_i}\,. \tag{3.20}$$

EXAMPLE 10 For the BEC, BSC, BAWGNC we check explicitly that $p(y_i|s_i) = p(-y_i| - s_i)$. One also computes $\mathrm{c}(h_i|1)$ and finds

$$\begin{aligned}
&\mathrm{c}(h|1) = (1-\epsilon)\delta_{+\infty}(h) + \epsilon\delta(h), \quad \mathrm{BEC}(\epsilon)\\
&\mathrm{c}(h|1) = (1-\epsilon)\delta(h - \ln\frac{1-\epsilon}{\epsilon}) + \epsilon\delta(h - \ln\frac{\epsilon}{1-\epsilon}), \quad \mathrm{BSC}(\epsilon)\\
&\mathrm{c}(h|1) = \frac{1}{\sqrt{2\pi\sigma^{-2}}} e^{-(h-\frac{1}{\sigma^2})^2/\frac{2}{\sigma^2}}, \quad \mathrm{BAWGNC}(\sigma^2)
\end{aligned}$$

The identity (3.20) is explicit on these expressions. □

Let us now prove identity (3.17). Consider first $\mathbb{E}_{\underline{h}|\underline{s}^{\mathrm{in}}}\big[s_i^{\mathrm{in}}\mathrm{sign}(\langle s_i\rangle)\big]$ in (3.16). The expectation $\mathbb{E}_{\underline{h}|\underline{s}^{\mathrm{in}}}$ is an integral over h_i's, and the bracket $\langle - \rangle$ contains sums (in a numerator and denominator) over s_i's. In the inetgrals and sums we may perform the change of variables

$$s_i \to \tau_i s_i, \qquad h_i \to \tau_i h_i, \qquad i = 1, \ldots, n \tag{3.21}$$

for any fixed code word $\underline{\tau} \in \mathcal{C}$. Now we note two crucial facts. First, under this transformation the posterior (3.10) remains *invariant*, and therefore $\langle s_i \rangle \to \tau_i \langle s_i \rangle$, where $\langle - \rangle$ is the *same* expectation on both sides of the equality. Second, because of channel symmetry $\mathbb{E}_{\tau_i h_i|s_i^{\mathrm{in}}} = \mathbb{E}_{h_i|\tau_i s_i^{\mathrm{in}}}$. Thus

$$\mathbb{E}_{\underline{h}|\underline{s}^{\mathrm{in}}}\big[s_i^{\mathrm{in}}\mathrm{sign}(\langle s_i\rangle)\big] = \mathbb{E}_{\underline{\tau}\star\underline{h}|\underline{s}^{\mathrm{in}}}[s_i^{\mathrm{in}}\tau_i\mathrm{sgn}\langle s_i\rangle] == \mathbb{E}_{\underline{h}|\underline{\tau}\star\underline{s}^{\mathrm{in}}}\big[s_i^{\mathrm{in}}\tau_i\mathrm{sign}(\langle s_i\rangle)\big] \tag{3.22}$$

where we used the notation $\underline{v}\star\underline{u}$ for a vector with components v_iu_i, $i = 1, \ldots, n$ ($\star$ is known as the "Hadamard product"). Now, we can choose $\underline{\tau} = \underline{s}^{\mathrm{in}}$ which implies $\mathbb{E}_{\underline{h}|\underline{s}^{\mathrm{in}}}\big[s_i^{\mathrm{in}}\mathrm{sign}(\langle s_i\rangle)\big] = \mathbb{E}_{\underline{h}|\underline{1}}\big[\mathrm{sign}(\langle s_i\rangle)\big]$ as well as (3.17).

The idea of using the transformation (3.21) turns out to be very useful in

the present framework. Since codewords $\underline{\tau} \in \mathcal{C}$ form a *group*,[5] the set of such transformations also forms a group. Moreover these transformations are *local* in the sense that for each i the variables get multiplied by different factors. Transformations with these two group and locality properties are ubiquitous in modern physics and are generally called *gauge transformations*. In the present context the invariance of the Gibbs distribution under gauge transformations can be traced back to channel symmetry. The use of such transformations allows to derive a number of useful consequences and identities. We will have the occasion to derive them as we proceed with the theory, but some of them can be found in the exercises. The independence of the error probability on the transmitted codeword, proved here, is one of them.

It is important to note that the invariance of the Gibbs distribution under gauge transformations is a consequence of the linearity of the code. For non-linear codes such an invariance would typically not be present. Also, for the random satisfiability problem where the constraints are "non-linear", and there is no obvious group structure, we do not have (or know) any useful gauge transformations. This is one of the reasons why this problem is a much harder one.

3.3 Conditional entropy and free energy in coding

Without loss of generality we assume from now on that the all all-zero codeword is transmitted. Thus we set $\underline{s}^{\text{in}} = \underline{1}$. The channel outputs are distributed as $p(\underline{y}|\underline{s}^{\text{in}} = \underline{1})$.

We explained in Chapter 2 that a lot can be learned from the free energy $-\frac{1}{n}\ln Z$ associated to the posterior (3.10) (recall $\beta = 1$ for bit-MAP decoding). For example differentiating with respect to h_i yields the magnetization $\langle s_i \rangle$ (see Equ. (3.15)). For spin glass models the free energy is random but concentrates in the thermodynamic limit $n \to +\infty$. Although this can be non-trivial to prove, we do have examples with effective proof techniques; these will be introduced in Chapter 13. We are therefore primarily interested in *average free energy* $-\frac{1}{n}\mathbb{E}[\ln Z]$ over the code ensemble and the channel outputs.

We will now show an important relation to the conditional entropy $H(\underline{X}|\underline{Y})$, i.e. the average entropy of the posterior $p(\underline{s}|\underline{y})$,

$$H(\underline{X}|\underline{Y}) = -\sum_{\underline{y}} p(\underline{y}|\underline{1}) \sum_{\underline{s}} p(\underline{s}|\underline{y}) \ln p(\underline{s}|\underline{y}) \tag{3.23}$$

This relation shows that computing the average free energy or the conditional entropy is basically equivalent. In part III we will develop powerful methods to compute the average free energy, and this will automatically allow us to compute

[5] In the bit language the group operation is of course the modulo two sum $\oplus$. In the spin langugae it is the Hadamard product $\star$.

the conditional entropy (averaged over the code ensemble) and in particular the MAP threshold of the code ensemble.[6]

For transmission over a symmetric channel and any fixed linear code (not necessarily an LDPC code) we have

$$\frac{1}{n}H(\underline{X}|\underline{Y}) = \frac{1}{n}\mathbb{E}_{\underline{h}|\underline{1}}[\ln Z] - \int_{-\infty}^{+\infty} dh\, \mathrm{c}(h|1)h. \tag{3.24}$$

Observe that this relation is valid for a *fixed* code. Of course it remains valid when we further average over the code ensemble.

The last term in (3.24) depends only on the channel. For the BSC it is equal to $(1-2\epsilon)\ln\frac{1-\epsilon}{\epsilon}$ and for the BAWGNC $1/\sigma^2$. For the BEC there is a little ambiguity here. Formally $\int_{-\infty}^{+\infty} dh\, \mathrm{c}(h|1)h$ is infinite, but this infinity is cancelled with another infinity in $\ln Z$. Indeed the weight factors $e^{h_i s_i}$ in Z diverge when $s_i = 1$ and $h_i = +\infty$. To proceed more neatly one should redefine the partition function replacing $e^{h_i s_i}$ by $e^{h_i s_i - h_i}$, so that the new Z is finite and the last term in (**??**) is not present. This must in principle be done for any channel having a non-zero weight on $h_i = +\infty$, but is more a nuisance than a real problem.

The proof of (3.24) is a good occasion to illustrate once a again the use of gauge transformations and channel symmetry. Replacing (3.10) in (3.23)

$$\begin{aligned} H(\underline{X}|\underline{Y}) &= \sum_{\underline{y}} p(\underline{y}|\underline{1}) \ln Z(\underline{y}) - \sum_{\underline{y}} p(\underline{y}|\underline{0}) \sum_{\underline{s}} p(\underline{s}|\underline{y}) \ln\Big\{ \prod_{a=1}^{m} \frac{1}{2}(1 + \prod_{i\in c} s_i) \Big\} \\ &\quad - \sum_{\underline{y}} p(\underline{y}|\underline{0}) \sum_{\underline{s}} p(\underline{s}|\underline{y}) \sum_{i=1}^{n} h_i s_i \\ &= \mathbb{E}_{\underline{h}|\underline{1}}[\ln Z] - \sum_{i=1}^{n} \mathbb{E}_{\underline{h}|\underline{1}}[h_i \langle s_i \rangle] \end{aligned} \tag{3.25}$$

To get the last equality we noticed that the second term vanishes because $p(\underline{s}|\underline{y})$ is supported on code words and $\ln(1) = 0$. Finally we replaced the expectation wity respect to $p(\underline{y}|\underline{1})$ by $\mathbb{E}_{\underline{h}|\underline{1}}$. To arrive at (3.24), it remains to show the identity

$$\mathbb{E}_{\underline{h}|\underline{1}}[h_i \langle s_i \rangle] = \mathbb{E}_{\underline{h}|\underline{1}}[h_i] \tag{3.26}$$

This is part of a whole class of relationships, called Nishimori identities, which follow from gauge invariance and channel symmetry. We will encounter a number of them in subsequent chapters. Using a gauge transformation $s_i \to \tau_i s_i$, $h_i \to \tau_i h_i$ and the channel symmetry in the form $\mathrm{c}(\tau_i h_i|1) = \mathrm{c}(h_i|1)e^{h_i\tau_i - h_i}$ we have

$$\begin{aligned} \mathbb{E}_{\underline{h}|\underline{1}}[h_i \langle s_i \rangle] &= \mathbb{E}_{\underline{\tau}\star\underline{h}|\underline{1}}[h_i \langle s_i \rangle] \\ &= \mathbb{E}_{\underline{h}\underline{1}}[h_i \langle s_i \rangle \prod_{j=1}^{n} e^{h_j \tau_j - h_j}] \end{aligned} \tag{3.27}$$

[6] Say that we will see this can be accessed from entropy or error prob.

Summing over all code words $\underline{\tau} \in \mathcal{C}$,

$$\begin{aligned}\mathbb{E}_{\underline{h}1}[h_i\langle s_i\rangle] &= \frac{1}{|\mathcal{C}|}\mathbb{E}_{\underline{h}|1}\Big[h_i\langle s_i\rangle\, Z\prod_{j=1}^{n} e^{-h_j}\Big] \\ &= \frac{1}{|\mathcal{C}|}\mathbb{E}_{\underline{h}|1}\Big[h_i\sum_{\underline{s}} s_i\prod_{a=1}^{m}\frac{1}{2}(1+\prod_{i\in\partial a} s_i)\prod_{j=1}^{n} e^{h_j s_j - h_j}\Big] \\ &= \frac{1}{|\mathcal{C}|}\sum_{\underline{s}}\Big\{s_i\prod_{a=1}^{m}\frac{1}{2}(1+\prod_{i\in\partial a} s_i)\mathbb{E}_{\underline{h}|1}\Big[h_i\prod_{j=1}^{n} e^{h_j s_j - h_j}\Big]\Big\} \\ &= \frac{1}{|\mathcal{C}|}\sum_{\underline{s}}\Big\{s_i\prod_{a=1}^{m}\frac{1}{2}(1+\prod_{i\in\partial a} s_i)\mathbb{E}_{h_i|1}\Big[h_i e^{h_i s_i - h_i}\Big]\prod_{j\neq i}\mathbb{E}_{h_j|1}\Big[e^{h_j s_j - h_j}\Big]\Big\}\end{aligned} \tag{3.28}$$

The identity (3.26) then follows from

$$\mathbb{E}_{h_i|1}[e^{h_i s_i - h_i}] = 1, \qquad \mathbb{E}_{h_i|1}[h_i e^{h_i s_i - h_i}] = s_i\mathbb{E}_{h_i|1}[h_i] \tag{3.29}$$

and $\sum_{\underline{s}}\prod_{a=1}^{m}\frac{1}{2}(1+\prod_{i\in\partial a} s_i) = |\mathcal{C}|$. Formulas (3.29) are trivial when $s_i = 1$. For $s_i = -1$ that they follow from channel symmetry, $\mathrm{c}(-h_i|1) = \mathrm{c}(h_i|1)e^{-2h_i}$.

3.4 Compressive Sensing as a spin glass model

Recall that we are considering the model

$$\underline{y} = A\underline{x} + \underline{z}, \tag{3.30}$$

where the measurement matrix A is an $m \times n$ real valued matrix with i.i.d zero mean Gaussian entries with variance $1/m$, the noise $\underline{z} = (z_1, \ldots, z_m)$ consists of m i.i.d zero-mean Gaussian entries of variance σ^2, and the signal $\underline{x} = (x_1, \ldots, x_n)$ consists also of n i.i.d entries distributed with the prior $p_0(x)$. We will assume this prior belongs to the *sparse* class $\mathcal{F}_\kappa$, that is

$$p_0(x) = (1-\kappa)\delta(x) + \kappa\phi_0(x) \tag{3.31}$$

where $0 < \kappa < 1$ and ϕ_0 is a continuous positive and normalized density. The expected number of non-zero entries in the signal is $k = \kappa n$.

The conditional probability of observing $\underline{y}$ given $\underline{x}$ is

$$p(\underline{y} \mid \underline{x}) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{1}{2\sigma^2}\|\underline{y} - A\underline{x}\|_2^2}, \tag{3.32}$$

and the joint distribution, taking the prior into account, has the form

$$p(\underline{x}, \underline{y}) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{1}{2\sigma^2}\|\underline{y} - A\underline{x}\|_2^2}\prod_{i=1}^{n} p_0(x_i)\,. \tag{3.33}$$

We discuss two scenarios. In the first one the prior is *known*, i.e., ϕ_0 is known, and in the second scenario which is more realistic the prior is *not known* but one

knows that it belongs to the sparse class $\mathcal{F}_\kappa$. In other words κ is assumed to be known but not ϕ_0. Furthermore we have in mind the regime of $n, m \to +\infty$ with $\kappa = k/n$ and $\mu = m/n$ fixed.

Known prior: MMSE estimator

When the prior is known a reasonable way to estimate the signal is to use the *minimum mean square estimator* (MMSE) estimator. This estimator is optimal in the sense that it minimizes the *mean square error* (MSE). The MSE is a functional

$$\mathrm{MSE}(\underline{\widehat{x}}(\cdot)) = \mathbb{E}[(\underline{\widehat{x}}(\underline{Y}) - \underline{X})^2] \tag{3.34}$$

defined over the "space" of estimators $\underline{\widehat{x}}(\underline{y}) : \mathbb{R}^m \to \mathbb{R}^n$. The expectation is with respect to the joint distribution (3.33) and the entries of A. A standard exercise shows that the minimum is attained by the MMSE estimator,[7]

$$\widehat{x}_i^{\mathrm{MMSE}}(\underline{y}) = \mathbb{E}_{\underline{X}|\underline{y}}[X_i] = \int d^n\underline{x}\, x_i\, p(\underline{x} \mid \underline{y}), \quad i = 1, \dots, n, \tag{3.35}$$

where

$$p(\underline{x}|\underline{y}) = \frac{p(\underline{x}, \underline{y})}{\int d^n\underline{x}\, p(\underline{x}, \underline{y})} \tag{3.36}$$

is the posterior associated to (3.33). This estimator makes explicit use of the prior $p_0(x)$. Analogously to the case of coding, we will shortly interpret the posterior as a (random) Gibbs distribution and the MMSE estimator as a "magnetization".

Unknown prior: LASSO estimator

A popular choice for the estimator is the LASSO (least absolute shrinkage selection operator)

$$\underline{\widehat{x}}^{\mathrm{LASSO}}(\underline{y}) = \mathrm{argmin}_{\underline{x}} \left\{ \frac{1}{2}\|\underline{y} - A\underline{x}\|_2^2 + \lambda\|\underline{x}\|_1 \right\}, \tag{3.37}$$

where the real parameter λ has to be chosen suitably. Since the prior is *unknown* natural guiding principle is to choose the best possible λ for the worse possible prior. Formally we have to compute the so-called *minimax risk*,

$$\inf_{\lambda>0} \sup_{p_0 \in \mathcal{F}_\kappa} \mathbb{E}[(\underline{\widehat{x}}^{\mathrm{LASSO}}(\underline{Y}) - \underline{X})^2] \tag{3.38}$$

The expectation is again (like in (3.34)) over the joint distribution (3.33) and the random matrix ensemble. Computing the minimax risk allows to fix λ in a principled way, and the numerical value of the risk constitutes a reasonable performance measure.

As explained in Chapter 1 it is not so easy to unambiguously justify a priori

[7] We use $d^n\underline{u} = \prod_{i=1}^r u_i$ for any vector $\underline{u} = (u_1, \dots, u_r)$.

the choice of the LASSO. For one thing, it has remarkable properties. In fact it turns out that the variational problem (3.38) can be solved analytically exactly. Quite remarkably we will find that, independently of the noise level, the minimax risk is finite in the same region of parameters (κ, μ) for which ℓ_0-ℓ_1 equivalence holds. At the boundary of this region the minimax risk diverges. This boundary has been called the Donoho-Tanner curve and will be derived in Chapter 8.

We will shortly give a different, somewhat more phenomenological justification for using LASSO which does not require to wait and develop the whole theory, and stems more or less naturally from the statistical mechanics interpretation.

MMSE and LASSO as spin glass models

The posterior (3.36) used for MMSE estimation is explicitly,

$$p(\underline{x} \mid \underline{y}) = \frac{1}{Z} \prod_{a=1}^{m} e^{-\frac{1}{2\sigma^2}(y_a - \underline{A}_a \cdot \underline{x})^2} \prod_{i=1}^{n} p_0(x_i), \tag{3.39}$$

where y_a, $a = 1, \ldots, m$ are the components of $\underline{y}$ and $\underline{A}_a$ the *line* vector equal to the a-th row of the matrix A. In other words $\underline{A}_a \cdot \underline{x} = \sum_{i=1}^{n} A_{ai} x_i$. The normalisation factor is given by

$$Z = \int d^n \underline{x} \prod_{a=1}^{m} e^{-\frac{1}{2\sigma^2}(y_a - \underline{A}_a \cdot \underline{x})^2} \prod_{i=1}^{n} p_0(x_i)\,. \tag{3.40}$$

The interpretations in terms of spin-glass concepts are analogous to the case of coding. The posterior (3.39) can be though of as a random Gibbs distribution and (3.40) as a partion function. This time the "spin variables" $x_i \in \mathbb{R}$ belong to a continuous alphabet, and one often speaks of "continuous spins". The distribution is random because of the measurement matrix A and the observations $\underline{y}$. These are the quenched variables.

The estimator (3.35) is the average of x_i with respect to the Gibbs distribution and in statistical mechanics notation is written as the bracket $\langle x_i \rangle$. One can interpret it as a "magnetization" for the continuous spins. In order to compute it all we need in principle is the marginal $p(x_i|\underline{y})$ given by integrating (3.39) over all spin variables except x_i. To sum up we have,

$$\widehat{x}_i^{\mathrm{MMSE}}(\underline{y}) = \langle x_i \rangle = \int d^n \underline{x}\, x_i\, p(\underline{x} \mid \underline{y}) = \int dx_i\, x_i\, p(x_i \mid \underline{y}), \tag{3.41}$$

What are the Hamiltonian and the inverse temperature associated to the Gibbs distribution in the present context? Writing (3.39) in the form $e^{-\beta \mathcal{H}(\underline{x})}/Z_\beta$ we find that a natural answer is to take

$$\mathcal{H}(\underline{x}) = \frac{1}{2\sigma^2} \sum_{a=1}^{m} (y_a - \underline{A}_a \cdot \underline{x})^2 - \sum_{i=1}^{n} \ln p_0(x_i) \tag{3.42}$$

and $\beta = 1$ (just as in coding any other other value of β would amount to rescale the Hamiltonian by the inverse of that value).

In coding we discussed a "finite temperature decoder" and noticed that it interpolates between the bit-MAP and block-MAP decoders. With the Hamiltonian view it is immediate to do something similar here. Let us define for any $\beta > 0$,

$$p_\beta(\underline{x}|\underline{y}) = \frac{1}{Z_\beta} e^{-\beta \mathcal{H}(\underline{x})} = \frac{1}{Z_\beta} \prod_{a=1}^{m} e^{-\frac{\beta}{2\sigma^2}(y_a - \underline{A}_a \cdot \underline{x})^2} \prod_{i=1}^{n} (p_0(x_i))^\beta \tag{3.43}$$

with Z_β the corresponding normalization factor given by the integral over all x_i's of the numerator. We define a βMMSE estimator as the magnetization at inverse temperature β,

$$\widehat{x}_i^{\beta\text{MMSE}}(\underline{y}) = \langle x_i \rangle_\beta = \int d^n \underline{x}\, x_i\, p_\beta(\underline{x} \mid \underline{y}) = \int dx_i\, x_i\, p_\beta(x_i \mid \underline{y}). \tag{3.44}$$

For $\beta = 1$ this is simply the usual MMSE estimator. In the limit of zero temperature, $\beta \to +\infty$, the integral over $d^n \underline{x}$ is concentrated on the spin configurations that minimize the Hamiltonian, in other words

$$\begin{aligned} \lim_{\beta \to +\infty} \widehat{x}_i^{\beta\text{MMSE}}(\underline{y}) &= \operatorname{argmin}_{\underline{x}} \mathcal{H}(\underline{x}) \\ &= \operatorname{argmin}_{\underline{x}} \Big(\frac{1}{2} \|\underline{y} - A\underline{x}\|_2 - \sigma^2 \sum_{i=1}^{n} \ln p_0(x_i) \Big) . \end{aligned} \tag{3.45}$$

Here we minimise a quantity closely analogous to the usual least square estimator but penalized by a term $-\ln p_0(x)$ which enforces the prior information.

Now we can formulate another justification for the LASSO. When the prior is unknown but it is only known that the signal is sparse the Laplacian prior $p_0(x) \propto e^{-\frac{\lambda}{\sigma^2}|x|}$ is a simple, and as it turns out, *tractable model* for the ensemble of possible priors. This ensemble is parametrized by a single parameter λ. As discussed before, the optimal value of λ (as a function of κ and μ) is determined from the minimax principle. In a sense, this point of view naturally leads to the AMP algorithm developed in Chapter 8. Summarizing, the LASSO can be viewed as a zero temperature limit of MMSE estimation generalized to arbitrary temperatures.

3.5 Free energy and conditional entropy in compressive sensing

In this paragraph we assume that the prior is *known* and consider the Gibbs distribution associated to the usual MMSE estimator (with $\beta = 1$). We show that the average free energy $-\mathbb{E}_{\underline{Y}}[\ln Z]/n$ (for any fixed measurement matrix A) and mutual information $I(\underline{X};\underline{Y})$ are essentially one and the same object. This relation is analogous to the one found for coding in section 3.3. There is one technical difference: because the distribution of $\underline{X}$ has a continuous part it is more convenient to work directly with the mutual information rather than conditional entropy in order to avoid pitfalls related to the differential entropy.

Consider the mutual information for a fixed measurement matrix,[8]

$$I(\underline{X};\underline{Y}) = \mathbb{E}_{\underline{Y}}\Big[\ln\Big\{\frac{p(\underline{x},\underline{y})}{p_0(\underline{x})p(\underline{y})}\Big\}\Big] \tag{3.46}$$

where $\mathbb{E}_{\underline{Y}}$ is the expectation with respect to $p(\underline{y}) = \int d^n\underline{x}\, p(\underline{x},\underline{y})$. From (3.33) we find

$$\ln\Big\{\frac{p(\underline{x},\underline{y})}{p_0(\underline{x})p(\underline{y})}\Big\} = -\frac{1}{2\sigma^2}\|\underline{y} - A\underline{x}\|_2^2 - \ln Z(\underline{y}) \tag{3.47}$$

The last term contributes $-\mathbb{E}_{\underline{Y}}[\ln Z]$ to the mutual information. To derive the contribution of the first term we write down explicitly the integrals,

$$\begin{aligned}\frac{1}{2\sigma^2}\int d^n\underline{x}\int d^m\underline{y}\, p(\underline{x},\underline{y})\|\underline{y} - A\underline{x}\|_2^2 \\ &= \frac{1}{2\sigma^2}\int\prod_{i=1}^n dx_i p_0(x_i)\int d\underline{y}\,\|y\|_2^2\frac{e^{-\frac{1}{2\sigma^2}\|\underline{y}\|_2^2}}{(2\pi\sigma^2)^{n/2}} \\ &= \frac{n}{2}.\end{aligned} \tag{3.48}$$

(The second line is obtained by a shift $\underline{y} \to \underline{y} + A\underline{x}$ in the $\underline{y}$-integral for each fixed $\underline{x}$). Putting everything together we find the very simple relationship

$$I(\underline{X};\underline{Y}) = -\mathbb{E}_{\underline{Y}}[\ln Z] - \frac{n}{2} \tag{3.49}$$

This relation can be further averaged over the random matrix ensemble.

3.6 Random K-SAT as a spin glass model

We briefly recall the formulation of the random satisfiability in Section 1.3. Pick a formula at random from the ensemble $\mathcal{F}(n,m,K)$; the formula corresponds to a biparttite factor graph with dashed and full edges (see Fig. 1.6). As for coding and compressed sensing we shall adopt the notation $i,j,k,\ldots$ for variable nodes and $a,b,c,\ldots$ for constraint nodes. In the random max-K-SAT problem our main task is to calculate the average minimum fraction of violated clauses. More precisely, consider the number of violated clauses for an assignment $\underline{x}$, then take the best possible assignment that minimizes the number of violated clauses and average over the ensemble of random formulas. This yields the average minimum fraction of violated clauses

$$e_m(\alpha) = \frac{1}{m}\mathbb{E}\big[\min_{\underline{x}}\sum_{a=1}^m(1 - \mathbb{1}_a(\underline{x}))\big]. \tag{3.50}$$

where $\mathbb{1}_a$ is the indicator function for the set of assignments which satisfy clause a. In Chapter 13 we prove that the thermodynamic limit, $\lim_{m\to+\infty} e_m(\alpha)$ with

[8] This is also the Kullback-Leibler divergence or relative entropy between $p(\underline{x},\underline{y})$ and $p_0(\underline{x})p(\underline{y})$.

$m/n = \alpha$ fixed, exists and is calculated using the cavity method developped in Chapters 16 and 17. This leads to a prediction for the threshold of the random max-K-SAT problem, $\alpha_{s,\max}(K)$ defined in Equ. 1.16. As explained there it is believed (although not proved for small $K \geq 3$) that $\alpha_{s,\max}(K) = \alpha_s(K)$ and we will generally not distinguish between the two.

The problem here is not directly formulated in terms of a Gibbs distribution, but a natural and fruitful idea is to consider the Gibbs distribution associated to the cost function

$$\sum_{a=1}^{m}(1 - \mathbb{1}_a(\underline{x})). \tag{3.51}$$

In particular, by studying the Gibbs distribution for very low temperatures we can get hold of $e_m(\alpha)$ and also much more. But before, it is instructive to look more closely at the zero temperature purely Hamiltonian formulation.

Hamiltonian formulation in the spin representation

We set $s_i = (-1)^{x_i}$. Furthermore if clause a contains the literal x_i (resp. $\bar{x}_i$) we associate a weight $J_{ai} = +1$ (resp. $J_{ai} = -1$) to the edge ai of the factor graph. Thus, for example on Fig. 1.6 full edges have $J_{ai} = +1$ and dashed edges have $J_{ai} = -1$. Moreover the J_{ai} are Bernoulli(1/2) random variables. With these conventions we see that the i-th variable satisfies clause a when $s_i = (-1)^{x_i} = -J_{ai}$ and does not satisfy it when $s_i = (-1)^{x_i} = +J_{ai}$. Therefore

$$1 - \mathbb{1}_a(\underline{x}) = \prod_{i \in \partial a} \frac{1}{2}(1 + s_i J_{ia}) \tag{3.52}$$

and the cost function, or Hamiltonian, of K-SAT takes the form

$$\mathcal{H}(\underline{s}) = \sum_{a=1}^{m} \prod_{i \in \partial a} \frac{1}{2}(1 + s_i J_{ia})\,. \tag{3.53}$$

Expanding the product in each term we see that this Hamiltonian involves "multispin interactions" of the form (2.3). This Hamiltonian is random in the sense that the underlying factor graph is random, and furthermore this randomness is quenched because once the formula has been chosen from the ensemble we stick to it. In terms of this spin-glass Hamiltonian the average minimum fraction of violated clauses (3.50) is expressed as

$$e_m(\alpha) = \frac{1}{m}\mathbb{E}\big[\min_{\underline{s}} \mathcal{H}(\underline{s})\big]. \tag{3.54}$$

The spin assignments that minimize the Hamiltonian (3.53) are often called *ground states*. Ground states with zero energy (or zero cost) satisfy $\mathcal{H}(\underline{s}) = 0$ and a look at (3.53) shows that they are solutions of the K-sat formula. Of course the converse is also true. An important problem is to count them, which amounts to

evaluate

$$\mathcal{N}_0 = \sum_{\underline{s}} \mathbb{1}(\mathcal{H}(\underline{s}) = 0) = \sum_{\underline{s}} \prod_{a=1}^{m} \Big(1 - \prod_{i \in \partial a} \frac{1}{2}(1 + s_i J_{ia}) \Big) . \tag{3.55}$$

It is often useful to take a broader view and count the number of spin assignment of total energy (or cost) E, in other words those that violate E clauses,

$$\mathcal{N}_E = \sum_{\underline{s}} \mathbb{1}(\mathcal{H}(\underline{s}) = E) . \tag{3.56}$$

At this point there is a natural connection with the notions of Boltzmann entropy and microcanonical distributions introduced in Section 2.7. The Boltzmann entropy here is equal to $\ln \mathcal{N}_E$ and the microcanonical distribution is $\mathbb{1}(\mathcal{H}(\underline{s}) = E)/\mathcal{N}_E$. There is however a difficulty with these concepts for constraint satisfaction problems such as K-SAT. Indeed a formula sampled from the ensemble $\mathcal{F}(n, m, \alpha)$ might not have any assignments which violate E clauses, in which case $\mathcal{N}_E = 0$. Even worse for large enough densities we have $\min_{\underline{s}} \mathcal{H}(\underline{s}) > E$ with a probability approaching 1 as $n, m \to +\infty$ with α fixed. One of the main advantages of the finite temperature formulation discussed in the next paragraph is that we can work with quantities that remain well defined for all formulas and densities from the outset.

Finite temperature formulation

The set of solutions of a K-sat formula, equivalently the set of ground states, is not easy to determine. One way to approach this problem would be to sample from this space according to a simple distribution. The simplest distribution one could imagine is the uniform one over solutions $\mathbb{1}(\mathcal{H}(\underline{s}) = 0)/\mathcal{N}_0$ (this is the microcanonical distribution for zero energy). We immediately face a problem here because, as just remarked above, some formulas from $\mathcal{F}(n, m, K)$ do not have any solution (and for high enough α this happens with overwhelming probability when n is large) so the uniform distribution is not well defined.

From the point of view of statistical mechanics there is a very natural regularisation of the uniform distribution. Namely one takes the (random) Gibbs distribution at positive temperature $\beta^{-1} > 0$,

$$p_\beta(\underline{s}) = \frac{1}{Z_\beta} e^{-\beta \mathcal{H}(\underline{s})} = \frac{1}{Z_\beta} \prod_{a=1}^{m} e^{-\beta \prod_{i \in \partial a} \frac{1}{2}(1 + s_i J_{ia})} \tag{3.57}$$

with the partition function

$$Z_\beta = \sum_{\underline{s}} \prod_{a=1}^{m} e^{-\beta \prod_{i \in \partial a} \frac{1}{2}(1 + s_i J_{ia})} . \tag{3.58}$$

In the zero temperature limit, formally at least, $p_\beta(\underline{s}) \to \mathbb{1}(\mathcal{H}(\underline{s}) = 0)/\mathcal{N}_0$.

We saw in Section 2.3 that from the free energy $F(\beta) = -\frac{1}{\beta} \ln Z$ at finite

temperature, we can compute the internal energy $\mathcal{E}(\beta) = \langle \mathcal{H}(\underline{s}) \rangle = \frac{d}{d\beta}(\beta F(\beta))$, and also the Gibbs entropy $S(\beta) = \beta(\mathcal{E}(\beta) - F(\beta)) = \frac{d}{d\beta^{-1}} F(\beta)$. To obtain ground state energy, (3.50) or (3.54), and deduce the threshold $\alpha_{s,\max}(K)$ we look at the zero temperature limit of the average energy per clause or of the average free energy per clause,

$$e_m(\alpha) = \lim_{\beta \to +\infty} \frac{1}{m} \mathbb{E}[\mathcal{E}(\beta)] = \lim_{\beta \to +\infty} \frac{1}{m} \mathbb{E}[F(\beta)]. \tag{3.59}$$

The proof of (3.59) does not involve any subtlety because $\mathbb{E}$ is an average over a finite number of possible formulas in $\mathcal{F}(n, m, K)$ and the statistical sums over $\underline{s}$ also involve a finite number of terms, so essentially the limit can be taken term by term. Similarly the zero temperature limit of the entropy is

$$\lim_{\beta \to +\infty} \frac{1}{n} \mathbb{E}[S(\beta)] = \frac{1}{n} \mathbb{E}[\ln \mathcal{N}_{\min_{\underline{s}} \mathcal{H}(\underline{s})}] \,. \tag{3.60}$$

Note that there is always at least one minimiser for finite n so $\mathcal{N}_{\min_{\underline{s}} \mathcal{H}(\underline{s})} \geq 1$ and this entropy is non-negative. In the satisfiable phase, i.e., for clause densities below the satisfiability threshold with overwhelming probability (for n large) $\min_{\underline{s}} \mathcal{H}(\underline{s}) = 0$ (and $\mathcal{N}_0 > 0$), so (3.60) approaches $\mathbb{E}[\ln \mathcal{N}_0 | F \text{ is satisfiable}]/n$ for n large.

In writing the above formulas we have swept an important point under the rug. In practice we are able to compute the free energy and related quantities only in the asymptotic regime of the thermodynamic limit. So a priori we have access to the zero temperature quantities only after the limit $n \to +\infty$ is taken. So in principle in order to use the finite temperature regularisation one must show that the limits $\beta \to +\infty$ and $n \to +\infty$ can be exchanged. This is not so obvious but can be shown at the level of (3.59) by the interpolation methods developped in Chapter 13.

3.7 Notes

The connection between coding and spin glasses dates back to (Sourlas 1989). It was slowly developed in a few early papers proposing the finite temperature decoder (Ruján 1993, Nishimori 1993, Sourlas 1994) and in (Amic & Luck 1995) which treated convolutional codes as a spin system. A surge of interest then appeared in the statistical physics community in the early 2000's soon after LDPC and turbocodes codes had come in the forefront among coding theorists. A good review with many references discussing LDPC and turbocodes from the spin glass perspective is (Kabashima & Saad 2004). Here we have followed closely the presentation in (Macris 2007*a*). Gauge symmetry in spin glass theory was introduced by (Nishimori 1980) and a good account, as well as applications to error correcting codes, is found in the monograph (Nishimori 2001).

The interest of statistical physicists towards the random K-SAT problem date

back to the mid 90's. In (Kirkpatrick & Selman 1994) the phase transition was observed based on numerical experiments, and the first analytical treatments using the replica method were proposed by (Monasson & Zecchina 1997, Monasson & al 1999). These early approaches can be found in the monograph (Nishimori 2001) but the correct phase diagram was then derived in a long series of subsequent works. A comprehensive treatment is found in (Mézard & Montanari 2009) and we return to this ongoing subject in part III where the reader can also find more details on the literature.

Compressive sensing in a more recent topic. Its connection with statistical physics developed quite quickly, and a good review is (Montanari 2012). Similar relationships have also been studied in the realm of communication with a "code division multiple access channel" with binary inputs (e.g. Nishimori 2001, Tanaka 2002, Macris & Korada 2010).

Problems

3.1 Nishimori identities for coding. Use the technique of gauge transformations to prove the identities $\mathbb{E}_{\underline{h}|\underline{1}}[\langle s_i\rangle^{2p-1}] = \mathbb{E}_{\underline{h}|\underline{1}}[\langle s_i\rangle^{2p}]$ for all integers $p \geq 1$. These identities are valid for any fixed linear code.

3.2 Special identities for a Gaussian channel. In the case of a BAWGNC(σ^2) identity (3.26) specializes to $\mathbb{E}_{\underline{h}|\underline{1}}[h_i\langle s_i\rangle] = \sigma^{-2}$. We propose a different proof that is specific to the Gaussian channel.

(i) First check by explicit calculation that $\sigma^2 \mathrm{c}(h)h = -\frac{\partial}{\partial h}\mathrm{c}(h) + \mathrm{c}(h)$.

(ii) Then use integration by parts and the Nishimori identity of the previous exercise (for $p = 1$) to derive $\mathbb{E}_{\underline{h}|\underline{1}}[h_i\langle s_i\rangle] = \sigma^{-2}$.

3.3 Relation between mutual information and average free energy in coding. We explore a different derivation of relation (3.24) which directly uses the mutual information and avoids the Nishimori identities. Relate the entropy $H(\underline{Y})$ directly to the average free energy $\mathbb{E}_{\underline{h}|\underline{1}}[\ln Z]$ and deduce a relation between the mutual information $I(\underline{X};\underline{Y}) = H(\underline{Y}) - H(\underline{Y}|\underline{X})$ and the average free energy. Use then $I(\underline{X};\underline{Y}) = H(\underline{X}) - H(\underline{X}|\underline{Y})$ to recover (3.24).

3.4 MMSE estimation. Consider the MSE functional (3.34) and prove that (3.35) is a minimiser.

3.5 LASSO for the scalar case. Let x and y scalar variables interpreted as "signal" and "measurement". Derive the explicitly expression of the LASSO estimator $\widehat{x}^{\mathrm{LASSO}}(y) = \mathrm{argmin}_x(\frac{1}{2}(y-x)^2 + \lambda|x|)$. The result is called the "soft thresholding estimator" and is an important input in the analysis of the vector case in Chapter 8.

3.6 Crude upper bound on the satisfiability threshold $\alpha_s(K)$. Sample a formula F from the random ensemble $\mathcal{F}(n, K, M)$ and consider the number of solutions $\mathcal{N}_0$ (the partition function at zero temperature),

(i) Prove the Markov inequality $\mathbb{P}[F \text{ is satisfiable}] \leq \mathbb{E}[\mathcal{N}_0]$.

(ii) From Equ. (3.55), show that $\mathbb{E}[\mathcal{N}_0] = 2^n(1-2^{-K})^m$.

(iii) Deduce the upper bound $\alpha_s(K) < (\ln 2)/|\ln(1-2^{-K})|$. For $K = 3$ this yields $\alpha_s(3) < 5.191$. It is conjectured that $\alpha_s(3) \approx 4.266$; a value calculated using a sophisticated form of the cavity method (see Chapter 16). The asymptotic behavior of this simple upper bound for $K \to +\infty$ is $2^K \ln 2$, which is known to be correct. However, even the first correction beyond the leading order is false.

4 Two Exactly Solvable Models

Before we start analysing our three basic problems, it is instructive to consider two very simple models for which the analysis can be carried out explicitly with fairly little effort. Despite their simplicity, both models share similarities with our three problems, at the technical as well as conceptual levels. This way we will encounter many useful concepts in their simplest incarnation.

We first consider the *Curie-Weiss* model. This is an Ising spin system defined on a *complete graph* (example 4 in Chapter 2). The model is admittedly special, but it has two advantages. First, it has a simple explicit solution. Secondly, and equally important, it still displays many of the interesting features of more complicated models such as variational expressions for the free energy, fixed point equations, and phase transitions.

The second exactly solvable model is the *Ising model on a tree* (example 5 in Chapter 2). We will see that the solution can be phrased in terms of another of our favourite themes, namely "iterative equations." We will also see that the model displays phase transitions similar to those of the Curie-Weiss model, despite the apparent differences in their Hamiltonians. This will throw some light on the universality of the phase transition phenomenon.

Analogous, but more complicated solutions and phase transitions occur in coding, compressive sensing and satisfiability models. It is perhaps natural that these models share some common features with the ones of this chapter. Indeed, coding and satisfiability are defined on sparse graphs that are *locally* tree like, and ompressed sensing is defined on a (bipartite) complete graph. On the other hand the situation is also considerably more complicated and interesting for at least two reasons. One is that in coding and satisfiablity the graphs are locally tree like but have loops. In satisfiability, for example, we will see that the presence of loops induces phase transitions not present on a tree. A second reason is that the Gibbs distributions are random and non-trivial spin glass concepts will come to bear on their analysis.

4.1 Curie-Weiss model

The Curie-Weiss model is an Ising spin system defined on a complete graph with vertex set $V = \{1, \ldots, n\}$ and edge set E constituted of all $n(n-1)/2$ pairs of

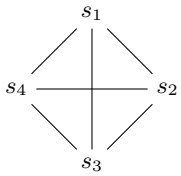


Figure 4.1 A complete graph with 4 nodes.

distinct vertices. An example is shown in Figure 4.1. The Hamiltonian is

$$\mathcal{H}(\underline{s}) = -\frac{J}{n} \sum_{\{i,j\}\in E} s_i s_j - h \sum_{i\in V} s_i \tag{4.1}$$

where $J > 0$ (ferromagnetic case) and $h \in \mathbb{R}$. In the first sum $\{i, j\}$ is an unordered pair so each edge is counted only once. Note that the interaction constant is divided by n to ensure that both terms in the Hamiltonian scale *linearly* in the system size; this is necessary in order to have an interesting thermodynamic limit.

The Gibbs distribution has the usual form $p(\underline{s}) = e^{-\beta\mathcal{H}(\underline{s})}/Z$ with the partition function given by the sum over all spin configurations $\underline{s} \in \{-1, +1\}^n$

$$Z = \sum_{\underline{s}} e^{\frac{\beta J}{n} \sum_{\{i,j\}\in E} s_i s_j + \beta h \sum_{i\in V} s_i}. \tag{4.2}$$

Recall from Chapter 2, $\beta = 1/k_B T$ where T is the temperature and k_B Boltzman's constant, so the behaviour of the Gibbs distribution depends on the dimensionless ratios $J/k_B T = \beta J$ and $h/k_B T = \beta h$.[1] For example, if we take $h = 0$ for simplicity, at high temperatures $k_B T \gg J$ (or $\beta J \ll 1$) we get an almost uniform measure, whereas in the low temperature case $k_B T \ll J$ (or $\beta J \gg 1$) only configurations of minimum energy, where most spins are aligned, count. Not surprisingly, we will see that $k_B T \approx J$ (or $\beta J \approx 1$) is a regime of great interest.

We will first calculate the free energy and then the magnetization. This will allow us to study the singularities of these functions, in other words the phase transitions displayed by the model.

4.2 Variational expression of the free energy

Recall that the free energy in the thermodynamic limit is given by

$$f(\beta, J, h) = -\lim_{n\to+\infty} \frac{1}{n\beta} \ln Z, \tag{4.3}$$

[1] One often defines dimensionless parameters $K = \beta J$, $H = \beta h$, however here we find it convenient to keep the β, J, h dependence as explicit as possible.

thus our main task is to compute the exponential growth rate of Z when $n \to +\infty$. The key point is the identity

$$\sum_{\{i,j\}\in E} s_i s_j = \frac{1}{2}\big(\sum_{i=1}^{n} s_i\big)^2 - \frac{1}{2}n. \tag{4.4}$$

which is valid because the sum over the edges of the complete graph carries over all $n(n-1)/2$ pairs of distinct vertices. Introducing the "magnetization of a spin configuration"

$$m_n(\underline{s}) = \frac{1}{n}\sum_{i=1}^{n} s_i,$$

with the help of (4.4) we can express the Hamiltonian as

$$\mathcal{H}(\underline{s}) = -n\Big(\frac{J}{2}\big(m_n(\underline{s})^2 + h m_n(\underline{s})\big) + \frac{J}{2}\Big). \tag{4.5}$$

Thus

$$Z = e^{-\frac{\beta J}{2}} \sum_{\underline{s}} e^{n\beta\left(\frac{J}{2}m_n(\underline{s})^2 + h m_n(\underline{s})\right)}. \tag{4.6}$$

The partition function can be computed by first summing over all spin configurations with a fixed magnetization m_n and then by summing over all possible magnetizations $m_n \in \{\frac{j}{n} | j = -n, -n+1, \ldots, n-1, n\}$. We get

$$Z = e^{-\frac{\beta J}{2}} \sum_{m_n} \mathcal{N}(m_n)\, e^{n\beta\left(\frac{J}{2}m_n^2 + h m_n\right)}. \tag{4.7}$$

where $\mathcal{N}(m_n)$ is the cardinality of the set $\{\underline{s} : \sum_{i=1}^{n} s_i = n m_n\}$. This is easily computed (see Example 3 in Chapter 2 for an analogous calculation). Given m_n, let n_+ and n_- be the number of positive and negative spins respectively. Since $n_+ + n_- = n$ and $n_+ - n_- = n m_n$ we have $n_+ = \frac{1+m_n}{2}n$ and therefore

$$\mathcal{N}(m_n) = \binom{n}{\frac{1+m_n}{2}n} \approx e^{n h_2(\frac{1+m_n}{2})}, \tag{4.8}$$

where $h_2(p) = -p\ln p - (1-p)\ln(1-p)$ the binary entropy function (expresed with the natural logarithm). The last approximation is asymptotically exact for $n \to +\infty$ and is obtained using Stirling's formula. Equations (4.7) and (4.8) lead to

$$Z \approx e^{-\frac{\beta J}{2}} \sum_{m_n} e^{n\beta\left(\frac{J}{2}m_n^2 + h m_n + \beta^{-1} h_2(\frac{1+m_n}{2})\right)}. \tag{4.9}$$

Since $m_n \in \{\frac{j}{n} | j = -n, -n+1, \ldots, n-1, n\}$ the right hand side of (4.9) is a Riemann sum which tends for $n \to +\infty$ to

$$Z \approx e^{-\frac{\beta J}{2}} n \int_{-1}^{+1} dm\, e^{n\beta\left(\frac{J}{2}m^2 + h m + \beta^{-1} h_2(\frac{1+m}{2})\right)}. \tag{4.10}$$

The integrand has the form $e^{-n\beta\Phi(m)}$ thus the asymptotic behavior of the integral for $n \to +\infty$ can be evaluated by the Laplace method. The dominant contribution comes from a small neighborhood of the value(s) of m where $\Phi(m)$ is minimized. This gives us

$$\begin{aligned} f(\beta, J, h) &= \min_{-1\leq m\leq 1} \{-\frac{J}{2}m^2 - hm - \beta^{-1}h_2(\frac{1+m}{2})\} \\ &\equiv \min_{-1\leq m\leq 1} \Phi(m). \end{aligned} \tag{4.11}$$

With a little bit more effort this formula can be converted into a theorem.

Equation (4.11) teaches us that the free energy is given by the solution of a *variational* problem. The function $\Phi(m)$ which is minimized has various names in the literature. Here we will call it the *potential function*. We will see that the free energies of the coding, compressive sensing and satisfiability problems are all given by such variational expressions involving (often quite complicated) potential functions or even functionals.

The reader should be warned that the free energies of low dimensional models (such as the canonical Ising model on regular grids) the excat expressions for free energies are usually not computable. In particular, they are not given by variational expressions involving the minimization of potential functions of a small number of variables. However this still is a fundamental and useful concept used in approximation schemes or semi-phenomenological theories and in such contexts the potential function is called a "Landau free energy" or a "mean field free energy." We shall consistently adopt the terminology "potential function" which seems to have been adopted in the coding theory literature.

4.3 Average magnetization

Recall from Chapter 2 that in the thermodynamic limit the *magnetization* is defined by the Gibbs average

$$m(\beta J, \beta h) = \lim_{n\to+\infty} \langle \frac{1}{n}\sum_{i=1}^{n} s_i \rangle = \lim_{n\to+\infty} \langle m_n(\underline{s}) \rangle \,. \tag{4.12}$$

This is a generic definition. As we will shortly see for $h = 0$ one must exercise care because the limits $h \to 0_\pm$ are different due to a discontinuity of $m(\beta J, \beta h)$ when $\beta J > 1$. We will see that this is intimately related to the existence of a *phase transition*. But for the moment let us just compute (4.15) for $h \neq 0$.

This computation is easily performed by repeating the calculations of the

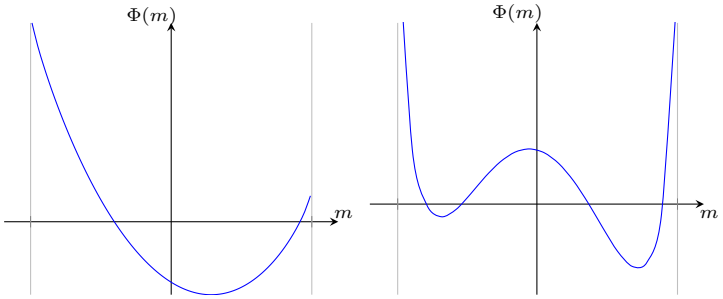


Figure 4.2 *Left*: $\Phi(m)$ for $\beta J < 1$ and $h > 0$. *Right*: $\Phi(m)$ for $\beta J > 1$ and $h > 0$. For $h < 0$ the plot is obtained by a symmetry about the vertical axis.

previous section. Indeed using (4.4) we obtain

$$\begin{aligned}\langle m_n(\underline{s})\rangle &= \frac{\sum_{\underline{s}} m_n(\underline{s}) e^{-\beta \mathcal{H}(\underline{s})}}{\sum_{\underline{s}} e^{-\beta \mathcal{H}(\underline{s})}} \\ &= \frac{\sum_{\underline{s}} m_n(\underline{s}) e^{n\beta\left(\frac{J}{2} m_n(\underline{s})^2 + h m_n(\underline{s})\right)}}{\sum_{\underline{s}} e^{n\beta\left(\frac{J}{2} m_n(\underline{s})^2 + h m_n(\underline{s})\right)}},\end{aligned} \tag{4.13}$$

and we have already found the asymptotic behaviour (4.10) of the denominator as $n \to +\infty$. It is then quite clear that the same arguments applied to the numerator lead to the asymptotics

$$\langle m_n(\underline{s})\rangle \approx \frac{\int_{-1}^{+1} dm\, m\, e^{-n\beta\Phi(m)}}{\int_{-1}^{+1} dm\, e^{-n\beta\Phi(m)}}\,. \tag{4.14}$$

Now for $h \neq 0$ the free energy function $\Phi(m)$ always has a *unique* global minimum. This is illustrated on Figure 4.2, and will be proven by analysis in Section 4.5. Thus, applying the Laplace method to the numerator and denominator of (4.14) one finds that the unique global minimum is selected as $n \to +\infty$, and

$$m(\beta J, \beta h) = \mathrm{argmin}_{-1\leq m\leq 1} \Phi(m)\,. \tag{4.15}$$

The case $h = 0$ is more subtle and interesting. Figure 4.3 shows that $\Phi(m)$ has a unique minimum for $\beta J < 1$, but has *two degenerate* minima for $\beta J > 1$. In the later case, i.e., $(\beta J > 1, h = 0)$, if we would blindly apply the Laplace method we would find a weighted average over the two minimizers, and because of the symmetry $\Phi(m) = \Phi(-m)$ when $h = 0$, this weighted average vanishes. This however is not the correct prescription to compute the "physically relevant" magnetization. In a real sample (or at least on some macroscopic piece of it) the symmetry is broken by an infinitesimal value of the magnetic field pointing in a

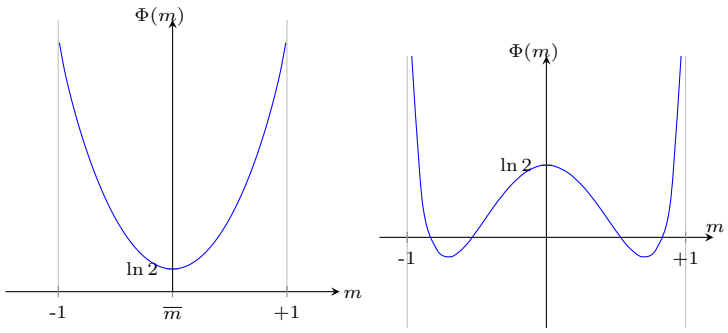


Figure 4.3 *Left*: $\Phi(m)$ for $\beta J < 1$ and $h = 0$. *Right*: $\Phi(m)$ for $\beta J > 1$ and $h = 0$.

definite direction. The correct way to capture this effect is to define

$$m_{\pm}(\beta J) = \lim_{h \to 0_{\pm}} m(\beta J, \beta h) = \lim_{h \to 0_{\pm}} \lim_{n \to +\infty} \langle \frac{1}{n} \sum_{i \in V} s_i \rangle \,. \tag{4.16}$$

When we apply the Laplace method, since h is infinitesimally positive or negative *only one* global minimum is selected, and we get a non-vanishing magnetization (positive or negative) corresponding to one of the two minima on Figure 4.3. Equivalently, the function $m(\beta J, h)$ given above by (4.15) is *discontinuous* on the half-line $(\beta J \geq 1, h = 0)$.

The order of the limits in (4.16) *is crucial*, and for a good physical reason. In a macroscopic system (e.g. a magnet) the symmetry is broken by infinitesimal residual magnetic fields $h = 0_{\pm}$ that select the magnetization of macroscopic domains within the sample (the so-called "Weiss domains"). One can also prepare the whole sample in a state of positive (resp. negative) magnetization by first applying a positive (resp. negative) magnetic field, then lower the temperature such that $\beta J > 1$ and finally remove the magnetic field $h \to 0_+$ (resp. $h \to 0_-$). Definition (4.16) correctly describes the magnetization under such conditions for the Weiss domains of the whole sample. In a Weiss domain one does not get to choose the orientation of the magnetic field (in other words which of the two limits $h \to 0_{\pm}$ applies) and the symmetry is said to be "spontaneously broken". The corresponding magnetization (4.16) is called a "spontaneous magnetization".

We conclude this section with a very important relationship between the free energy $f(\beta, J, h)$ and the magnetization $m(\beta J, \beta h)$. As we mentioned in Chapter 2, Gibbs averages can be obtained by differentiating the free energy. Here we have

$$\langle \sum_{i=1}^{n} s_i \rangle = \frac{1}{\beta} \frac{\partial}{\partial h} \ln Z_n. \tag{4.17}$$

Dividing by n and taking the thermodynamic limit $n \to +\infty$ we find

$$m(\beta J, \beta h) = -\frac{\partial}{\partial h} f(\beta, J, h). \tag{4.18}$$

The careful reader may question the interchange of thermodynamic limit and partial differentiation. In fact this is permitted as long as $h \neq 0$ or $(\beta J < 1, h = 0)$ and (4.18) is correct. On the half-line $(\beta J \geq 1, h = 0)$ the magnetization $m(\beta J, \beta h)$ is discontinuous and the free energy is non-differentiable. More precisely the free energy has different left and right derivatives and the precise version of the relation between magnetization and free energy is

$$m_{\pm}(\beta J) = -\lim_{h \to 0_{\pm}} \frac{\partial}{\partial h} f(\beta, J, h) \tag{4.19}$$

Let us now prove (4.18) away from $(\beta J \geq 1, h = 0)$. According to (4.11) the free energy is obtained by evaluating the potential function $\Phi(m)$ at its unique global minimum $m(\beta J, \beta h)$,

$$f(\beta, J, h) = -\frac{J}{2} m(\beta J, \beta h)^2 + h m(\beta J, \beta h) - \beta^{-1} h_2\left(\frac{1 + m(\beta J, \beta h)}{2}\right). \tag{4.20}$$

We differentiate by carefully taking into account the explicit and implicit h-dependencies in $\Phi(m)$,

$$\begin{aligned} -\frac{\partial}{\partial h} f(\beta, J, h) &= m(\beta J, \beta h) + \frac{\partial}{\partial h} m(\beta J, \beta h) \left.\frac{\partial \Phi}{\partial m}\right|_{m(\beta J, \beta h)} \\ &= m(\beta J, \beta h) \end{aligned} \tag{4.21}$$

To get the last equation we used that $(\partial \Phi / \partial m)|_{m(\beta J, \beta h)} = 0$ and that $\partial m / \partial h$ is well defined and bounded when the parameters are not on the line $(\beta J \geq 1, h = 0)$. This last point follows from the detailled analysis in section 4.5.

Finally let us point out that (4.19) follows from (4.21) by taking the limits $h \to 0_{\pm}$.

4.4 Phase diagram and phase transitions

Consider the *free energy function* $\Phi(m)$ and look at the minimiser $m(\beta J, \beta h)$. As explained in the previous section for $h \neq 0$ this minimizer is unique and there is no ambiguity, so we think of this case. Instead of plotting $m(\beta J, \beta h)$ as a function of βJ and βh, we plot $m(\beta J, \beta h)$ as a function of $1/(\beta J) = k_B T / J$ ("temperature" axis) and $\beta h = h / k_B T$ (the "magnetic field" axis). Figure **??** shows the resulting plot.

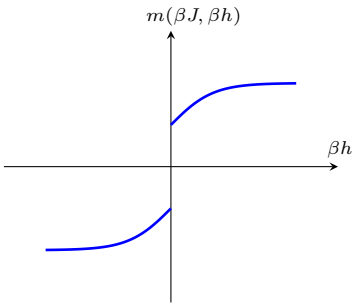


Figure 4.4 A phase transition of first order. The magnetization has a jump discontnuity when $0 \leq k_B T/J < 1$ and $h = 0$.

Why are we interested in this figure? As discussed in the previous section this function represents the average magnetization, a quantity describing the global behavior of the system as a function of the parameters. For some values of the parameters $(\beta J, \beta h)$ the system behaves smoothly when we slightly perturb these parameters. But for some other parameters the system behavior changes abruptly. These are so-called *phase transitions*. A look at the figure reveals two different types of phase transitions.

First and second order phase transitions

When we move along the h-axis, the magnetization $m(\beta J, \beta h)$ has a jump discontinuity when we cross the line segment $(k_B T/J < 1, h = 0)$. This jump of the magnetization is plotted on Figure 4.4 (which shows the cross section with $(k_B T/J < 1$ fixed and h varying). This is called a *first order* phase transition.

At the tip of the line segment $(k_B T/J < 1, h = 0)$, i.e., at the point $(k_B T/J = 1, h = 0)$ the magnetization is continuous but not differentiable. For example if we move along the temperature axis or along the magnetic field axis across the point $(k_B T/J = 1, h = 0)$, the function $m(\beta J, \beta h)$ changes continuously, but its derivatives with respect to temperature or magnetic field jump. This is shown on Figure 4.5 where we move accross the point $(k_B T/J = 1, h = 0)$ by varying the temperature or the magnetic field. Such behavior is called a *second order phase transition* (or sometimes a continuous phase transition).

Finally, for all values of $(\beta J, \beta h)$ away from the segment $(k_B T/J \leq 1, \beta h = 0)$ the function $m(\beta J, \beta h)$ changes smoothly and is in fact analytic (i.e. infinitely differentiable with an absolutely convergent Taylor expansion).

To understand the terminology "first" and "second" order, recall Equ. (4.18). At a first order transition, where the magnetization jumps, the first derivative of the free energy is discontinuous. At a second order phase transition the magnetization is continuous but its first derivative is discontinuous, so equivalently the second derivative of the free energy is discontinuous.

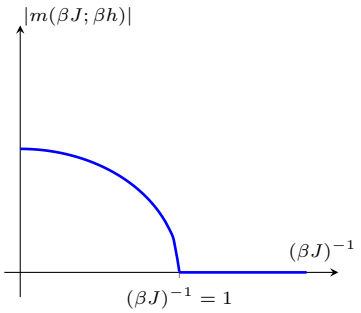


Figure 4.5 Phase transitions of second order as a function of temperature. *Left*: $m_{\pm} \propto \pm|1 - k_BT/J|^{1/2}$ for $\beta J \to 1_-$. *Right*: $m_{pm} \propto \pm|\beta h|^{1/3}$. Both behaviours are continuous but non-differentiable.

The phase diagram

Let us now consider a sligthly different perspective and explain some more terminology. We look at Figure **??** again, but this time we consider the picture "from the top," i.e., we only show the temperature and magnetic field axis. This is shown in Figure 4.6 and is commonly called a *phase diagram*. On this diagram we have indicated the various ways to change parameters and the corresponding phase transition.

The line segment $(k_BT < 1, h = 0)$ is called the *coexistence line*. This name is easily explained. If we approach this line from the top or the bottom, i.e., we consider the limit $h \to 0_{\pm}$, then we get two (opposite) values $m_{\pm}$ for the magnetization. So "on the line" we can think of having two possible "coexisting" phases.

The tip of the line $(k_BT/J = 1, h = 0)$ is called the *critical point*. Across this point the magnetization is continuous but non-differentiable. For example if we set $h = 0$ and vary only the temperature one finds

$$m_{\pm} \propto \pm|1 - \frac{k_BT}{J}|^{1/2}$$

for $k_BT/J \to 1_-$; if one sets $k_BT/J = 1$ and varies only the magnetic field one finds

$$m_{\pm} \propto \pm|\beta h|^{1/3}$$

for $h \to 0_{\pm}$ (see Section 4.5 for a derivation). This sort of power law behaviour is called a *critical behavior* and is characterized by *critical exponents*, here 1/2 and 1/3. Critical exponents do not depend on the details of the Hamiltonian but only on general features such as the dimensionality of the graph and the symmetry of the model (see Section 4.9 for more details).

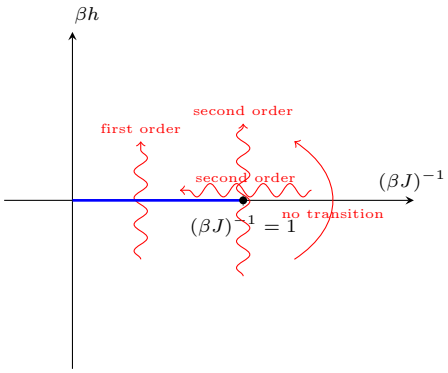


Figure 4.6 On the *coexistence line* $(0 \leq \beta J)^{-1} < 1, h = 0)$ two thermodynamic phases with (opposite) magnetizations $m_{\pm}(\beta J)$ may coexist. Crossing this line corresponds to a first order phase transition. The line terminates at the *critical point* $((\beta J)^{-1} < 1, h = 0)$. Going through the tip ia a second order phase transition. In the rest of the phase diagram the magnetization varies smoothly (is analytic).

Continuity and concavity of the free energy

The variational expression (4.11) for the free energy has a very important property. We are minimizing a function $\Phi(m)$ which is linear in h and $\beta^{-1} = k_B T$. The outcome $f(\beta, J, h)$ is necessarily *concave* in h and T (see exercise). This property is not particular to the Curie-Weiss model but must always hold if the model is to be "physically acceptable".[2] This is a very important guiding principle because most of the time even if the model is physically sound one cannot compute exactly the free energy and a non-concavity indicates that we have to correct our approximations or modify the end result by an educated guess. More about this in Chapter 12!

Note that in our coding, compressive sensing and satifiability models are not physical systems and there is no absolute a priori requirement that their "free energies" should satisfy this guiding principle. Nevertheless it is true that this requirement must be satisfied at least with respect to the temperature T. This is true for a mathematical reason. Indeed the generic definition of the free energy (for a finite system) is $f = -\frac{1}{n} \ln Z$ with $Z = \sum_{\underline{s}} e^{-\mathcal{H}(\underline{s})/k_B T}$. It can easily be checked from formula (2.19), which relates the variance of the Hamiltonian to the second derivative of the free energy with respect to T, that f is a concave function of T. More generally, the same argument can be made for other parameters λ that enter linearly[3] in the Hamiltonian through some term of the form $\lambda A(\underline{s})$,

[2] Thermodynamics teaches us that at when a system has reached the state of thermal equilibrium the "thermodynamic potentials" must be concave as a function of the "intensive parameters" (e.g. temperature, magnetic field, chemical potential) and convex as a function of "extensive parameters" (e.g. volume, number of particles). The free energy considered here (the "Gibbs free energy") depends on intensive quantities and must be concave.

[3] such λ's are called "intensive" parameters

because the second derivative of the free energy with respect to λ is equal to minus the variance $\langle A(\underline{s})^2\rangle - \langle A(\underline{s})\rangle^2$. An example is the magnetic field h which enters through the term $h\sum_{i=1}^{n} s_i$. To summarize, concavity of the free energy must hold even for non-physical systems at least with respect to parameters that enter linearly in the Hamiltonian. This is an important guiding principle that an approximation scheme must respect.

Concavity of the free energy has a few interesting consequences. First the limit of a sequence of concave functions is concave and continuous (concave functions are continuous on open sets). Thus the free energy never has jumps as a function of the temperature and/or magnetic field, even in the thermodynamic limit. Phase transitions manifest themselves only as discontinuities in the derivatives of the free energy. A first order phase transition is one where the first derivative jumps, a second order transition is one where the first derivative is continuous but the second derivative jumps. More generally if the first $n-1$ derivatives of the free energy are continuous and the n-th derivative has a jump one say the phase transition is of order n (there exist phase transitions of "infinite order" where all derivatives of the free energy are continuous but the function has a non-analyticity). This classification of phase transitions due to Ehrenfest. It is not the only one, nor the most modern one,[4] but it is one that suits our purposes.

Phase transitions that occur as singularities in the derivatives of the free energies are also often called *static* or *thermodynamic* phase transitions. This nomenclature allows to distinguish them from abrupt changes in the behaviour of algorithms or of some dynamics. The later type of abrupt changes are called *dynamical* or *algorithmic* phase transitions. We stress that they are *not* visible as singularities in the derivatives of the free energy and cannot be discovered by computing only the free energy. Also, they depend on the algorithm at hand. However we will see that there are some dynamical transitions that are in some sense "fundamental". What makes some dynamical transitions "fundamental" because of their intimate connection to static transitions.

For the Curie-Weiss model, so far, we have only discussed static phase transitions (of first and second order). In the next section we turn to the mathematical analysis of the fixed point equation and will discover as a by-product a special line in the phase diagram - the so called *spinodal line* - which is related to algorithmic phase transitions of message passing algorithms studied in part II.

4.5 Analysis of the fixed point equation

We have plotted the three-dimensional picture of $m(\beta J, \beta h)$ and from this we can in principle see all phase transitions. But there is value in rederiving our conclusions in a more quantitative and classical way using calculus. By doing so,

[4] Phase transitions can also be classified according to changes in "symmetry" or even according to "topological" properties of the states of matter.

we not only will add details to our picture, but we also encounter new notions which will reappear in later chapters

Curie-Weiss mean field equation

We solve the variational problem (4.11) by writing down the stationarity condition $d\Phi(m)/dm = 0$[5] and obtain the *fixed point equation*

$$m = \tanh(\beta(Jm + h)). \tag{4.22}$$

Of course this equation may have many solutions, and one has to select the ones which minimize $\Phi(m)$. If no solution is present then the minimum is attained at the boundaries of the interval over which we minimize, i.e., $m = \pm 1$. This case happens only for $\beta = +\infty$ (zero temperature) and will not concern us too much in the following.

Equ. (4.22) is also called the Curie-Weiss *mean field* equation. Let us explain the terminology here. Equation (4.22) expresses the magnetization as that of an hypothetical single spin seeing an effective magnetic field $Jm + h$. Indeed the Hamiltonian of this single spin would be $-(Jm + h)s$ and its magnetization

$$m = \langle s \rangle = \frac{\sum_{s=\pm 1} s e^{-\beta(Jm+h)s}}{\sum_{s=\pm 1} e^{-\beta(Jm+h)s}} = \tanh(\beta(Jm + h)) \tag{4.23}$$

One can think of Jm as a mean magnetic field felt by a each spin on the vertices of the complete graph, and which adds up to the external field h.

This way of thinking is at the basis of the "mean field theory" of magnetism pioneered by Curie and Weiss and is also at the basis of the generic "mean field approximations" for general Ising spin systems. In the Curie-Weiss model it turns out that the mean field theory is exact. For Ising models on low dimensional regular grids such theories are not exact but often give a valuable first insight.[6] It must not be thought that mean field equations are always easy to derive, let alone assess wether they are exact or not. We will see that the solutions of all our problems are intimately related to mean field equations but that these are considerably more subtle to derive than in the Curie-Weiss model (let alone assess whether they are exact or not).

Solutions of the Curie-Weiss equation

Now our task is to find solutions of the Curie-Weiss equation (4.22) and select the ones that minimize $\Phi(m)$. The easiest way to determine the whole set of solutions is to look at the equivalent equation

$$h(m) = -Jm + (2\beta)^{-1} \ln(\frac{1+m}{1-m}). \tag{4.24}$$

[5] Differentiating explicitly leads to $\beta(Jm + h) - \frac{1}{2}\ln(\frac{1+m}{1-m}) = 0$. Then one uses the identity $\tanh\{\frac{1}{2}\ln(\frac{1+m}{1-m})\} = m$ to obtain 4.22.

[6] The interested reader can find more information about this point in Section 4.9.

Figure 4.7 Solutions of the Curie-weiss equation for $\beta J < 1$ (left), $\beta J = 1$ (middle), $\beta J > 1$ (right). These are three instances of the curve (4.24) with m on the vertical axis and h on the horizontal axis.

Here the magnetic filed is viewed as a function of $m \in [-1, +1]$. This function has two vertical asymptotes $h(m) \to +\infty$, $m \to \pm 1$. From

$$\frac{dh}{dm} = (1 - \beta J(1 - m^2))/\beta(1 - m^2)$$

we immediately see that: (i) for $\beta J < 1$ the curve is monotonously increasing; (ii) for $\beta J = 1$ an inflexion point with two degenerate stationary points develops at $m = 0$, $h(0) = 0$; (iii) for $\beta J > 1$ the curve has a local maximum and a local minimum at

$$m_{\rm sp} = \pm\sqrt{1 - (\beta J)^{-1}}, \qquad h_{\rm sp} = \pm h\big(\sqrt{1 - (\beta J)^{-1}}\big).$$

The points $(h_{\rm sp}, m_{\rm sp})$ are called *spinodal points*, a terminology that we explain a bit later. The curves $h(m)$ are plotted on Figure 4.7 for the three distinct cases $\beta J < 1$, $\beta J = 1$ and $\beta J > 1$, with the m-axis vertical and the h-axis horizontal (this choice of axis is teh conventional one in the present context). We discuss each case separately.

For $\beta J < 1$ (high temperatures) the solution of the Curie-Weiss equation is unique for all h and the free energy function $\Phi(m)$ is convex with a single minimum as shown on Figures 4.2 and 4.3.

In the case $\beta J > 1$ (low temperatures) we have to distinguish between various values of the magnetic field: $|h| > |h_{\rm sp}|$ and $|h| < |h_{\rm sp}|$. When $|h| > |h_{\rm sp}|$ (large field) the solution of the Curie-Weiss equation is again unique and $\Phi(m)$ is convex with one minimum. On the other hand for $|h| < |h_{\rm sp}|$ (small field) there are three solutions $m_- < m_0 < m_+$. We remark that $\frac{d^2\Phi}{dm^2} = \frac{dh}{dm}$ and a look at the plot of $h(m)$ shows that $dh/dm|_{m_\pm} > 0$ whereas $dh/dm|_{m_0} < 0$. Thus the two extremal solutions $m_\pm$ are (locally) minima of $\Phi(m)$ and . the middle solution is a (local) maximum. This is depicted on Figure 4.3. The free energy is given by the global minimum, that is $\Phi(m_-)$ for $h \in [-h_{\rm sp}, 0[$ and $\Phi(m_+)$ for $h \in [0, h_{\rm sp}, 0[$.

So far we have left out two borderline cases. For $\beta J = 1$ the inflexion point of $h(m)$ means that $\Phi(m)$ is convex (but not strictly convex) with two degenerate minima at $m = 0$, and the free energy is $\Phi(0) = 0$. Finally for $\beta J > 1$ and

$h = \pm|h_{\rm sp}|$ the solution m_0 is degenerate with $m_\mp$, $\Phi(m)$ has a an inflexion point at $m_0 = m_\mp$ and a minimum at $m_\pm$, and the free energy is given by $\Phi(m_\pm)$.

Now that we have derived in detail all solutions of the Curie-Weiss equation we can fill in a few gaps in our previous description of first and second order phase transitions.

First order transition, metastability and spinodal decomposition

In the *low temperature* phase, $\beta J > 1$, the (equilibrium) magnetization given by the *global* minimizers of $\Phi(m)$ is the discontinuous function of h shown on Figure 4.4 This function is the part of the curve in Figure 4.7 (right plot) that corresponds to the *global* minimum of $\Phi(m)$. It exhibits a jump at $h = 0$ which constitutes the first order phase transition.

But what is the physical interpretation of the rest of the curve on Figure 4.7? We distinguish two parts corresponding to different physical phenomena: the thick-dotted and the thin-dotted pieces of the curve.

Make the curve 4.7 with dotted parts to refer to it more easily in this paragraph

First we interpret the thick-dotted piece. Imagine that we have a piece of iron that we prepare in a state of positive magnetization which corresponds to the global minimum of $\Phi(m)$. Now we diminish the magnetic field very slowly starting from h positive and large. Under the right conditions as h slowly becomes negative and as long as it stays greater than $-|h_{\rm sp}|$ the magnetization will *not* jump to the negative branch but will follow the positive branch which corresponds to the local minimum of $\Phi(m)$. As h goes from positive to negative values in $[-|h_{\rm sp}|, 0]$ the global minimum of the free energy function transforms into a local minimum, and if this transformation occurs sufficiently slowly the system remains trapped in this same potential well. Similarly if we start with a negative magnetization state at large negative magnetic fields and slowly increase h, as long as $h \in [0, |h_{\rm sp}|]$ the magnetization remains negative and trapped in the local minimum of $\Phi(m)$. Local minima of the free energy function (equivalently the positive magnetization branch for $h \in [-|h_{\rm sp}|, 0[$ and the negative branch for $h \in [0, |h_{\rm sp}|]$) are called *metastable states*. These are called "metastable" because in physical systems they have a finite lifetime, and decay through a process known as *nucleation*. Because of thermal fluctuations clusters of spins with their magnetization in the stable global minimum appear, and when a nucleus of sufficient size forms it grows and the stable magnetization state takes over. The lifetime is essentially determined by the probability that a nucleus with sufficient size forms. We should stress that in the Curie-Weiss model the lifetime of metastable states is in fact infinite because on a complete graph a nucleus cannot grow whatever its size. We will come back to this mechanism in Chapter 14.

Let us now discuss the thin-dotted piece of the curve in Figure 4.7. This part has a *negative* slope for $|h| < |h_{\rm sp}|$ and corresponds to the solution m_0 which is a local *maximum* of $\Phi(m)$. This is an *unstable* solution which a physical sys-

Figure 4.8 *Left*: coexistence and spinodal lines in the $h - T$ plane. *Right*: equilibrium magnetisation and spinodal line in the $m - T$ plane. In the region between the equilibrium and spinodal lines a physical system supports metastable states. In the interior of the spinodal line a homogeneous state is unstable and spinodal decomposition into domains of equilibrium magnetozation occurs.

tem cannot sustain,[7] and the system will spontaneously transition towards its natural equilibrium state. If a piece of material is forced in such an unstable magnetization state by applying a sudden quench,[8] the magnetisation cannot remain homogeneous and regions with opposite equilibrium magnetisations will spontaneously develop. These regions are separated by *domain walls* where the magnetization transitions between the two equilibrium values $m_{\pm}$. This process proceeds homogeneously throughout the system because there is no energy barrier, and the energy gained in the regions where the magnetization takes equilibrium values goes into the formation of the domain walls. This phenomenon is called the *spinodal decomposition*.

Figure 4.8 summarises the discussion above. In the (T, h) phase diagram the dotted line $h_{\rm sp} = \pm h\big(\sqrt{1-(\beta J)^{-1}}\big)$ as a function of T is called the *spinodal line*. Metastable states can exist only within the region bounded by this line; outside of this region there are no metastable states. In the (T, m) plane we have the equilibrium line $m_{\pm}$ (also called bimodal line), the spinodal line $m_{\rm sp} = \pm\sqrt{1-(\beta J)^{-1}}$ which marks the limit of metastability, and inside we have the region of instability where a spinodal decomposition occurs.

Metastability and spinodal decomposition constitute important chapters of non-equilibrium statistical physics. These phenomena are difficult to approach from first principles because they are dynamical processes that go beyond the pure Gibbs equilibrium description. An important and natural approach to estimate the lifetime of a metastable state is based on the analysis of Markov Chain Monte Carlo (MCMC) dynamics satisfying a detailed balance condition such that the Gibbs state is a stationary distribution. For the Curie-Weiss model that

[7] This would correspond to a magnetization that decreases with increasing magnetic field, which is unphysical.

[8] For example one can start at high temperatures and $h = 0$ so that the magnetization is initially zero and suddenly cool the system below the critical temperature to obtain an unstable state with $m = 0$, the maximum of $\Phi(m)$ at $h = 0$.

concerns us here the lifetime of metastable states is infinite (more precisely the mixing time of MCMC dynamics diverges with the system size) and in a sense this is why we have accessed them without any dynamical description. The local minima of the potential function contain the information about the metastable states. Moreover, the domain walls that occur in a spinodal decomposition are described by the unstable part of the curve.

In our problems (coding, compressed sensing and satisfiability) we will see in due time that there are algorithmic analogs of metastability and spinodal lines. Algorithms can remain trapped in metastable states and spinodal lines indicate the limits of existence of these trapping states.

Second order transition and critical behaviour

As one traverses the tip of the coexistence line the magnetisation varies in a continuous but non-differentiable way as shown on Figure 4.5. Here we derive these critical behaviours and the associated exponents.

First we look at the behaviour of the magnetization for $h = 0$ as a function of $(\beta J)^{-1} = k_B T/J$. For βJ close to $\beta J = 1$ all solutions of the Curie-Weiss equation are close to zero, so we expand around $m = 0$,

$$m = \tanh \beta J m \approx \beta J m - \frac{(\beta J)^3}{3} m^3 + \dots \tag{4.25}$$

For $\beta J < 1$ the only real solution is $m = 0$, and for $\beta J > 1$ we have two extra solutions which are the ones that minimise the free energy function

$$m_\pm \approx \pm 3(\beta J - 1)^{1/2} \propto |1 - \frac{T}{T_c}|^{1/2}, \quad \beta J \to 1_+ \quad \text{or} T \to T_c = J/k_B \,. \tag{4.26}$$

Next we set $\beta J = 1$ (or $T = T_c$) and look what happens for $h \to 0_\pm$. Again expanding the Curie-Weiss equation around $m = 0$ we find

$$m = \tanh(m + J^{-1}h) \approx m + J^{-1}h - \frac{1}{3}(m + J^{-1}h)^3 + \dots \tag{4.27}$$

which yields

$$m \approx \pm(3J^{-1}|h|)^{1/3} \propto \pm|h|^{1/3}, \quad h \to 0_\pm. \tag{4.28}$$

4.6 Ising model on a tree

We consider a regular finite tree with a total of n vertices and "coordination number" $k \geq 3$.[9] All vertices have degree k, except for the leaf nodes which have degree 1. The tree is depicted on figure 4.9. On this figure we have singled

[9] For $k = 2$ the tree is a line and the model is solved by the transfer matrix method in the exercises of Chapter 2. The present method of solution also works in this case and is the subject of an exercise. Recall however that there is no phase transitions for strictly positive temepratures.

out the "central" node o, furthermore there are L "concentric levels" with the neighbords of the root forming level 1, and so on till the leaf nodes forming level L. Denoting by V_L and E_L the vertex and edge sets for the tree with L levels, the Hamiltonian is simply

$$\mathcal{H}_L = -J \sum_{\{i,j\}\in E_L} s_i s_j - h \sum_{i\in V_L} s_i \tag{4.29}$$

were $J > 0$, $h \in \mathbb{R}$ (each edge is counted once in the first sum). We call this model the *tree-Ising* model for short.

Our goal is to compute the magnetization of the central node in the limit $L \to +\infty$, that is

$$m_o(\beta, h) \equiv \lim_{L\to+\infty} \langle s_o \rangle_L$$

Here we are not directly interested in $-\log Z/\beta n$ and will *not* compute this quantity for the tree. Why is this so? For $k \geq 3$ the leaf nodes form a positive fraction, namely $(k-2)/(k-1)$, of the total number of vertices.[10] In the limit of large coordination number $k \to +\infty$ this fraction even goes to one. As a result $-\ln Z/\beta n$ contains the "spurious" effect of leaf nodes whose contribution dominate in the thermodynamic limit. However this is *not* what we want to capture with this model. Indeed our coding and satisfiability problems are defined on random graphs which are locally tree like but at the same time have all their nodes equivalent. It is therefore much more relevant for us to compute the magnetization of the root node of the tree which is "deep inside" the graph.

In fact a model that is very relevant for us is an Ising model on a random k-regular graph (this graph ensemble can be described by the Gallager $(k, 2)$ factor graphs). On a random regular graph all vertices are equivalent and the average free energy $\mathbb{E}[\log Z]/\beta n$ is indeed a very relevant object that we wish to compute. A rigorous derivation however requires more advanced methods that will only be developed in part III. Because the k-regular random graphs are locally tree like, a computation of the magnetization $m_o(\beta, h)$ of the central node of the tree is a good start. We show in Section 4.7 how this computation suggests a quick heuristic derivation of the average free energy of the random graph ensemble.

Recursive equations for the magnetization

On the tree with L levels the magnetization of the root node o is by definition

$$\langle s_o \rangle_L = \frac{1}{Z_L} \sum_{\underline{s}} s_o \, e^{-\beta \mathcal{H}_L} \tag{4.30}$$

We can compute this average from a recursive equation obtained by summing over leaf nodes. A spin s_i attached to a leaf node interacts *only* with the spin

[10] The total number of vertices on the tree is $n = 1 + k + k(k-1) + k(k-1)^2 + \cdots + k(k-1)^{L-1} = 1 + k\frac{(k-1)^L - 1}{k-2}$, and the number of leaf nodes is $k(k-1)^{L-1}$. For $L \to +\infty$ we find $k(k-1)^{L-1}/n \to (k-2)/(k-1)$.

Figure 4.9 A finite tree with coordination number k. All vertices except for the leafs have k neighbords. The dotted circles represent the levels $\ell = 1, \dots, L$.

$s_{p(i)}$ at the parent node $p(i)$. So the spins at the leafs occur in the Gibbs weight only in a term of the form

$$e^{\beta \sum_{i \text{ at level } L} (J s_{p(i)} + h) s_i} \tag{4.31}$$

and the statistical sum over $s_i \in \{-1, +1\}$ can be performed. This yields the contribution

$$\prod_{i \text{ at level } L} \left(e^{\beta J s_{p(i)} + \beta h} + e^{-\beta J s_{p(i)} - \beta h} \right) \tag{4.32}$$

and each term in this product equals

$$\begin{aligned} e^{\beta h}\big(\cosh(\beta J) + s_{p(i)} \sinh(\beta J)\big) + e^{-\beta h}\big(\cosh(\beta J) - s_{p(i)} \sinh(\beta J)\big) & \\ \propto \cosh(\beta h) \cosh(\beta J) + s_{p(i)} \sinh(\beta h) \sinh(\beta J)) & \\ \propto 1 + s_{p(i)} \tanh(\beta h) \tanh(\beta J) & \\ \propto e^{s_{p(i)} \operatorname{atanh}\{\tanh(\beta h) \tanh(\beta J)\}} . & \end{aligned} \tag{4.33}$$

In the last two terms we have not written explicitely the proportionality constants (independent of $s_{p(i)}$) because they cancel out in the ratio (4.30). Now, since each parent node $p(i)$ at level $L-1$ has $k-1$ children i at level L, the product (4.32) is proportional to

$$\prod_{i \text{ at level } L-1} e^{s_i (k-1) \operatorname{atanh}\{\tanh(\beta h) \tanh(\beta J)\}} \tag{4.34}$$

and (4.30) becomes

$$\langle s_o \rangle_L = \frac{1}{Z^{(1)}_{L-1}} \sum_{\underline{s}} s_o \, e^{-\beta \mathcal{H}^{(1)}_{L-1}} \tag{4.35}$$

where $\mathcal{H}^1_{L-1}$ is the Hamiltonian of a tree-Ising model with $L-1$ levels

$$\mathcal{H}^{(1)}_{L-1} = -J \sum_{(i,j) \in E_{L-1}} s_i s_j - h \sum_{i \in V_{L-2}} s_i - u_1 \sum_{i \text{ at level } L-1} s_i \tag{4.36}$$

and a "renormalized" magnetic field acting on the spins of level $L-1$

$$u_1 = h + \beta^{-1} (k-1) \operatorname{atanh}\{\tanh(\beta h) \tanh(\beta J)\} \tag{4.37}$$

This renormalized magnetic field acting on spins at level $L-1$ can be interpreted as the "bare field" plus the sum of $k-1$ "effective fields" created by the interaction with the $k-1$ children spins. Iterating this calculation $L-1$ times we find an Ising model defined on the root o and its k neighbords with Hamiltonian

$$\mathcal{H}_1^{(L-1)} = -J\sum_{i=1}^{k} s_o s_i - u_{L-1}\sum_{i=1}^{k} s_i \tag{4.38}$$

where u_{L-1} is calculated iteratively from

$$u_\ell = h + \beta^{-1}(k-1)\,\mathrm{atanh}\big\{\tanh(\beta u_{\ell-1})\tanh(\beta J)\big\}, \qquad u_0 = h \tag{4.39}$$

for $\ell = 1, \dots L-1$. Finally, we calculate the magnetization of the root spin using the Hamiltonian $\mathcal{H}_1^{(L-1)}$, and we find

$$\langle s_o\rangle_L = \tanh\big(\beta h + k\,\mathrm{atanh}\big\{\tanh(\beta u_{L-1})\tanh(\beta J)\big\}\big) \tag{4.40}$$

Note that the "renormalized" field that acts on the root spin is equal to the "bare field" h plus the sum of k "effective fields" created by the interaction with its k neighbords at level 1.

The magnetization in thermodynamic limit is given by taking $n \to +\infty$, or equivalently $L \to +\infty$. Thus we have to compute u_∞ from the recursion (4.39) with the initial condition $u_0 = h$, and then evaluate the magnetization

$$\begin{aligned} m_o(\beta, h) &= \lim_{L\to+\infty} \langle s_o\rangle_L \\ &= \tanh\big(\beta h + k\,\mathrm{atanh}\big\{\tanh(\beta u_\infty)\tanh(\beta J)\big\}\big)\,. \end{aligned} \tag{4.41}$$

It is possible to write down an explicit fixed point equation for the magnetization by expressing $\tanh(\beta u_\infty)$ as a function of $m_o(\beta, h)$ and replacing in the fixed point condition of the recursion (4.39) (see exercises).

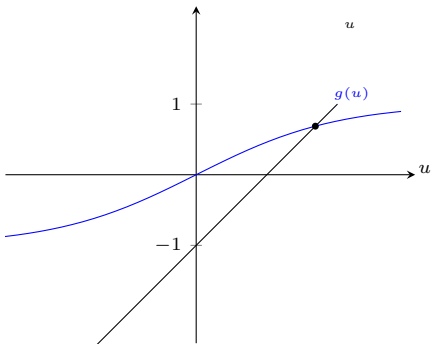
Analysis of the recursive equation (4.39)

Before we turn our attention to the phase transitions we must determine the fixed point of the iteration (4.39). We first discuss the set of solutions of the associated fixed point equation

$$f(u) = g(u) \tag{4.42}$$

where $f(u) = u - h$ and $g(u) = \beta^{-1}(k-1)\,\mathrm{atanh}\big\{\tanh(\beta u)\tanh(\beta J)\big\}$. Both functions are obviously monotone increasing (tanh and atanh are monotone increasing), and $g(u)$ has two horizontal asymptotes at infinity $\lim_{u\to\pm\infty} g(u) = \pm(k-1)J$, and a slope at the origin equal to $\lim_{u\to 0} g(u)/u = (k-1)\tanh(\beta J)$. It is also easy to see that the maximum slope of $g(u)$ is attained at $u = 0$. We distinguish two cases: $(k-1)\tanh(\beta J) < 1$ and $(k-1)\tanh(\beta J) > 1$.

The *high temperature* case $(k-1)\tanh(\beta J) < 1$ is illustrated on Figure 4.10. Since the maximum slope of $g(u)$ is less than one it follows that there is only one solution to the fixed point equation (4.42), and the iterations (4.39) necessarily



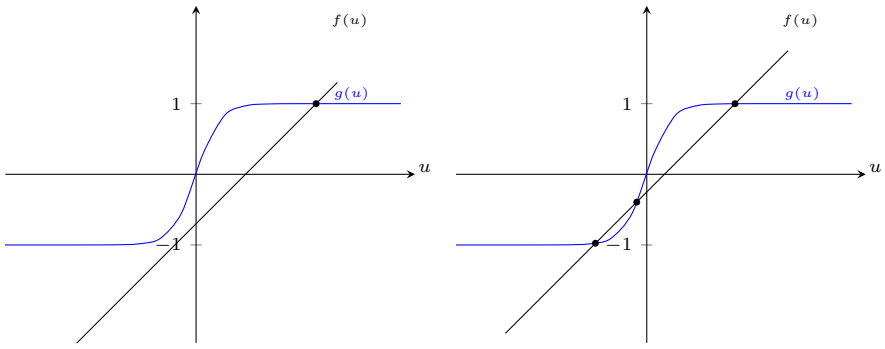
(a) Fixed points

Figure 4.10 Fixed points and iterations for $(k-1)\tanh(\beta J) < 1$ and $h > 0$

converge to this unique fixed point. These iterations are shown as a dotted line on Fig. 4.10. In particular, for $h = 0$ it is obvious to see that the iterations initialized with $u_0 = h = 0$ yield $u_\ell = 0$ for all ℓ. Thus the spontaneous magnetization vanishes: $\lim_{h\to 0} m_o(\beta J, h) = m_o(\beta J, 0) = 0$.

The *low temperature* situation $(k-1)\tanh(\beta J) > 1$ is richer. Here the slope of $g(u)$ at the origin is greater than one so multiple fixed points are possible. As shown on Figure 4.11 for $|h| < h_{\rm sp}$ Equ. (4.42) has three solutions while for $|h| > h_{\rm sp}$ it has only one solution. The value of h can be analytically computed from the condition that $f(u)$ and $g(u)$ are tangent (this is left as an exercise).

The question that remains to be answered when there are multiple fixed points is: how do we choose the correct one? In the Curie-Weiss case, for $h \neq 0$, the correct solution was selected by minimizing the free energy, and for $h = 0$ it was determined by a limiting process $h \to 0_\pm$. Here, when $h \neq 0$, the correct solution is enforced by the initial condition of the iteration (4.39). Figure 4.11 (dotted line) shows that starting with $u_0 = h > 0$ the iterations always converge to the maximal solution $u_+ > 0$. Similarly if $h < 0$ starting from $u_0 = h < 0$ the iteration always converges to the minimal solution $u_- < 0$. The most interesting case is $h = 0$. The iterations initialized with $u_0 = h = 0$ trivially give $u_\ell = 0$ for all ℓ and this does not give the correct "physical" magnetization. We already discussed at length how to solve this conundrum: physically the symmetry is broken by an infinitesimal magnetic field $h = 0_\pm$. This means that the physically correct magnetization, $m_\pm = \lim_{h\to 0_\pm} m_o(\beta, h)$ is again determined by the maximal fixed points $u_\pm$. Said differently, for $(k-1)\tanh(\beta J) > 1$ and $h = 0$ the trivial fixed point $u = 0$ is *unstable*: iterations initialized with an infinitesimally positive or negative value of u_0 are driven to the non-trivial *stable* fixed points $u_\pm$.



(a) Fixed points (b) Fixed points

Figure 4.11 Fixed points and iterations for $(k-1)\tanh(\beta J) > 1$ and $h > h_{\rm sp}$ (left), $0 < h < h_{\rm sp}$ (right).

Phase transitions and phase diagram

Using the preceding results one finds a phase diagram with the same qualitative features than the one of the Curie-Weiss model (see Figure 4.6). In the (T, h) plane there is a coexistence line given by $(k-1)\tanh(\beta J) > 1$ and $h = 0$. The tip of the line on the $h = 0$ axis is at $(k-1)\tanh(\beta J) = 1$ or equivalently $k_B T_c = \frac{J}{2}\{\ln(k/(k-2))\}^{-1}$. This tip is the critical point. When we fix the temperature below the critical point and vary h accross the coexistence line the magnetization has a jump equal to $m_+ - m_- \neq 0$ where $m_\pm = \lim_{h\to 0_\pm} m(\beta J, h)$. This is the first order phase transition just as in the Curie-Weiss (Figure 4.4).

Going accross the critical point we find that the magnetization vanishes continuously but is not differentiable. This is a second order phase transition and the critical exponents, $1/2$ and $1/3$ governing the magnetization behaviour as a function of temperature and magnetic field, are the same than in the Curie-Weiss model (see Figure 4.5 and equations (4.26), (4.28)). To see this one may linearize the fixed point equation (4.42) around $u = 0$ (because close to the critical point this equation admits only small solutions) and deduce the leading behavior of the magnetization (see exercises).

Finally let us point out that the spinodal lines in the (T, h) and (T, m) planes can be calculated analytically by looking at the condition that $f(u)$ and $g(u)$ are tangent when $(k-1)\tanh(\beta J) > 1$.

4.7 Free energy for the random k-regular graph

The magnetization of the root node in the tree can be used to derive the average free energy of the Ising model on a random k-regular graph. Here we provide a heuristic derivation which is made rigorous in part III once we have more advanced tools at our disposal. The important point we wish to illustate already

now, is that the average free energy is given by the minimum of a potential function and the structure of the solution is similar to the Curie-Weiss model (even if more tecnically involved).

On the random k-regular graph all vertices are equivalent and therefore

$$-\frac{1}{\beta n}\frac{\partial}{\partial h}\mathbb{E}[\ln Z] = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\langle s_i\rangle] = \mathbb{E}[\langle s_o\rangle] \tag{4.43}$$

where o is any randomly chosen node. A random graph is locally tree-like, which means that the restriction of the graph to a distance d from o is with probability $1 - O(k^d/n)$ a tree. In the k-regular case the tree has coordination number k. We therefore expect that

$$\lim_{n\to+\infty}\mathbb{E}[\langle s_o\rangle] = m_o(\beta, h)\,. \tag{4.44}$$

where the right hand side is the magnetization of the root node on an infinite regular tree. From (4.43) and (4.44) we calculate the average free energy by integrating the magnetization,

$$-\lim_{n\to+\infty}\frac{1}{\beta n}\mathbb{E}[\ln Z]|_{h_1}^{h_2} = \int_{h_1}^{h_2} dh\, m_o(\beta, h) \tag{4.45}$$

For $h_1 \to -\infty$ all spins polarise towards $s_i = -1$, thus the free energy is simply equal to the energy of this configuration,[11] $-J|E| + h_1 n$ which equals $(-Jk/2 + h_1)n$ because $kn = 2|E|$ on a k-regular graph. Therefore replacing $h_2 \to h$ and $h_1 \to -\infty$ in (4.45) we find

$$-\lim_{n\to+\infty}\frac{1}{\beta n}\mathbb{E}[\ln Z] = -\frac{Jk}{2} + h + \int_{-\infty}^{h} dh'\,(m_o(\beta, h') + 1). \tag{4.46}$$

Remarkably the integral can be computed and one finds[12]

$$-\lim_{n\to+\infty}\frac{1}{\beta n}\mathbb{E}[\ln Z] = \Psi(u_\infty) \tag{4.47}$$

where

$$\begin{aligned}\Psi(u) = &-\frac{k}{2\beta}\ln\cosh(\beta J) + \frac{k}{2\beta}\ln\bigl(1 + \tanh(\beta J)(\tanh u)^2\bigr)\\ &-\frac{1}{\beta}\ln\bigl\{e^{\beta h}(1 + \tanh(\beta J)\tanh(\beta u))^k + e^{-\beta h}(1 - \tanh(\beta J)\tanh(\beta u))^k\bigr\}\end{aligned} \tag{4.48}$$

and u_∞ is the fixed point of the recursion (4.39). One can also check that this fixed point equation is precisely the stationarity condition for $\Psi(u)$ and u_∞ is a minimum.

Summarizing, the free energy on the random k-regular graph is given by

[11] With all spins polarized to the value $s_i = -1$ there is no entropic contribution.

[12] The direct computation of the integral is a bit lengthy, but the result can be checked by carefully differentiating $\Psi(u_\infty)$ with respect to h which appears explicitly in (4.48) and implicitly in $u_\infty(h)$.

$\min_u \Psi(u)$, where $\Psi(u)$ plays the role of a natural potential function for this model. Moreover the minimum of the potential function yields the magnetization through Equation (4.41).

4.8 Mean field behaviour[13]

All the qualitative aspects of the phase transitions for the Curie-Weiss and tree-Ising models are identical. The precise location of their critical points and spinodal lines are different, but the qualitative nature of the first and second phase transitions are the same. In particular, the critical exponents (that govern the power law behavior of the magnetization at the critical point) take the *same* values. Note also that for the model on a tree the critical exponents do not depend on the coordination number $k \geq 3$. Given that the Curie-Weiss and tree-Ising models are on face value quite different, this might seem slightly surprizing.

In this respect, it is worth pointing out that there is a direct way to connect the Ising model on a tree with coordination number $k \to +\infty$ to the Curie-weiss model. To get a well defined limit when $k \to +\infty$ the coupling constant in the Hamiltonian is rescaled as $J \to J/k$. Then from (4.39) and (4.41) we recover the Curie-Weiss equation $m = \tanh(\beta h + \beta J m)$ when $k \to +\infty$ (see exercises). In this limit, not only the two models are qualitatively equivalent, but also their phase diagrams become rigorously identical.

There is a deeper and very generic way to understand why models whose solution is governed by a potential function and a fixed point equation *must*, under mild assumptions, have qualitatively identical phase diagrams. Such models are called "mean field models" and the arguments that we now outline constitute the basis of the "mean field theory" of phase transition. This generic theory goes back to Landau and is often also called "Landau theory."

Suppose that a spin model possesses a $\mathbf{Z}_2$ symmetry group[14] in the sense that the Hamiltonian is invariant under the transformation $h \to -h$, $s_i \to -s_i$, $i \in V$. This hypothesis holds for the Curie-Weiss and tree-Ising models (and also for the canonical Ising model on $\mathbf{Z}^d$). Suppose also that the solution of the model is governed by a fixed point equation

$$m = t(h, m)$$

for some smooth function $t(h, m)$, monotone increasing in h and m, and which depends also on β. This is true for the Curie-Weiss and tree-Ising models (but for the canonical Ising model on a square grid there is no such fixed point equation). Because of the $\mathbf{Z}_2$ symmetry, when (m, h) satisfies the fixed point equation then $(-m, -h)$ must also be a solution. Thus $t(h, m)$ must be an odd function of (h, m). In particular $t(0, m)$ is an odd function of m and thus $(h = 0, m = 0)$

[13] This section is not needed for the main development and can be skipped in a first reading.
[14] Here we stick to the simplest situation. Mean field theory can be developed for general symmetry groups and predictions will typically depend on the group.

must always be a trivial solution (for all β). Now suppose furthermore that there exist a second order transition with a critical point,[15] say $T = T_c$ and $h = 0$ (in our examples T_c is given by $\beta_c J = 1$ or $(k-1)\tanh(\beta_c J) = 1$). Then, by definition of the second order transition, the magnetization vanishes continuously at the critical point and for $T \uparrow T_c$, $h \to 0_\pm$ it is small. Therefore we perform a Taylor expansion

$$t(m,h) = a(T)m + b(T)m^3 - c(T)h + \dots$$

where only odd powers contribute since the function is odd (we have dropped terms of higher order $h^3, m^2h, mh^2, \dots$ which do not change the arguments below). Landau's theory makes mild assumptions on the temperature dependence of the coefficients in this expansion. We require that $m = 0$ is a triple solution at the critical point $(T = T_c, h = 0)$, which branches out into three distinct solutions $m_- < m_0 = 0 < m_+$ for $T < T_c$. This implies $a(T_c) = 1$ and $a(T) > 1$ for $T < T_c$. We therefore assume that generically $a(T) \approx 1 + a'(T_c)(T - T_c)$ with $a'(T_c) < 0$. Furthermore one can argue from statbility requirements that $b(T) > 0$ and $c(T) > 0$.

With these assumptions for $h = 0$ and $T \approx T_c$ the fixed point equation becomes

$$a'(T_c)(T - T_c)m + b(T_c)m^3 \approx 0$$

For $T > T_c$ the only solution is $m = 0$, and for $T < T_c$ we have two extra non trivial solutions $m \propto \pm(1 - T/T_c)^{1/2}$. Similarly at $T = T_c$ and $h \approx 0$ the fixed point equation becomes

$$b(T_c)m^3 - c(T_c)h \approx 0$$

which implies $m \propto \mathrm{sign}(h)|h|^{1/3}$. These power law behaviours have precisely the same critical exponents than those of the Curie-Weiss and tree-Ising models.

To summarize Landau's mean field theory assumes reasonable forms of the fixed point equation (or the potential function) close to the critical points, dictated by symmetry and stability requirements, and predicts critical exponents which are independent of the microscopic details of the underlying Hamiltonian (such as the coordination number k for example). One of the main developments in modern statistical mechanics has been the realization that this is also largely true in situations where Landau's mean field arguments break down. Let us give a glimpse of this last point.

Typically the range of validity of mean field theory is confined to models defined on high dimensional graphs for which the numbers of neighbords of a vertex grows fast enough with the graph-distance to the vertex (as is the case for the complete graph and the tree). For such systems one can neglect spatial fluctuations of the local magnetization and phase transitions are controlled by a fixed point equation involving a uniform magnetization. For finite dimensional models on regular grids (say $\mathbf{Z}^d$) there is no such fixed point equation that rigorously

[15] There are systems described by a fixed point equation which have only first order phase transitions and no critical points. An example is provided in exercise 4.6.

gives the true free energy. One has to take into account the fluctuations of the local magnetization into account. The *renormalization group* theory which takes into account fluctuations and yields correct critical exponents in low dimensions, scores as one of the big successes of statistical mechanics. Roughly speaking above some "critical dimension" $d \geq d_c$, fluctuations do not matter much, and mean field or Landau theory yields correct critical exponents independent of d (though the quantitative details of the phase diagram such as the precise value of the critical temperature are only approximate). Below this critical dimension $d < d_c$, fluctuations matter, Landau's theory breaks down, and the critical exponents depend on d. The renormalization group theory predicts that models can be classified in "universality classes" with identical critical exponents depending only on general features such as the symmetry group of the Hamiltonian and the dimensionality of space, but not on other details of the Hamiltonian. This "universality" explains why an oversimplified model such as the canonical Ising model in three spatial dimensions correctly predicts the critical exponents observed in very different physical systems such as alloys, fluids and magnets.

Despite its shortcomings Landau's theory (or extensions of it) often forms a good basis for the renormalization group treatment. The mean field analysis corresponds to an approximation that neglects fluctuations, but in doing so already gives a good idea of what the phase diagram look like.

4.9 Phase transitions in the canonical Ising model[16]

In Chapter 2 we introduced the canonical Ising model on a regular grid $\mathbb{Z}^d$. Models with a low dimensional regular underlying graph have *geometrical* features that are absent in the Curie-Weiss, tree-Ising, and in our three basic problems. It turns out the solutions and mathematical methods of analysis for low dimensional models are for the most part quite different than thsoe discussed here. Nevertheless there is value in briefly reviewing a few fundamental results on the canonical Ising model.

The one dimensional Ising model introduced by Lenz and Ising in 1925 can be exactly solved. For example one can apply the solution on a tree to the case $k = 2$, or one can use the "transfer matrix" method outlined in exercise 2.6. The free energy per spin as well as the magnetization in the thermodynamic limit are perfectly analytic for any $\beta^{-1} > 0$ and $h \in \mathbb{R}$. Therefore there are no phase transitions for any strictly positive temperature. In particular there is no spontaneous magnetization, i.e., $\lim_{h \to 0_\pm} m(\beta J, \beta h) = 0$ for all positive temperatures $\beta^{-1} > 0$.[17] Note that there is a "zero temperature" first order

[16] This section is not needed for the main development and can be skipped in a first reading.

[17] General theorems ensure that this is true for a wide class of one dimensional models with sufficiently short range interactions. If the coupling constant decays faster than $J_{ij} \sim |i-j|^{-\alpha}$, $\alpha > 2$ there is no phase transition at any positive temperature (Ruelle 1969)e.g.

phase transition in the sense that $\lim_{\beta\to+\infty} f(\beta, J, h) = -J - |h|$ which yields a spontaneous magnetization equal to ± 1 when $h \to 0_\pm$ (this can be seen from the expression of the free energy in exercise 2.6 or more directly by minimising the Hamiltonian).

In all dimensions $d \geq 2$ there are bona fide first and second order phase transitions at finite temperatures. This was not understood until the (essentially) rigorous proof by Peierls in 1930 who showed the existence of a phase transition in two dimensions. The phase diagram similar to Figure 4.6 in the (T, h) plane displays a coexistence line $(T < T_c, h = 0)$ which terminates at a critical point $(T = T_c, h = 0)$. Away from the coexistence line and critical point the free energy and magnetization are analytic.

The first order transition consists of a jump in the first derivative of the free energy across the coexistence line. Equivalently there is a non-vanishing spontaneous magnetization $m_\pm = \lim_{h\to 0_\pm} m(\beta J, \beta h)$ for $\beta > \beta_c$. Because of the $\mathbf{Z}_2$ symmetry of the model $m_- = -m_+$. The symmetry also implies that for $\beta < \beta_c$ where the free energy and magnetization are analytic we must have $\lim_{h\to 0_\pm} m = 0$.

At the critical point the first derivative of the free energy is continuous but the second derivative has a jump. More precisely the magnetization has power law behaviours governed by critical exponents commonly denoted β and δ, $m_\pm \propto \pm|1 - T/T_c|^\beta$ for $T \uparrow T_c$ and $m_\pm \propto \pm|h|^{1/\delta}$ for $h \to 0_\pm$ and $T = T_c$. As pointed out in the previous section the critical exponents depend on the dimension for $d < d_c$ and are given by the mean field exponents for $d > d_c$. For the Ising model $d_c = 4$.

In two dimensions for $h = 0$ the analytical expressions of the free energy and spontaneous magnetization are known and where first described by Onsager in his celebrated 1948 solution (the complete derivation of the spontaneous magnetization was explicitly given later by Yang in 1952). The spontaneous magnetization for $\beta > \beta_c$ is given by a remarkably simple exact formula

$$m_\pm(\beta) = |1 - (\sinh(2\beta J))^{-1}|^{1/8}$$

where the critical temperature is given by $\sinh(2\beta_c J) = 1$. It should be pointed out that we still do not know of an exact expression for the magnetization when $h \neq 0$. The main importance of Onsager's solution was to show that $m_\pm(\beta) \propto \pm|1-T/T_c|^{1/8}$ for $T \uparrow T_c$ with a critical exponent $1/8$. It is also known that in two dimensions $\delta = 15$. Thus the second order phase transition is not qualitatively identical to the Curie-Weiss one and the simple Landau theory outlined in the previous section does not apply.

The analytical solution of the three dimensional model is still an open problem. To compute the critical exponents one has to resort to the renormalisation group methods developed in the seventies. The most recent progress based on assumptions of scale invariance of the "magnetization field" near the critical point yields very precise critical exponents $\beta = 0.32642(2)$ and $\delta = 4.78982(7)$.

Similarly to the two dimensional case, the critical behaviour in three dimensions is different from the Curie-Weiss one and is not described by Landau's theory.

the renormalization group methods predict that for all dimensions $d \geq 4$ the critical exponents are the same than in the Landau theory. We have $\beta = 1/2$ and $\delta = 3$. It is beyond our scope to explain here why $d_c = 4$ constitutes the critical dimension above which the second order phase transition has mean field behaviour. Nevertheless it is not very difficult to understand why this is so when $d \to +\infty$, so let us discuss this point.

On $\mathbf{Z}^d$ the number of neighbords of a vertex is equal to $2d$ and the interaction term of a spin with its neighbords is of the form $-Js_i \sum_{j=1}^{2d} s_j$. For d very large we expect that the fluctuations of $\frac{1}{2d} \sum_{j=1}^{2d} s_j$ are negligible and that the sum *concentrates* over the magnetization m. The Hamiltonian effectively becomes for d very large

$$-2dJm \sum_{i=1}^{n} s_i - h \sum_{i=1}^{n} s_i \, .$$

This describes a system of independent spins in an effective magnetic field $2dJm + h$ and an elementary calculation shows that for this effective system the magnetization $m = \langle s_i \rangle$ satisfies the Curie-Weiss equation

$$m = \tanh(2d\beta Jm + \beta h) \, .$$

The arguments outlined here constitute what is called the "mean field approximation" for the Ising model. With a bit of work one can show that this approximation becomes *exact* when $d \to +\infty$ for an Ising model where we rescale the coupling constant $J \to J/2d$. In this limit one can truly neglect the fluctuations of the magnetization and $m = \tanh(\beta Jm + h)$ holds for the rescaled model. Thus, the Curie-Weiss model on a complete graph, the tree-Ising model with $k \to +\infty$ and the canonical Ising model with $d \to +\infty$ are all equivalent.

The smaller the dimension the worse the predictions of the Curie-Weiss equation. For "infinite" dimension all its predictions are correct. For finite dimensions the value of the critical temperature is wrong but the critical exponents are correct as long as $d \geq 4$ because fluctuations do not affect the power law behaviours close to the critical point. For three and two dimensions the Curie-Weiss critical exponents are wrong because fluctuations play a role in the second order phase transition. In one dimension the Curie-Weiss equation completely fails since it predicts a phase transition at finite temperatures, which we know does not happen.

4.10 Notes

The Ising model on a complete graph was first introduced as such by (Kac 1968) who named it the Curie-Weiss model. Marc Kac also introduced various other limits of the Ising model on high dimensional graphs or long range interactions

which are exactly solvable and have a "mean field" behaviour in the limit. We will in fact have the occasion to discuss similar limits when we introduce spatial coupling.

The Ising model on trees was analyzed at least since the 50's in various works that clarified the role of the "boundary" leaves of the tree. In the physics literature a distinction is made between the so-called "Cayley tree" and "Bethe lattice" models. Both are Ising models on a tree but the former computes the partition function and free energy of the tree taking into account the leaves at the "boundary", and the second eliminates boundary effects by computing the magnetization of the root. It is the second setting that we discussed in this chapter because it is the one that is relevant for us. Indeed, when we go to random graphs there are no boundary leaves.

Exactly solvable models of classical statistical mechanics and more generally rigorous results have played, and still play, an important role in the investigations of the nature of various phase transitions. There are two essential categories of solvable models: one and two dimensional models on regular grids solved by the "transfer matrix" method (the main idea of this method is presented in exercise 2.6 for one space dimension) and infinite dimensional models. The infinite dimensional case, by which we mean models on complete graphs, trees and random graphs, is very relevant in coding, compressive sensing and satisfiability. A classic reference for the beautiful topic of exactly solvable models is (Baxter 1982). The Curie-Weiss and tree-Ising mode, their fixed point equations and phase diagrams, are analyzed in detail in that reference. A rigorous derivation of the free energy of the Ising model on random sparse graphs can be found in (Dembo & Montanari 2010).

The first exact solution of the two-dimensional canonical Ising model for zero field and the calculation of the spontaneous magnetization is due to (Onsager 1944) and (Onsager 1952). This solution showed for the first time that one had to go beyond mean field theories to corrcetly calculate critical exponents in low dimensions. See (McCoy & Wu 1973) for a classic book on the two dimensional Ising model. There are numerous other rigorous results about the canonical Ising model and its variations that we have not reviewed at all. Let us only mention that while it was clear from the very beginning that the one-dimensional Ising model has no phase transition (see exercises 2.6 and 4.2), and more generally that phase transitions do not occur in one dimension for short range interactions, the higher dimensional case $d \geq 2$ was more mysterious. It is only after the famous argument of (Peierls 1936) that the existence of phase transitions in Ising models for $d \geq 2$ was recognized. The mathematical proofs of the existence of phase transitions, were established in important papers by (Griffiths 1964) and (Dobrushin 1965). The rigorous theory of phase transitions for finite dimensional models on regular graphs then became a discipline of its own (e.g Ruelle 1969, Simon 1993).

The Curie-Weiss equation itself goes back to the works of Curie and Weiss on the para and ferro-magnetic states. The "molecular field hypothesis" of (Weiss

1907) was a crucial step for the development of the mean field approach to the theory of phase transitions. The mean field approach was developped for alloys by (Bragg & Williams 1934) and then improved in a famous paper of (Bethe 1935) who Bethe investigated corrections to Weiss's theory. These corrections are important for sparse random and tree like graphs, but also for random interactions on complete graphs. As we will discover they are naturally embodied in message passing algorithms.

There are deep similarities between phase transitions in very different systems such as alloys, fluids or magnets, and which where explained by Landau's mean field theory (Landau 1937). A very good historical account of these developments as well as our introduction to mean field theory is provided by (Kadanoff 2009). Extensive treatments can be found in numerous books (e.g Stanley 1971, Chaikin & Lubensky 2007). The renormalization group treatment that takes into account fluctuations beyond mean filed theory can be found in many classic textbooks (e.g Ma 1976, Huang 1987, Goldenfeld 1993).

Problems

4.1 $2p$-SPIN MODEL ON A COMPLETE GRAPH. Consider a set of n spins with Hamiltonian

$$\mathcal{H}(\underline{s}) = -\frac{J}{(2p-1)!N^{2p-1}} \sum_{i_1 \neq i_2 \neq \cdots \neq i_{2p}} s_{i_1} s_{i_2} \dots s_{i_{2p}} - h \sum_{i=1}^{n} s_i$$

where $p \geq 1$ is an integer and the first sum carries over $2p$-tuples with distinct indices. For $p = 1$ this is the Curie-Weiss model. Repeat the calculations of the usual Curie-Weiss model to obtain a variational expression for the free energy and show that the magnetization satisfies the fixed point equation

$$m = \tanh(\beta J m^{2p-1} + \beta h)$$

Determine the phase diagram. Show in particular that for $p \geq 2$ there is *only* a first order phase transition (and no second order phase transition).

4.2 ONE DIMENSIONAL ISING MODEL. Apply the solution of the Ising model on a tree to $k = 2$ (a line) and that there is no phase transition for any strictly positive temperature. Check that one recovers the same solution than the one obtained from the transfer matrix method in exercise 2.6.

4.3 CRITICAL EXPONENTS OF THE ISING MODEL ON A TREE. Analyze the fixed point equation for this model close to the critical point. Calculate the critical exponents governing the power law behaviour of the magnetization.

4.4 SPINODAL LINES. Consider the Ising model on a tree. Calculate analytically the spinodal points $h_{\rm sp}$ and $m_{\rm sp}$ as a function of the temperature. Plot the spinodal lines in the (T, h) and (T, m) planes.

4.5 FIXED POINT EQUATIONS FOR ISING MODEL ON A TREE. Use (4.39) and (4.41) to deduce the fixed point equation for the magnetization $m_o(\beta, h)$ of the

central node,

$$m_o = \tanh\big\{\beta h + \beta y(m; \beta h, \beta J)\big\}. \tag{4.49}$$

with

$$y(m_o; \beta h, \beta J) = \beta^{-1} k \operatorname{atanh}\left(m_o \frac{\tanh(\beta J) + (\tanh((\beta h - \operatorname{atanh} m_o)/k))^2}{\tanh(\beta J) + 1} \right).$$

Rescale the coupling constant $J \to J/k$ and check that this equation reduces to the Curie-Weiss one in the limit of infinite coordination number $k \to +\infty$.

4.6 $2p$-SPIN MODEL ON A TREE. Consider a tree factor graph with variable nodes of degree k and constraint nodes of even degree $2p$. There are L levels of factor nodes and each factor node contributes a term $-Js_{i_1} s_{i_2} \dots s_{i_{2p}}$ (for $i_1, i_2, \dots, i_{2p}$ attached to the factor node). For $p = 1$ this is the usual tree-Ising model. Generalize the calculations of section 4.6 and deduce the analog of the iterative equations (4.39) and (4.41). Show that for $p \geq 2$ there is only a first order phase transition (and no second order phase transition).

Part II

Analysis of Message Passing Algorithms

5 Marginalization and Sum-Product Equations

We have seen that computing the marginals of the Gibbs distributions is a central problem. For example in coding and compressed sensing the tasks of decoding and signal estimation can both be reduced to the determination of a "magnetization" which in turn is easy to obtain once we know the marginals. Unfortunately, for general Gibbs distributions computing marginals is an intractable problem. Nevertheless all is not lost, much to the contrary. Indeed, we saw in Chapter 1 that the factor graphs of our models are always either locally tree like (coding and K-SAT) or complete (compressive sensing); and in Chapter 4 we learned how to exactly solve two simple Ising models, on the tree and the complete graph, which are toy versions of our more ambitious models.

In this chapter we will concentrate on an *efficient* calculation of marginals for the case where the factor graph is a *tree*. The emphasis here is on the word "efficient". We will see that this question has a natural answer in the form of a message-passing algorithm. The message-passing paradigm is the basis for the *low-complexity* algorithms which we will apply to our problems even when the factor graph *is not* a tree. There is a price to pay on non-tree graphs because low-complexity algorithms do not necessarily perform exact marginalization is a priori not exact. Therefore our low complexity message passing algorithms are *suboptimal* in the sense that they do not give correct solutions up to the "optimal" thresholds. For example message passing decoders do not work up to the MAP threshold of the code ensemble; K-SAT solvers based on message passing find solutions only for densities α quite smaller than the satsifiability threshold α_s. In the analysis of message passing we will find *algorithmic thresholds* which are smaller (i.e. worse) than the optimal thresholds.

There is a surprise however. Message-passing algorithms are also the key for the analysis of the optimal thresholds and phase transitions of our three examples. A priori it is not obvious that there should be any connection between optimal thresholds and low-complexity algorithms. For example optimal thresholds are non-differentiability points of the free energy, but algorithmic thresholds are not visible on the free energy (since away from phase transition points this quantity is analytic). Nevertheless these two worlds are connected as we will see in the third part of our lectures. Quite remarkably one can also go one step further. In Chapter 14 we will consider a class of ensembles - called spatially coupled ensembles - for which the optimal and algorithmic thresholds may even be equal.

For these ensembles the low complexity message passing methods work all the way up to the optimal thresholds and allow optimal solutions!

So far we have associated a factor graph to Hamiltonians or cost functions. In the next section this idea is taken a little bit further by associating the factor graph to the Gibbs distribution itself. We then use this representation to help organize the marginalization on trees and derive the message passing algorithm. As we will see, on trees marginalization ultimately boils down to an application of a distributive law of multiplication and addition. Finally we illustrate through simple examples how the formalism is applied to our three problems.

5.1 Factor graph representation of Gibbs distributions

One important characteristic of the Gibbs distributions of our three problems is its *factorized form*. Generically

$$p(\underline{x}) = \frac{1}{Z}\prod_{c=1}^{m} f_c(\underline{x}_{\partial c}), \qquad Z = \sum_{\underline{x}\in\mathcal{X}^n}\prod_{c=1}^{m} f_c(\underline{x}_{\partial c}) \tag{5.1}$$

where $\underline{x}_{\partial c}$ is the set (or vector) of variables x_i entering as arguments of the factors f_c.

The simplest incarnation of this factorization occurs in K-SAT (see (3.57)) where in spin language $x_i \to s_i = (-1)^{x_i}$ and the alphabet is $\mathcal{X} = \{-1, +1\}$ and the factors are $f_a(\underline{s}_{\partial a}) = \exp\{-\beta\prod_{i\in\partial a}\frac{1}{2}(\frac{1+s_i J_{ia}}{2})\}$. For coding (see Equ. (3.10)) we have two types of factors $f_i(s_i) = e^{h_i s_i}$ and $f_a(\underline{s}_{\partial a}) = \frac{1}{2}(1+\prod_{i\in\partial a} s_i)$. For compressed sensing (see Equ. (3.43)) the alphabet is continuous $\mathcal{X} = \mathbb{R}$ so in (5.1) the sums must be interpreted as integrals $\int d^n\underline{x}$ and there are two types of factors $f_i(x_i) = (p_0(x_i))^\beta$ and $f_a(\underline{x}_{\partial a}) = e^{-\frac{\beta}{2\sigma^2}(y_a - \underline{A}_a^T\underline{x})^2}$. Analogous identifications for general Ising models of Chapter 2 and also for the Curie-Weiss model are left as an exercise. Note that the factorization is not unique, but usually it is pretty clear how to find a natural one.

From now on we will focus on a generic factorization (5.1) and we come back to specific illustrations in Sections 5.4, 5.5 and 5.6. We associate with this factorization a *factor graph* which is mildly different from the ones introduced in Chapter 1. For each variable x_i draw a *variable node* (circle) and for each factor f_c draw a *factor node* (square). Connect a variable node to a factor node by an *edge* if and only if the corresponding variable appears in this factor.

EXAMPLE 11 (Simple Example) Let's start with an example that will serve as our running example. Consider a distribution with factorization

$$p(x_1, x_2, x_3, x_4, x_5, x_6) = \frac{1}{Z} f_1(x_1, x_2, x_3) f_2(x_1, x_4, x_6) f_3(x_4) f_4(x_4, x_5). \tag{5.2}$$

The resulting graph for this distribution is shown on the Figure 5.1. □

The factor graph is *bipartite*. This means that the set of vertices is partitioned

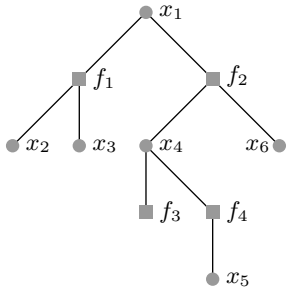


Figure 5.1 Factor graph of f given in Example 11.

into two groups, the set of nodes corresponding to variables and the set of nodes corresponding to factors, and moreover that an edge only connects a variable node to a factor node. For our particular example the factor graph is a (bipartite) *tree*. This means that there are no *cycles* in the graph; i.e., there is one and only one path between each pair of nodes.

As we will show in the next section, for factor graphs that are trees marginals can be computed efficiently by *message-passing* algorithms.

5.2 Marginalization on trees

We first remark that in order to carry out the marginalization in practice one can first ignore the partition function Z. Indeed suppose that we want to compute the marginal $\nu_1(x_1)$ for (5.1) (recall definition (2.24)). If we first compute the "marginal" of the numerator only[1]

$$\mu_1(x_1) = \sum_{\sim x_1} \prod_c f_c(x_{\partial c}), \tag{5.3}$$

then clearly $\nu_1(x_1) = \mu(x_1)/Z \propto \mu_1(x_1)$. So the only difference between $\nu_1(x_1)$ and $\mu_1(x_1)$ is a proportionality factor which serves to normalize the marginal. Thus, assuming that we are able to compute $\mu(x_1)$, we simply get the marginal by normalizing

$$\nu_1(x_1) = \frac{\mu_1(x_1)}{\sum_{x_1 \in \mathcal{X}} \mu_1(x_1)}. \tag{5.4}$$

This last step is an easy task that involves only one sum or an integral in the denominator. Note also that $Z = \sum_{x_1} \mu_1(x_1)$.

In the following and also in practice we just deal with the "marginalization" of the numerator and normalize the result in the very last step.

[1] Recall that the notation $\sum_{\sim x_1}$ means "summation over all variables except x_1" which remains fixed.

Distributive Law

On trees marginalization can be achieved by a careful application of the distributive law. Let $\mathbb{F}$ be a field (think of $\mathbb{F} = \mathbb{R}$) and let $a, b, c \in \mathbb{F}$. The *distributive law* states

$$ab + ac = a(b + c). \tag{5.5}$$

This simple relation, properly applied, can significantly reduce computational complexity. Consider for example the evaluation of

$$\sum_{i,j=1}^{n} a_i b_j \quad \text{as} \quad (\sum_{i=1}^{n} a_i)(\sum_{j=1}^{n} b_j)\,.$$

Instead of n^2 multiplications and n^2-1 additions we perform 1 multiplication and $2n$ additions. Factor graphs provide an appropriate framework to take advantage of the distributive law in a systematic way.

Let's start with Example 11. The numerator of p is a function f with factorization

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = f_1(x_1, x_2, x_3) f_2(x_1, x_4, x_6) f_3(x_4) f_4(x_4, x_5). \tag{5.6}$$

We are interested in computing the "marginal" of f with respect to x_1

$$\mu_1(x_1) = \sum_{\sim x_1} f(x_1, x_2, x_3, x_4, x_5, x_6).$$

What is the complexity of a brute force computation? Assume that all variables take values in a finite alphabet $\mathcal{X}$. Determining $\nu(x_1)$ for all values of x_1 by brute force requires $\Theta\left(|\mathcal{X}|^6\right)$ operations, where we assume a naive computational model in which all operations (addition, multiplication, function evaluations, etc.) have the same cost. But we can do better: taking advantage of the factorization, we can rewrite $\nu(x_1)$ as

$$\mu(x_1) = \Big[\sum_{x_2,x_3} f_1(x_1, x_2, x_3)\Big]\Big[\sum_{x_4} f_3(x_4)\Big(\sum_{x_6} f_2(x_1, x_4, x_6)\Big)\Big(\sum_{x_5} f_4(x_4, x_5)\Big)\Big].$$

Fix x_1. The evaluation of the first square bracket can be accomplished with $\Theta\left(|\mathcal{X}|^2\right)$ operations. The second square bracket depends only on x_4, x_5, and x_6. It can be evaluated efficiently in the following manner. For each value of x_4 (and x_1 fixed), determine $\sum_{x_5} f_4(x_4, x_5)$ and $\sum_{x_6} f_2(x_1, x_4, x_6)$. Multiply by $f_3(x_4)$ and sum over x_4. Therefore, the evaluation of the second bracket requires $\Theta\left(|\mathcal{X}|^2\right)$ operations as well. Since there are $|\mathcal{X}|$ values for x_1, the overall task has complexity $\Theta\left(|\mathcal{X}|^3\right)$. This compares favorably to the complexity $\Theta\left(|\mathcal{X}|^6\right)$ of the brute force approach.

Recursive Determination of Marginals

Consider the factorization of a generic function g (for example the numerator of a Gibbs distribution (5.1)) and suppose that the associated factor graph is a (bipar-

tite) tree. Suppose that we are interested in marginalizing g with respect to the variable z; in other words we are interested in computing $\mu(z) = \sum_{\sim z} g(z, \dots)$. Since the factor graph of g is a bipartite tree, g has a generic factorization of the form

$$g(z, \dots) = \prod_{k=1}^{K} [g_k(z, \dots)] \tag{5.7}$$

for some integer K with the following crucial property: z appears in each of the factors g_k, but all other variables appear in *only one* factor. To see this assume to the contrary that another variable is contained in two of the factors. This implies that besides the path that connects these two factors via variable z, another path must exist. But this contradicts the assumption that the factor graph is a tree.

For the function f of Example 11 this factorization is

$$f(x_1, \dots) = [f_1(x_1, x_2, x_3)]\,[f_2(x_1, x_4, x_6) f_3(x_4) f_4(x_4, x_5)]\,,$$

so that $K = 2$. The generic factorization and the particular instance for our running example f are shown in Figure 5.2. Taking into account that the indi-

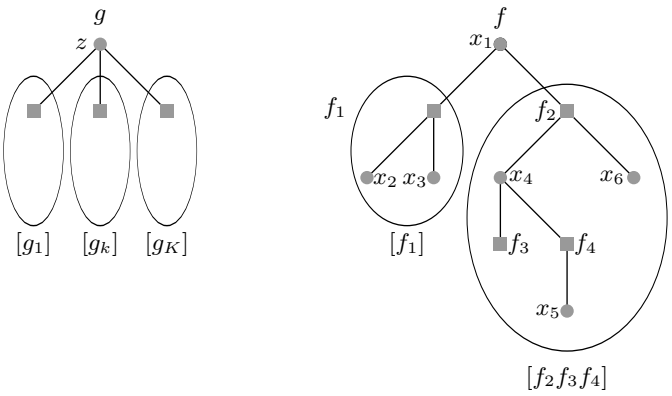


Figure 5.2 Generic factorization and the particular instance.

vidual factors $g_k(z, \dots)$ in (5.7) only share the variable z, an application of the distributive law leads to

$$\mu(z) = \sum_{\sim z} g(z, \dots) = \sum_{\sim z} \prod_{k=1}^{K} [g_k(z, \dots)] = \prod_{k=1}^{K} \Big[\sum_{\sim z} g_k(z, \dots) \Big]. \tag{5.8}$$

In words, the marginal $\sum_{\sim z} g(z, \dots)$ is the product of the individual marginals $\sum_{\sim z} g_k(z, \dots)$. In terms of our running example we have

$$\mu_1(x_1) = \Big[\sum_{\sim x_1} f_1(x_1, x_2, x_3) \Big] \Big[\sum_{\sim x_1} f_2(x_1, x_4, x_6) f_3(x_4) f_4(x_4, x_5) \Big].$$

This single application of the distributive law leads, in general, to a non-negligible

reduction in complexity. But we can go further and apply the same idea recursively to each of the terms $g_k(z, \dots)$.

In general, each g_k is itself a product of factors. In Figure 5.2 these are the factors of g that are grouped together in one of the ellipsoids. Since the factor graph is a bipartite tree, g_k must in turn have a generic factorization of the form

$$g_k(z, \dots) = h(z, z_1, \dots, z_J) \prod_{j=1}^{J} [h_j(z_j, \dots)] ,$$

where z appears only in the "kernel" $h(z, z_1, \dots, z_J)$ and each of the z_j appears at most twice, possibly in the kernel and in at most one of the factors $h_j(z_j, \dots)$. All other variables are again unique to a single factor. For our running example we have

$$f_2(x_1, x_4, x_6) f_3(x_4) f_4(x_4, x_5) = f_2(x_1, x_4, x_6) \, [f_3(x_4) f_4(x_4, x_5)] \, [1] .$$

where $f_2(x_1, x_4, x_6)$ is the kernel and the factors are $[f_3(x_4) f_4(x_4, x_5)]$ and $[1]$. The generic factorization and the particular instance for our running example are shown in Figure 5.3. Another application of the distributive law gives

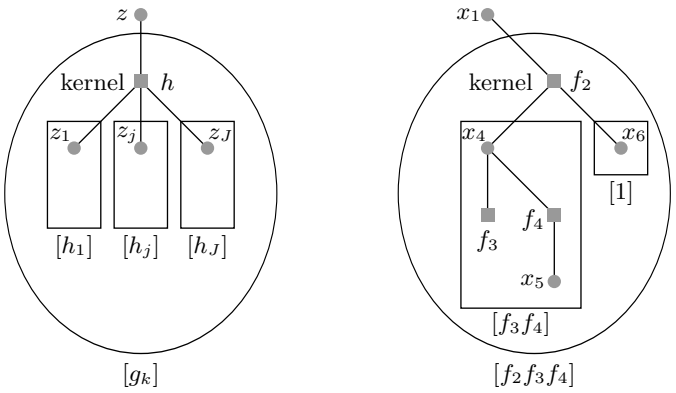


Figure 5.3 Generic factorization of g_k (left) and the particular instance (right).

$$\begin{aligned} \sum_{\sim z} g_k(z, \dots) &= \sum_{\sim z} h(z, z_1, \dots, z_J) \prod_{j=1}^{J} [h_j(z_j, \dots)] \\ &= \sum_{\sim z} h(z, z_1, \dots, z_J) \prod_{j=1}^{J} \Big[\sum_{\sim z_j} h_j(z_j, \dots) \Big] . \end{aligned} \tag{5.9}$$

In words, the desired marginal $\sum_{\sim z} g_k(z, \dots)$ can be computed by multiplying the kernel $h(z, z_1, \dots, z_J)$ with the product of individual marginals $\sum_{\sim z_j} h_j(z_j, \dots)$ and summing out all remaining variables other than z.

We are back to where we started. Each factor $h_j(z_j, \dots)$ has the same generic form as the original function $g(z, \dots)$, so that we can continue to break down the

marginalization task into smaller pieces. This recursive process continues until we have reached the leaves of the tree. The calculation of the marginal then follows the recursive splitting in reverse. In general, nodes in the graph compute marginals, which are functions over $\mathcal{X}$, and pass these on to the next level. The message combining rules at function nodes is explicit in (5.9). And at a variable node we simply perform pointwise multiplication. In the next section we will elaborate on this method of computation known as message passing.

Let us consider the initialization of the process. At the leaf nodes the task is simple. A function leaf node has the generic form $g_k(z)$, so that $\sum_{\sim z} g_k(z) = g_k(z)$. This means that the initial message sent by a function leaf node is the function itself. To find out the correct initialization at a variable leaf node consider the simple example of computing $\sum_{\sim x_1} f(x_1, x_2)$. Here, x_2 is the variable leaf node. By the message-passing rule (5.9) the marginal is equal to $\sum_{\sim x_1} f(x_1, x_2) \cdot \mu(x_2)$, where $\mu(x_2)$ is the initial message that we send from the leaf variable node x_2 towards the kernel $f(x_1, x_2)$. We see that to get the correct result this initial message should be the constant function 1.

5.3 Marginalization via Message Passing

In the previous section we have seen that, in the case where the factor graph is a tree, the marginalization problem can be broken down into smaller and smaller tasks according to the structure of the tree.

This gives rise to the following efficient *message-passing* algorithm. The algorithm proceeds by sending messages along the edges of the tree. Messages are functions on $\mathcal{X}$, or, equivalently, vectors of length $|\mathcal{X}|$. Message passing originates at the leaf nodes, messages are passed up the tree and, as soon as a node has received the messages from all its children, they are processed and the result is passed up to the parent node. Finally they are combined to form the marginal of the whole function.

EXAMPLE 12 (Message-Passing Algorithm for f of Example 11) Consider this procedure in detail for the case of our running example as shown in Figure 5.4. The top leftmost graph is the factor graph. Message passing starts at the leaf nodes as shown in the middle graph on the top. The variable leaf nodes x_2, x_3, x_5, and x_6 send the constant function 1 as discussed at the end of the previous section. The factor leaf node f_3 sends the function f_3 up to its parent node. In the next time step the factor node f_1 has received messages from both its children and can therefore proceed. According to (5.9), the message it sends up to its parent node x_1 is the product of the incoming messages times the "kernel" f_1, after summing out all variable nodes except x_1. This message is $\sum_{\sim x_1} f_1(x_1, x_2, x_3)$. In the same manner factor node f_4 forwards to its parent node x_4 the message $\sum_{\sim x_4} f_4(x_4, x_5)$. This is shown in the rightmost figure in the top row. Now, variable node x_4 has received messages from all its children. It

forwards to its parent node f_2 the product of its incoming messages, in agreement with (5.8), which says that the marginal of a product is the product of the marginals. This message, which is a function of x_4, is $f_3(x_4)\sum_{\sim x_4} f(x_4, x_5) = \sum_{\sim x_4} f_3(x_4)f_4(x_4, x_5)$. Next, function node f_2 can forward its message, and, finally, the marginalization is achieved by multiplying all incoming messages at the root node x_1. □

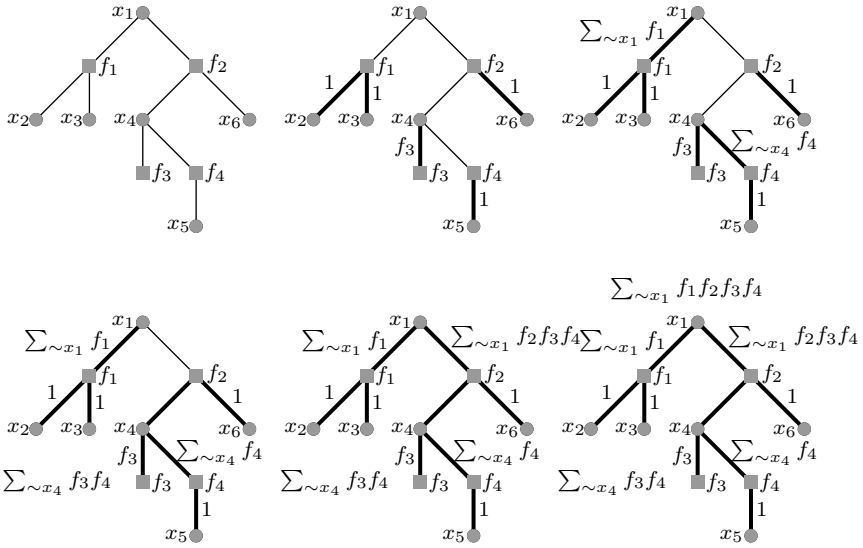


Figure 5.4 Marginalization of function f from Example 11 via message passing. Message passing starts at the leaf nodes. A node that has received messages from all its children processes the messages and forwards the result to its parent node. Bold edges indicate edges along which messages have already been sent.

Complexity of message passing

Before stating the message-passing rules formally, consider the following important generalization. Whereas so far we have considered the marginalization of a function f with respect to a single variable x_1 we are actually interested in marginalizing for all variables. We have seen that a single marginalization can be performed efficiently if the factor graph of f is a tree, and that the complexity of the computation essentially depends on the largest degree of the factor graph and the size of the underlying alphabet. Consider now the problem of computing *all* marginals. We could draw for each variable a tree rooted in this variable and execute the single marginal message-passing algorithm on each rooted tree. It is easy to see, however, that the algorithm does not depend on which node is the root of the tree and that in fact all the computations can be performed simultaneously. Simply start at all leaf nodes and for every edge compute the outgoing message along this edge as soon as you have received the incoming messages

along all other edges that connect to the given node. Continue in this fashion until a message has been sent in *both directions* along every edge. This computes *all* marginals so it is more complex than computing a single marginal but only by a factor roughly equal to the average degree of the nodes. We now formalise this discussion.

Belief propagation equations

Messages flow on edges in both directions. Messages from variables nodes to factor nodes are denoted $\mu_{i\to c}$, and messages from function nodes to variable nodes $\widehat{\mu}_{c\to i}$. The letters $a, b, c, \ldots$ are reserved for factor nodes and $i, j, k, \ldots$ for variable nodes. Although this may sometimes be redundant notation, in order to avoid confusions it is convenient to reserve μ for messages from variable nodes to factor nodes and $\widehat{\mu}$ for messages from factor nodes to variable nodes. Marginals, once normalized, will be denoted by ν. Messages and marginals are functions on $\mathcal{X}$ and for finite alphabets it is sometimes useful to think of them as vectors with $|\mathcal{X}|$ components.

Message passing starts at leaf nodes. Consider a node and one of its adjacent edges, call it e. As soon as the incoming messages to the node along all other adjacent edges have been received these messages are processed and the result is sent out along e. This process continues until messages along all edges in the tree

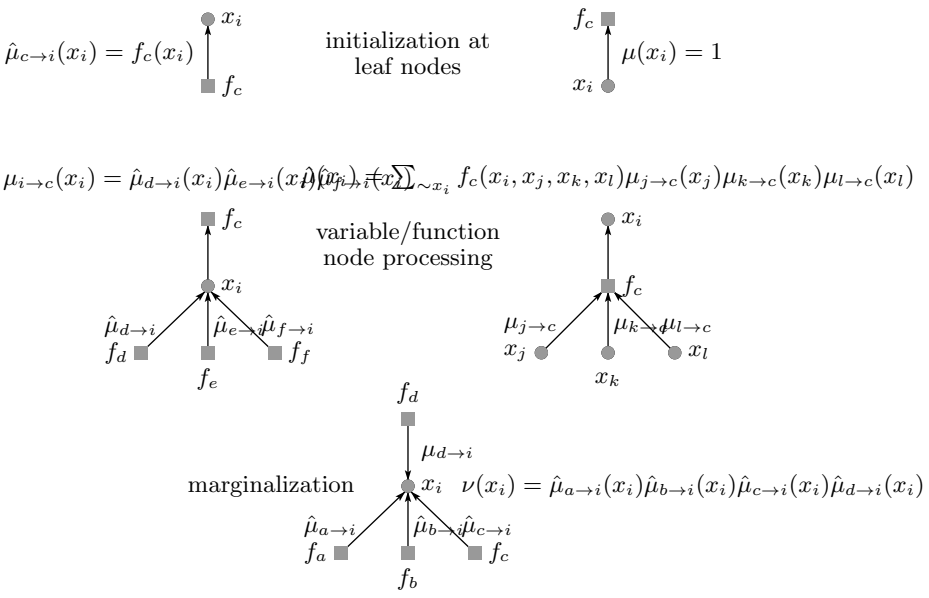


Figure 5.5 Message-passing rules. The top row shows the initialization of the messages at the leaf nodes. The middle row corresponds to the processing rules at the variable and function nodes, respectively. The bottom row explains the final marginalization step.

have been processed. In the final step the marginals are computed by combining all messages which enter a particular variable node. The initial conditions and processing rules are summarized in Figure 5.5. Since the messages represent (unormalized) probabilities or *beliefs*, the algorithm is also known as the *Belief Propagation* (BP) algorithm. From now on we will mostly refer to it under this name.

We sumarize the BP relations here for further reference[2]

$$\begin{cases} \mu_{i\to a}\left(x_i\right) & = \prod_{b\in\partial i\smallsetminus a} \widehat{\mu}_{b\to i}\left(x_i\right), \\ \widehat{\mu}_{a\to i}\left(x_i\right) & = \sum_{\sim x_i} f_a\left(x_{\partial a}\right) \prod_{j\in\partial a\smallsetminus i} \mu_{j\to a}\left(x_j\right). \end{cases} \tag{5.10}$$

with the proviso that at leaf nodes $\mu_{i\to c}(x_i) = 1$ and $\widehat{\mu}_{c\to i}(x_i) = f_c(x_{\partial c})$. The marginals are obtained as

$$\begin{cases} \nu_i\left(x_i\right) = \frac{\prod_{a\in\partial i} \widehat{\mu}_{a\to i}(x_i)}{\sum_{x_i} \prod_{a\in\partial i} \widehat{\mu}_{a\to i}(x_i)} \\ \nu_a\left(x_{\partial a}\right) = \frac{f_a(x_{\partial a}) \prod_{i\in\partial a} \mu_{i\to a}(x_i)}{\sum_{x_{\partial a}} f_a(x_{\partial a}) \prod_{i\in\partial a} \mu_{i\to a}(x_i)}. \end{cases} \tag{5.11}$$

When we compute the marginals it is not important how the μ and $\widehat{\mu}$ messages are normalized because in (5.11) the normalizations cancel out. We will sometimes exploit this fact and write (5.10) as proportionality relations. This often simplifies many calculations.

Algorithmic versus static point of view

As explained, BP relations allow to compute exact marginals on trees. By starting the process at leaf nodes we are sure that it converges in a finite number of steps to the exact marginals. On non-tree graphs the situation is not as simple because this process usually *does not* yield exact marginals. There, the BP relations form the basis of an algorithm which outputs *BP-marginals* (not necessarily equal to true marginals) which are used to make decisions about the decoded bit, signal estimate, etc. To run the algorithm we have to decide on an initial condition, schedule, and a running time. These aspects will be clarified separately for each problem in subsequent chapters.

In the third part of these notes the BP equations will be used in a "statistical mechanics" non-algorithmic way, namely as *fixed point* equations. We will see that the fixed point form of the BP equations arises when the so-called *Bethe free energy* is minimised, much as the Curie-Weiss fixed point equation appeared in Chapter 4 when we minimized the potential function. This point of view will become key when we relate low complexity algorithms to optimal solutions.

[2] These relations are also called *Sum-Product* equations, a name that reflects their algebraic structure.

5.4 Message Passing in Coding

Assume we transmit over a binary-input memoryless channel using a linear code. Recall the formulation in Chapter 3: the rule (3.14) for the bit-wise maximum a posteriori (MAP) decoder reads $\widehat{s}_i(\underline{h}) = \operatorname{argmax}_{s_i \in \{\pm 1\}} \nu_i(s_i|h) = \operatorname{sign}\langle s_i \rangle$ which is immediate to compute once we have $\nu_i(s_i|\underline{h})$, the marginal of distribution (3.10). So we have to marginalise the numerator of

$$p(\underline{s}|\underline{h}) = \frac{1}{Z} \prod_{a=1}^{m} \frac{1}{2}(1 + \prod_{i \in \partial a} s_i) \prod_{i=1}^{n} e^{h_i s_i}. \tag{5.12}$$

and eventually normalize the resulting function of s_i. The numerator of (5.12) has a factorized form with two types of factors,

$$f_i(s_i) = e^{h_i s_i} \qquad \text{and} \qquad f_a(\underline{s}_{\partial a}) = \frac{1}{2}(1 + \prod_{i \in \partial a} s_i),$$

which are associated to square nodes in the factor graph representation of (5.12). The first factor is attached in the factor graph to a single bit and describes the influence of the channel. The second one is attached to several bits and describes the parity-check constraints.

EXAMPLE 13 (Bit-wise MAP decoding) Consider the code defined by the parity-check matrix with Tanner graph shown on the left of Fig. 5.6.

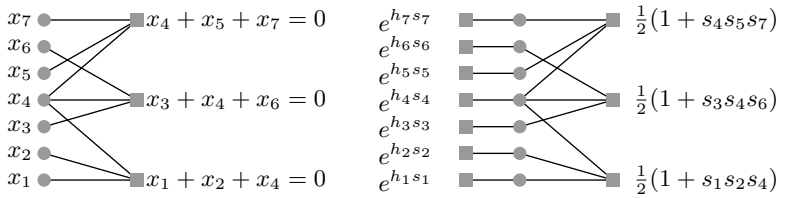


Figure 5.6 Left: graphical representation of a simple parity check code. Note that the factor graph is the same as the one of example 11. Right: factor graph associated to the Gibbs distribution (5.12).

The factor graph corresponding to the distribution (5.12) is shown on the right of this figure. It includes the (Tanner) graph of the parity check code, but additionally contains extra factor nodes which represent the effect of the channel. For this particular case the resulting graph is a tree. We can therefore apply the message-passing algorithm to this example to perform exact bit-wise MAP decoding. □

In principle the messages are uniquely specified by the general message-passing rules and we could simply move on to the next example. Indeed, the real power of the factor graph approach lies in the fact that, once the graph and the factors are specified, no thought is required to work out the messages. For the current

example perhaps the result is quite intuitive and this might seem as no big deal. But in "real life" systems substantially more complicated factor graphs are encountered and in such cases without the message passing rules it might be quite difficult to figure out how to correctly combine messages. Despite the fact that we could just blindly follow the rules, it is instructive to explicitly work out a few steps of the belief propagation algorithm for this example.

EXAMPLE 14 (Message passing algorithm for decoding) We give the first three steps of belief propagation for the tree in Figure 5.6. In the first step the initial messages are sent from leaf nodes. Here all leaf nodes are factor nodes whose factor is the prior, thus the initial messages are $\widehat{\mu}_{k\to k}(s_k) = e^{h_k s_k}$ for $k = 1, \ldots, 7$. At the second step six variable nodes send messages to factor nodes, namely the variable nodes that participate in only a single parity-check constraint: $\mu_{1\to 1}(s_1) = e^{h_1 s_1}$, $\mu_{2\to 1}(s_2) = e^{h_2 s_2}$, $\mu_{3\to 2}(s_3) = e^{h_3 s_3}$, $\mu_{5\to 1}(s_5) = e^{h_5 s_5}$, $\mu_{6\to 2}(s_6) = e^{h_6 s_6}$, $\mu_{7\to 1}(s_7) = e^{h_7 s_7}$. At the third step the three factor nodes have received all their input, except the input from variable node 4. Hence, they can send back their messages in direction of node 4. These are

$$\widehat{\mu}_{1\to 4}(s_4) = \sum_{s_1, s_2} \frac{1}{2}(1 + s_1 s_2 s_4) e^{h_1 s_1} e^{h_2 s_2},$$

$$\widehat{\mu}_{2\to 4}(s_4) = \sum_{s_3, s_6} \frac{1}{2}(1 + s_3 s_4 s_6) e^{h_3 s_3} e^{h_6 s_6},$$

$$\widehat{\mu}_{3\to 4}(s_4) = \sum_{s_5, s_7} \frac{1}{2}(1 + s_4 s_5 s_7) e^{h_5 s_5} e^{h_7 s_7}.$$

The sums involved in the messages are easy to compute. For example using $e^{h_i s_i} = \cosh h_i (1 + s_i \tanh h_i)$ the first one is equal to

$$\widehat{\mu}_{1\to 4}(s_4) = (2 \cosh h_1 \cosh h_2)(1 + s_4 \tanh h_1 \tanh h_2)\,.$$

Looking at one more step, note that at this point all incoming messages to variable node 4 are known and so we can compute the marginal $\mu_4(s_4)$ (of the numerator of (5.12)) by multiplying all messages incoming into variable node 4. Explicitly,

$$\begin{aligned}\mu_4(s_4) =& (2 \cosh h_4)(1 + s_4 \tanh h_4)(2 \cosh h_1 \cosh h_2)(1 + s_4 \tanh h_1 \tanh h_2) \\ &\times (2 \cosh h_3 \cosh h_6)(1 + s_4 \tanh h_3 \tanh h_6) \\ &\times (2 \cosh h_5 \cosh h_7)(1 + s_4 \tanh h_5 \tanh h_7)\,.\end{aligned}$$

To get the true marginal $\nu_4(s_4) = \nu_4(s_4|\underline{h})$ one has to normalize $\mu(s_4)$,

$$\nu_4(s_4|\underline{h}) = \frac{\mu_4(s_4)}{\mu_4(1) + \mu_4(-1)}\,.$$

To obtain the other marginals one continues in this fashion with further steps of belief propagation. As a final remark, note that (in the binary case) messages can equivalently be considered as vectors with two components, or, equivalently, as Bernoulli distributions. □

5.5 Message Passing in Compressed Sensing

Recall the spin glass setting for compressed sensing in Section 3.4. From the marginals $\nu_{i,\beta}(x_i|\underline{y})$ of the posterior distribution (3.43)

$$p_\beta(\underline{x}|\underline{y}) = \frac{1}{Z_\beta} \prod_{a=1}^{r} e^{-\frac{\beta}{2\sigma^2}(y_a - \underline{A}_a \cdot \underline{x})^2} \prod_{i=1}^{n} (p_0(x_i))^\beta, \tag{5.13}$$

we can compute the Gibbs average $\widehat{x}_{i,\beta}(\underline{y}) = \langle x_i \rangle_\beta$. To get the MMSE estimate (when the prior is known) we set $\beta = 1$; to get the LASSO estimate (when we only know that the prior is in the sparse class $\mathcal{F}_\kappa$) we take $p_0(x) \propto e^{-\frac{\lambda}{\sigma^2}|x|}$ and send $\beta \to +\infty$.

For compressive sensing marginalization involves integrals instead of discrete sums. Formally, the distributive law (5.5) is replaced by

$$\int dx\, a(x)b(x) + \int dx\, a(x)c(x) = \int dx\, a(x)(b(x) + c(x))$$

but otherwise the marginalization proceeds exactly in the same way as in the discrete case if we simply replace sums by integrals in the message-passing rules (note that in our applications all integrals remain finite).

To obtain $\nu_{i,\beta}(x_i|\underline{y})$, it is sufficient to marginalize the numerator in (5.13) and eventually normalize the resulting function of x_i. Similarly to coding, this numerator has a factorized form with two types of factors

$$f_i(x_i) = (p_0(x_i))^\beta \qquad \text{and} \qquad f_a(\underline{x}_{\partial a}) = e^{-\frac{1}{2\sigma^2}(y_a - \underline{A}_a \cdot \underline{x})^2}.$$

We already associated a factor graph to the measurement matrix A in Chapter 2. Here we go one step further. In the factor graph representation for the distribution (5.13) we add extra square nodes corresponding to the factors $(p_0(x_i))^\beta$ and attach them to variable nodes. The other square nodes already present in the representation of the measurement matrix are associated to the factors $f_a(\underline{x}_{\partial a})$. Let us discuss a concrete illustration.

EXAMPLE 15 (Factor graph for compressive sensing) Figure 5.7 shows a factor graph associated to (5.13). Edges are present if and only if $A_{ai} \neq 0$ (one may think of $A_{ai} \neq 0$ as the "strength" of an edge). This factor graph contains the graph representing A itself, and has also additional factor nodes which represent the prior for the signal. □

A few comments are in order. In this example we take a factor graph that is a tree for the purpose of illustration of the message passing rules below. However in compressive sensing the graph is far from being a tree; it typically is a complete graph. Indeed we typically assume that the entries of the measurement matrix are independent and identical Gaussians, so the matrix is dense. This is one important difference between the compressive sensing and coding models. In coding our analysis will rely heavily on the fact that the graph is sparse and that when we look at very large instances the graph will "locally" be a tree. At first

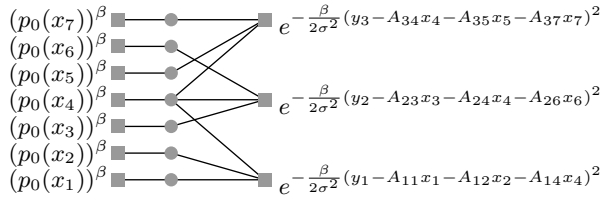


Figure 5.7 Factor graph for compressive sensing. The edges represent the non-zero elements of the measurement matrix. The signal has seven components and there are three measurements.

glance it therefore appears that message-passing techniques which explicitly rely on the graph being a tree are of no use in the compressive sensing context. But, as we will see, perhaps surprisingly, we are still able to analyze this situation. The key in this case is that despite the fact that we will not face a tree, the influence of each edge vanishes in the limit of large graphs. This relies heavily on the $1/n$ scaling of the variance of the matrix elements A_{ai}.

Let us now discuss belief propagation for the example.

EXAMPLE 16 (Message passing algorithm for compressive sensing) We give the first three steps of belief propagation for the tree in Figure 5.7. As remarked above, the messages are continuous distributions and instead of performing binary sums one has compute integrals. This is the main difference with the coding case. In the first step, the initial messages are sent from leaf nodes: $\widehat{\mu}_{k\to k}(x_k) = (p_0(x_k))^\beta$ for $k = 1, \ldots, 7$. At the second step six variables (namely the ones that participate in only one measurement) send messages to factor nodes: $\mu_{1\to 1}(x_1) = (p_0(x_1))^\beta$, $\mu_{2\to 1}(x_2) = (p_0(x_2))^\beta$, $\mu_{3\to 2}(x_3) = (p_0(x_3))^\beta$, $\mu_{5\to 3}(x_5) = (p_0(x_5))^\beta$, $\mu_{6\to 2}(x_6) = (p_0(x_6))^\beta$ $\mu_{7\to 1}(x_7) = (p_0(x_7))^\beta$. At the third step the three factor nodes send messages back to variable node 4. These are

$$\widehat{\mu}_{1\to 4}(x_4) = \int\int dx_1 dx_2\, (p_0(x_1))^\beta (p_0(x_2))^\beta e^{-\frac{\beta}{2\sigma^2}(y_1-A_{11}x_1-A_{12}x_2-A_{14}x_4)^2},$$

$$\widehat{\mu}_{2\to 4}(x_4) = \int\int dx_3 dx_6\, (p_0(x_3))^\beta (p_0(x_6))^\beta e^{-\frac{\beta}{2\sigma^2}(y_2-A_{23}x_3-A_{24}x_4-A_{26}x_6)^2},$$

$$\widehat{\mu}_{3\to 4}(x_4) = \int\int dx_5 dx_7\, (p_0(x_5))^\beta (p_0(x_7))^\beta e^{-\frac{\beta}{2\sigma^2}(y_3-A_{34}x_4-A_{35}x_5-A_{37}x_7)^2}.$$

Note that all integrals are certainly convergent as long as the prior $(p_0(x))^\beta$ is integrable. At this point we can compute the marginal $\mu_4(x_4)$. Indeed all messages incoming into variable node 4 are known, so

$$\mu_{4,\beta}(x_4) = (p_0(x_4))^\beta \widehat{\mu}_{1\to 4}(x_4)\widehat{\mu}_{2\to 4}(x_4)\widehat{\mu}_{3\to 4}(x_4)$$

To get the marginal $\nu_{4,\beta}(x_4|\underline{y})$ we normalize,

$$\nu_{4,\beta}(x_4 \mid \underline{y}) = \frac{\mu_{4,\beta}(x_4)}{\int dx_4\, \mu_{4,\beta}(x_4)}.$$

Finally, the computation of other marginals requires further steps of belief propagation. □

This time, contrary to the coding example where binary sums could easily be computed, in general the integrals cannot be performed analytically but have to be evaluated numerically. One exception where a complete analytical calculation is easy, is the case where the prior is Gaussian which leads to messages that are Gaussians throughout the whole belief propagation algorithm. A popular prior in the context of compressed sensing which also leads to explicit although rather complicated formulas, is a mixture of Bernoulli and Gaussian distributions. Note however, that the Laplacian prior $\propto e^{-\frac{\lambda}{\sigma^2}|x_k|}$ does not lead to completely analytically tractable integrals because of the absolute value. We will see in Chapter 8 that even when exact evaluation of the integrals is not possible, due to the proper scaling of the dense measurement matrix we can make useful approximations that lead to a tractable set of message passing equations.

LASSO estimate and min-sum rules

We remarked in 3.4 that the LASSO estimate can be obtained by taking the prior $p_0(x_i) \propto e^{-\frac{\lambda}{\sigma^2}|x_i|}$ and letting $\beta \to +\infty$. Taking the $\beta \to +\infty$ limit of the message passing rules developed here leads to the so-called *min-sum* rules. To obtain a well defined limit for the message passing rules it is convenient to define

$$\widehat{E}_{a\to i}(x_i) = -\lim_{\beta\to+\infty} \frac{1}{\beta} \ln \widehat{\mu}_{a\to i}(x_i), \quad \text{and} \quad E_{i\to a}(x_i) = -\lim_{\beta\to+\infty} \frac{1}{\beta} \ln \mu_{i\to a}(x_i)\,.$$

and the *marginal energy costs*

$$E_i(x_i) = -\lim_{\beta\to+\infty} \frac{1}{\beta} \ln \mu_i(x_i)\,.$$

The meaning of this "marginal" becomes intuitive once we notice

$$E_i(x_i) = \min_{\sim x_i}\{\frac{1}{2\sigma^2}\|\underline{y} - A\underline{x}\|_2^2 + \lambda\|x\|_1\}$$

so that $E_i(x_i) - \min_{x_i} E_i(x_i)$ is the cost incurred when variable i is set to the value x_i.

It is instructive to work this out in detail for the current example. The initial messages from leaf square nodes to variables are $\widehat{E}_{k\to k}(x_k) = \frac{\lambda}{\sigma^2}|x_k|$ for $k = 1, \ldots, 7$. At the second step the six variables $k = 1, 2, 3, 5, 7$ participating in a single measurement send messages to factor nodes: $E_{1\to1}(x_1) = \frac{\lambda}{\sigma^2}|x_1|$, $E_{2\to1}(x_2) = \frac{\lambda}{\sigma^2}|x_2|$, $E_{3\to2}(x_3) = \frac{\lambda}{\sigma^2}|x_3|$, $E_{5\to3}(x_5) = \frac{\lambda}{\sigma^2}|x_5|$, $E_{6\to2}(x_6) = \frac{\lambda}{\sigma^2}|x_6|$, $E_{7\to3}(x_7) = \frac{\lambda}{\sigma^2}|x_7|$. At the third step the three factor nodes send messages to variable node 4. These are deduced from the finite β messages by applying the

Laplace method to the integrals,

$$\widehat{E}_{1\to 4}(x_4) = \min_{x_1,x_2}\big\{E_{1\to 1}(x_1) + E_{2\to 1}(x_2) + \frac{1}{2\sigma^2}(y_1 - A_{11}x_1 - A_{12}x_2 - A_{14}x_4)^2\big\}$$
$$\widehat{E}_{2\to 4}(x_4) = \min_{x_3,x_6}\big\{E_{3\to 2}(x_3) + E_{6\to 2}(x_6) + \frac{1}{2\sigma^2}(y_2 - A_{22}x_2 - A_{24}x_4 - A_{26}x_6)^2\big\},$$
$$\widehat{E}_{3\to 4}(x_4) = \min_{x_5,x_7}\big\{E_{5\to 3}(x_5) + E_{7\to 3}(x_7) + \frac{1}{2\sigma^2}(y_3 - A_{34}x_4 - A_{35}x_5 - A_{37}x_7)^2\big\}.$$

The marginal energy cost of node 4 is

$$E_4(x_4) = \widehat{E}_{4\to 4}(x_4) + \widehat{E}_{1\to 4}(x_4) + \widehat{E}_{2\to 4}(x_4) + \widehat{E}_{3\to 4}(x_4)$$

and the LASSO estimate for variable node 4 is simply $\widehat{x}_4 = \operatorname{argmin} E_4(x_4)$. These relations constitute the min-sum algorithm.

There is also an alternative route to derive these min-sum relations. The belief propagation (or sum-product) equations were derived from the distributive law once we applied it to a factor graph which is a tree. This led to the marginalization of a function. But instead of using the operations of summing and multiplying (leading to the sum-product algorithm) we can use as basic operations the minimization and summing. The corresponding distributive law for this case reads

$$\min(a+b, a+c) = a + \min(b, c). \tag{5.14}$$

We can now formaly proceed just as in the Section 5.2. A quick way to develop the formalism is to use the correspondence $(+, \times) \to (\min, +)$ which transforms $ab + ac = a(b + c)$ to $\min(a + b, a + c) = a + \min(b, c)$. The derivation of the min-sum message passing rules from the distributive law is left as an exercise.

5.6 Message passing in satisfiability

We illustrate two applications of message passing for satisfiability. In the first one we count solutions of a K-SAT formula and in the second we discuss the determination of minimum energy assignments.

Counting solutions through message passing

Recall in the satisfiability problem we introduced in Section 3.6 the number of solutions of a K-SAT formula,

$$\mathcal{N}_0 = \sum_{\underline{s}} \prod_{a=1}^{m} (1 - \prod_{j\in\partial a} \frac{1}{2}(1 + s_j J_{aj})). \tag{5.15}$$

We illustrate here how one could attempt to compute $\mathcal{N}_0$ by message passing methods. Suppose we can count the number of solutions having a fixed value

$s_i = \pm 1$ for the i-th variable, namely

$$\mathcal{N}_i(s_i) = \sum_{\sim s_i} \prod_{a=1}^{m} (1 - \prod_{j \in \partial a} \frac{1}{2}(1 + s_j J_{aj})). \tag{5.16}$$

where the sum carries over all variables except s_i. The total number of solutions is then obtained as $\mathcal{N}_0 = \mathcal{N}_i(+1) + \mathcal{N}_i(-1)$. The task of computing (5.16) is nothing else than our marginalization problem. The factor graph associated to (5.15) has only one type of factor

$$(1 - \prod_{j \in \partial a} \frac{1}{2}(1 + s_j J_{aj}))$$

associated to the square nodes. Again, message passing provides an exact solution on a tree-graph. When the graph is not a tree it forms the basis of a "solution finding" message passing algorithm, called Belief Propagation Guided Decimation (BPGD), which we will study in Chapter 9.3. Let us for now just illustrate how the marginalization proceeds on our simple tree graph example.

EXAMPLE 17 (Counting solutions in 3-SAT) Consider the 3-SAT formula shown on Fig. 5.8. Here we keep the signs $J_{ai} = \pm 1$ associated to the edges open in order to see more clearly the structure of the messages (so we have a set of $2^9 = 512$ formulas here). The factors associated to each square are the indicator functions of the clause. For example clause number 1 is *not* satisfied by the assignment $s_1 = J_{11}$, $s_2 = J_{12}$, $s_4 = J_{14}$ and is satisfied by the 7 other assignments. Note that contrary to coding and compressed sensing there is no "prior," so no degree-one square nodes with factors attached to variable nodes. Here

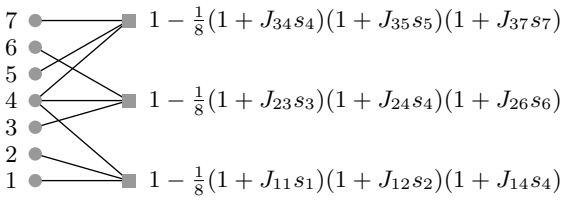


Figure 5.8 Factor graph for the K-SAT counting problem. The graph represents the formula and the factors associated to the square nodes are the indicator functions of each constraint written in spin language.

message passing starts at leaf nodes, namely the variable nodes $i = 1, 2, 3, 5, 6, 7$ which send the trivial initial messages $\mu_{i\to 1}(s_i) = \mu_{i\to 2}(s_i) = \mu_{i\to 3}(s_i) = 1$. In the second step all clauses send one outgoing message towards variable node 4

by taking into account their factor and two incomimg messages. In detail,

$$\widehat{\mu}_{1\to 4}(s_4) = \sum_{s_1,s_2} \big(1 - \frac{1}{8}(1+J_{11}s_1)(1+J_{12}s_2)(1+J_{14}s_4)\big) \times 1 \times 1,$$

$$\widehat{\mu}_{2\to 4}(s_4) = \sum_{s_3,s_6} \big(1 - \frac{1}{8}(1+J_{23}s_3)(1+J_{24}s_4)(1+J_{26}s_6)\big) \times 1 \times 1,$$

$$\widehat{\mu}_{3\to 4}(s_4) = \sum_{s_5,s_7} \big(1 - \frac{1}{8}(1+J_{34}s_4)(1+J_{35}s_5)(1+J_{37}s_7)\big) \times 1 \times 1$$

The binary sums are easily performed and yield $\widehat{\mu}_{a\to 4}(s_4) = 4 - \frac{1}{2}(1+J_{a4}s_4)$ for $a = 1, 2, 3$. In the next step we can compute the "marginal" for variable node 4 from the three incoming messages,

$$\mathcal{N}_4(s_4) = \mu_4(s_4) = (4 - \frac{1}{2}(1+J_{14}s_4))(4 - \frac{1}{2}(1+J_{24}s_4))(4 - \frac{1}{2}(1+J_{34}s_4)) \tag{5.17}$$

For example if the formula has $J_{14} = 1$, $J_{24} = 1$ and $J_{34} = -1$ the number of solutions with $s_4 = +1$ equals $\mathcal{N}_4(1) = 3 \times 3 \times 4 = 36$ and the number of solutions with $s_4 = -1$ equals $\mathcal{N}_4(-1) = 4 \times 4 \times 3 = 48$. The total number of solutions is $\mathcal{N}_0 = 36 + 48 = 84$. Note that we obtained this result without going through the remaining marginalization steps. This calculation also teaches us something about the uniform distribution over solutions. Indeed if we sample uniformly among solutions the probabilities that a solution has $s_4 = \pm 1$ are $\mathcal{N}_4(\pm 1)/\mathcal{N}_0 = 3/7$ and $4/7$. To calculate all such probabilities one has to go through the other marginalization steps. We obtain again these probabilities from a different point of view in the next paragraph. □

Message passing at positive and zero temperatures

Recall the Gibbs distribution in the finite temperature formulation of K-SAT

$$p(\underline{s}) = \frac{1}{Z} \sum_{\underline{s}} \prod_{a=1}^{m} \exp\big\{-\beta \prod_{i\in\partial a} \frac{1}{2}(1 + s_i J_{ai})\big\}. \tag{5.18}$$

Again we associate a factor graph to this distribution with one type of factor attached to the clauses, namely

$$f_a(\underline{s}_{\partial a}) = \exp\big\{-\beta \prod_{i\in\partial a} \frac{1}{2}(1 + s_i J_{ai})\big\}.$$

EXAMPLE 18 (Belief propagation at positive temperature for 3-SAT) Consider again the 3-SAT formula shown on Fig. 5.8. The factors associated to the square nodes are now β-dependent weights entering in (5.18). Message passing originates at leaf nodes $i = 1, 2, 3, 5, 6, 7$ which send the trivial initial messages $\mu_{i\to 1}(s_i) = \mu_{i\to 2}(s_i) = \mu_{i\to 3}(s_i) = 1$. In the second step all clauses send their message to

variable node 4,

$$\widehat{\mu}_{1\to 4}(s_4) = \sum_{s_1,s_2} \exp\Big\{-\frac{\beta}{8}(1+J_{11}s_1)(1+J_{12}s_2)(1+J_{14}s_4)\Big\} \times 1 \times 1,$$

$$\widehat{\mu}_{2\to 4}(s_4) = \sum_{s_3,s_6} \exp\Big\{-\frac{\beta}{8}(1+J_{23}s_3)(1+J_{24}s_4)(1+J_{26}s_6)\Big\} \times 1 \times 1,$$

$$\widehat{\mu}_{3\to 4}(s_4) = \sum_{s_5,s_7} \exp\Big\{-\frac{\beta}{8}(1+J_{34}s_4)(1+J_{35}s_5)(1+J_{37}s_7)\Big\} \times 1 \times 1$$

Using $e^{-\beta x} = 1 + (e^{-\beta} - 1)x$ for $x \in \{0, 1\}$ we can easily calculate the binary sums. For example

$$\begin{aligned}\widehat{\mu}_{1\to 4}(s_4) &= \sum_{s_1,s_2} \left(1 + \frac{1}{8}(e^{-\beta}-1)(1+J_{11}s_1)(1+J_{12}s_2)(1+J_{14}s_4)\right) \\ &= 4 + \frac{1}{2}(e^{-\beta}-1)(1+J_{14}s_4)\,.\end{aligned}$$

At this step we can already calculate the "marginal" $\mu_{4,\beta}(s_4)$ by multiplying all messages incoming into variable node 4

$$\begin{aligned}\mu_{4,\beta}(s_4) =& (4 + \frac{1}{2}(e^{-\beta}-1)(1+J_{14}s_4))(4 + \frac{1}{2}(e^{-\beta}-1)(1+J_{24}s_4)) \\ &\times (4 + \frac{1}{2}(e^{-\beta}-1)(1+J_{34}s_4))\end{aligned}$$

and the true marginal is obtained as usual by normalization

$$\nu_{4,\beta}(s_4) = \frac{\mu_{4,\beta}(s_4)}{\mu_{4,\beta}(1) + \mu_{4,\beta}(-1)}\,.$$

For the remaining marginals one has to perform extra message passing steps. □

Given a formula and given that solutions exist for this formula, when we take $\beta \to +\infty$ the Gibbs distribution tends to the uniform distribution over solutions. Therefore in the limit we have

$$\lim_{\beta\to+\infty} \nu_{i,\beta}(s_i) = \frac{\mathcal{N}_i(s_i)}{\mathcal{N}_0}\,. \tag{5.19}$$

This is easily checked explicitly in the last example above: with $J_{14} = 1$, $J_{24} = 1$, $J_{34} = -1$ we find $\lim_{\beta\to+\infty} \nu_{4,\beta}(1) = 3/7$ and $\lim_{\beta\to+\infty} \nu_{4,\beta}(-1) = 4/7$ which agrees with the computation in the counting example 17.

We now turn to the zero temperature case in more detail. Suppose we want to determine the assigments $\underline{s}$ that minimize the K-SAT Hamiltonian $\mathcal{H}(\underline{s})$ (3.53). When the graph associated to the formula is a tree message passing methods yield an exact solution; while in the non-tree case they form the basis of algorithms for finding solutions that we study in Chapters 9 and **??**. As for the LASSO estimator, we can take two alternative routes. We can directly set up the min-sum message passing rules by a proper use of the distributive law (5.14), or we can look at the $\beta \to +\infty$ limit of the belief propagation relations. The second

method is more convenient for us here, since we have already developed all the finite β formalism. This is illustrated with our running example.

EXAMPLE 19 (Zero temperature limit: min-sum for 3-SAT) Consider the same 3-SAT formula as in Fig. 5.8. The correct limiting behavior of messages is captured by introducing the energy costs

$$\widehat{E}_{a\to i}(s_i) = -\lim_{\beta\to+\infty}\frac{1}{\beta}\ln\widehat{\mu}_{a\to i}(s_i), \quad \text{and} \quad E_{i\to a}(s_i) = -\lim_{\beta\to+\infty}\frac{1}{\beta}\ln\mu_{i\to a}(s_i).$$

and

$$E_i(s_i) = -\lim_{\beta\to+\infty}\frac{1}{\beta}\ln\mu_i(s_i) = \min_{s_i}\mathcal{H}(\underline{s})$$

where $\mathcal{H}(\underline{s})$ is the Hamiltonian of K-SAT (3.53).

The initial messages from leaf nodes $i = 1, 2, 3, 5, 6, 7$ are $E_{i\to 1}(s_i) = E_{i\to 2}(s_i) = E_{i\to 3}(s_i) = 0$. Next, all clauses send a message to variable node 4,

$$\widehat{E}_{1\to 4}(s_4) = \min_{s_1,s_2}\Big\{\frac{1}{8}(1+J_{11}s_1)(1+J_{12}s_2)(1+J_{14}s_4) + E_{1\to 1}(s_1) + E_{2\to 1}(s_2)\Big\},$$
$$\widehat{E}_{2\to 4}(s_4) = \min_{s_3,s_6}\Big\{\frac{1}{8}(1+J_{23}s_3)(1+J_{24}s_4)(1+J_{26}s_6) + E_{3\to 2}(s_3) + E_{6\to 2}(s_6)\Big\},$$
$$\widehat{E}_{3\to 4}(s_4) = \min_{s_3,s_6}\Big\{\frac{1}{8}(1+J_{34}s_4)(1+J_{35}s_5)(1+J_{37}s_7) + E_{5\to 3}(s_5) + E_{7\to 3}(s_7)\Big\}.$$

The minima are easily calculated directly from these expressions. For example testing all four possibilities $(s_1, s_2) = (\pm J_{11}, \pm J_{12})$ yields $\widehat{E}_{1\to 4}(s_4) = 0$. Similarly we have $\widehat{E}_{2\to 4}(s_4) = \widehat{E}_{3\to 4}(s_4) = 0$. The resulting *marginal energy cost* for variable node 4 vanishes for both values of $s_4 = \pm 1$, namely

$$E_4(s_4) = \widehat{E}_{1\to 4}(s_4) + \widehat{E}_{2\to 4}(s_4) + \widehat{E}_{3\to 4}(s_4) = 0 \tag{5.20}$$

Since $E_4(s_4) = \min_{\sim s_4}\mathcal{H}(\underline{s})$ we deduce that there exist zero energy assignments that satisfy the formula with both values $s_4 = \pm 1$. In the present example this is true for all 512 possible formulas corresponding to the choices of edge signs. □

5.7 Notes

The belief propagation algorithm appeared independently and in different guises in various communities and contexts, and it is difficult to pinpoint its origin. The reader must have already sensed that the solution of the Ising model on a tree presented in Chapter 4 already contains the germ of the algorithm. In this context the belief propagation equations were clearly written down in (Morita 1979). The point of view was not algorithmic but rather that Bethe's improved mean field theory (Bethe 1935) becomes exact on trees. In the context of Low-Density Parity-Check codes message passing was used already by (Gallager 1962). Belief propagation was also invented as an algorithm for inference (Pearl 1982, Kim

& Pearl 1983). The generality of the message passing idea is nowadays well recognized in many communities overlapping computer science, digital communication, probability theory and statistical physics (e.g Pearl 1988, Frey 1998, Jordan 1999, Yedidia, Weiss & Freeman 2003, MacKay 2003, Loeliger 2004, Mézard & Montanari 2009, Richardson & Urbanke 2007).

The approach taken in this chapter is largely due to (Wiberg, Loeliger & Kötter 1995) and (Kschischang, Frey & Loeliger 2001). These works established the convenience of the factor graph representation and clarified the role and power of the distributive law. Our presentation in sections 5.1-5.3 closely follows (Richardson & Urbanke 2007). There are other convenient and beautiful graphical approaches that we have not discussed (e.g Shafer & Shenoy 1990, Aji & McEliece 2000, Forney 2001, Mao & Kschischang 2005).

Problems

5.1 Factor graph representations. Consider the Gibbs distributions for the Curie-Weiss and tree-Ising models, as well as Ising on a k-regular random graph and canonical Ising model on $\mathbf{Z}^d$ (introduced in Chapters 4 and 2). For each model identify the bipartite factor graphs representing each distribution. In particular give the expressions of the factors associated to the factor nodes. Same question for the p-spin models of exercices 4.1 and 4.6.

5.2 Belief propagation equations for the Ising model on a tree. Consider the factor graph representation of the tree-Ising model of Section 4.6, with a root node o, coordination number k, L levels, and uniform magnetic field and coupling constant. We want to use the belief propagation formalism to recover equations (4.39) and (4.41) for the magnetization of the root node o (for $k = 2$ this gives a message passing solution of the one-dimensional Ising model).

First consider the factor graph representation and write directly the belief propagation equations. Parametrize the messages as

$$\mu_{i\to a}(s_i) \propto e^{\beta u_{i\to a} s_i}, \qquad \widehat{\mu}_{a\to i}(s_i) \propto e^{\beta \widehat{u}_{a\to i} s_i}$$

to recover (4.39) and (4.41). It helps to note that the messages flowing out of leaf factor nodes are $\propto e^{\beta h s_i}$. Same question for the p-spin model of exercise 4.6.

5.3 Curie-Weiss equation from belief propagation. Consider the factor graph representation for the Gibbs distribution of the Curie-Weiss model and write down the belief propagation equations. Consider a flooding schedule (see Section 6.2 for a detailed definition) where at each round $t \geq 0$ all variable nodes send their messages to the factor and each factor sends back its messages to the variable node. Show that in the limit $n \to +\infty$ one recovers an iterative form of the Curie-Weiss equation, namely $m_{t+1} = \tanh(\beta J m_t + \beta h)$ with initialisation $m_0 = \tanh h$. A similar parametrization to the one of the previous exercise is useful. Same question for the p-spin model of exercise 4.1.

5.4 Gaussian belief propagation. Write down the sum-product equations for the marginalization of Gaussian probability distributions of the form

Give a few hints or teh final equations here

$$p(\underline{x})\frac{1}{Z}e^{-\frac{1}{2}\underline{x}^T J\underline{x}+\underline{h}^T\cdot\underline{x}}$$

where $\underline{h} \in \mathbb{R}^n$ and J is an $n \times n$ positive definite matrix (here we assume that the factor graph of J is a tree so that the sum-product rules are exact). Perform explicitly all Gaussian integrals.

5.5 Min-Sum Message Passing rules. The main property we used to derive the belief propagation (or sum-product) equations is the *distributive law* for the two operations $+$ and $\times$ (on some field). Consider the following generalization. Take the "commutative semiring" of extended real numbers (i.e., $\bar{\mathbb{R}}$ including ∞) with the two operations $(\min, +)$ replacing the usual operations $(+, \times)$. Show that: (i) both operations are commutative; (ii) the identity element under min is ∞ and the identity element under $+$ is 0; (iii) the distributive law holds, $\min(a + b, a + c) = a + \min(b, c)$.

If we formally exchange in our original marginalization $+$ with min and $\times$ with $+$, then what corresponds to the marginalization of a function? What are the message passing rules and what is the initialization?

5.6 Min-sum rules for least square regression with an ℓ_2 penalty. Consider the following regularized least square minimization problem

$$\min_{\underline{x}}\{\frac{1}{2}\|\underline{y} - A\underline{x}\|_2^2 + \alpha\|x\|_2^2\}\,.$$

where $\alpha > 0$ (see problem 1.4). This kind of ℓ_2 penalty is often used for ill-posed problems (e.g., if $\underline{y} = A\underline{x}$ is an undertermined system of equations) and is often called a Tikhonov regularization, or also *ridge regression* in statistics.

Write down the min-sum rules for this minimization problem when the factor graph (corresponding to matrix A) is a tree. You can proceed by formulating the finite temperature problem first and then take a zero temperature limit, or you can directly use the distributive law for the operations $(\min, +)$.

Define a "marginal energy cost",

$$E_i(x_i) = \min_{\sim x_i}\left\{\frac{1}{2}\|\underline{y} - A\underline{x}\|_2^2 + \lambda\|x\|_2^2\right\},$$

where $\min_{\sim x_i}$ denotes minimization with respect to all variables, except x_i which is held fixed. Describe how this marginal energy cost is computed from the min-sum messages and how the regularized least square estimate is deduced.

6 Coding: Belief Propagation and Density Evolution

Message passing methods have been very successful in providing efficient and analyzable algorithms for the coding problem. In this chapter we provide an introduction to this analysis. From now on we adopt the specific terminology of coding and to refer to "Belief Propagation" (BP) algorithms and leave the term "message-passing" as a generic term.

In Chapter 5 we learned how to marginalize a Gibbs distribution whose factor graph is a tree, by employing BP rules. We saw that on trees BP starts at the leaf nodes and that a node which has received messages from all its children processes the messages and forwards the result to its parent node. On a tree this BP algorithm is equivalent to MAP decoding since we are computing without any approximation the marginals of the posterior distribution.

If the graph is not a tree then we can still use BP, but we need to define a *schedule* which determines when to update what messages. It is not clear how well such an algorithm will perform. It is the aim of the present chapter to clarify these issues. We will carry out the analysis in detail for the binary erasure channel (BEC) and explain the main ideas involved in for the general case of binary-input memoryless symmetric channels. The BEC channel has the advantage that its analysis can be done by pen and paper. The general case is conceptually not much harder, but there are a significant number of mathematical tools one has to introduce, which make the analysis more involved.

6.1 Message-Passing Rules for Bit-wise MAP Decoding

We illustrated the message passing rules for coding on a small coding example in Section 5.4. Recall that the Gibbs distribution has two type of factors: $e^{h_i s_i}$ and $\frac{1}{2}(1+\prod_{j\in\partial a} s_j)$. The first kind of factor is associated to factor nodes $\widehat{i}$ of degree one (representing channel output observations) attached to variable nodes i and generates a message $\mu_{\widehat{i}\to i}(s_i) = e^{h_i s_i}$. The other relevant messages flow from the parity checks to variable nodes $\widehat{\mu}_{a\to i}(s_i)$ and from variable nodes to parity checks $\mu_{i\to a}(s_i)$. Thus for coding the general BP equations (5.10) read

$$\begin{cases} \mu_{i\to a}(s_i) & = e^{h_i s_i} \prod_{b\in\partial i\smallsetminus a} \widehat{\mu}_{b\to i}(s_i)\,, \\ \widehat{\mu}_{a\to i}(s_i) & = \sum_{\sim s_i} \frac{1}{2}(1+\prod_{j\in\partial a} s_j) \prod_{j\in\partial a\smallsetminus i} \mu_{j\to a}(s_j)\,. \end{cases} \tag{6.1}$$

In the binary case of interest here these equations can be simplified by adopting a convenient parametrization of the messages. Indeed we already remarked at the end of Section 5.3 that their normalizations cancel out in the final computation of "marginals". So all that should matter are the log-likelihood ratios[1]

$$l_{i\to a} = \frac{1}{2}\ln\left\{\frac{\mu_{i\to a}(+1)}{\mu_{i\to a}(-1)}\right\}, \qquad \widehat{l}_{a\to i} = \frac{1}{2}\ln\left\{\frac{\widehat{\mu}_{a\to i}(+1)}{\widehat{\mu}_{a\to i}(-1)}\right\} \tag{6.2}$$

which do not involve the normalization.

To see what form the first BP equation in (6.1) takes with this parametrization, we write it for each value $s_i = \pm 1$, and take the ratio

$$\frac{\mu_{i\to a}\,(+1)}{\mu_{i\to a}\,(-1)} = e^{2h_i}\prod_{b\in\partial i\setminus a}\frac{\widehat{\mu}_{b\to i}\,(+1)}{\widehat{\mu}_{b\to i}\,(-1)}\,.$$

The logarithm then yields the *variable node rule*

$$l_{i\to a} = h_i + \sum_{b\in\partial i\setminus a}\widehat{l}_{b\to i}\,. \tag{6.3}$$

The reduction of the second BP equation in (6.1) to a form involving only the log-likelihood ratios (6.2) involves a little more algebra. First we write (6.1) for each spin value $s_i = \pm 1$ and consider the ratio,

$$\frac{\widehat{\mu}_{a\to i}\,(+1)}{\widehat{\mu}_{a\to i}\,(-1)} = \frac{\sum_{\sim s_i}(1+\prod_{j\in\partial a\setminus i}s_j)\prod_{j\in\partial a\setminus i}\mu_{j\to a}\,(s_j)}{\sum_{\sim s_i}(1-\prod_{j\in\partial a\setminus i}s_j)\prod_{j\in\partial a\setminus i}\mu_{j\to a}\,(s_j)}\,.$$

Next, we divide the numerator and denominator by $\prod_{j\in\partial a\setminus i}\mu_{j\to a}\,(-1)$ and use the identity

$$\frac{\mu_{j\to a}\,(s_j)}{\mu_{j\to a}\,(-1)} = e^{l_{j\to a}(s_j+1)} = (1+s_j\tanh l_{j\to a})e^{l_{j\to a}}\cosh l_{j\to a}$$

to obtain

$$\frac{\widehat{\mu}_{a\to i}\,(+1)}{\widehat{\mu}_{a\to i}\,(-1)} = \frac{\sum_{\sim s_i}(1+\prod_{j\in\partial a\setminus i}s_j)\prod_{j\in\partial a\setminus i}(1+s_j\tanh l_{j\to a})}{\sum_{\sim s_i}(1-\prod_{j\in\partial a\setminus i}s_j)\prod_{j\in\partial a\setminus i}(1+s_j\tanh l_{j\to a})}\,. \tag{6.4}$$

The summations in the numerator and denominator can be performed explicitly. We first expand the products in each $\sum_{\sim s_i}$ into a sum of monomials of the spin variables

$$\begin{aligned}
&(1\pm\prod_{j\in\partial a\setminus i}s_j)\prod_{j\in\partial a\setminus i}(1+s_j\tanh l_{j\to a})\\
&\quad=(1\pm\prod_{j\in\partial a\setminus i}s_j)\sum_{J\subset\partial a\setminus i}\prod_{j\in J}s_j\prod_{j\in J}\tanh l_{j\to a}\\
&\quad=\sum_{J\subset\partial a\setminus i}\prod_{j\in J}s_j\prod_{j\in J}\tanh l_{j\to a}\pm\sum_{J\subset\partial a\setminus i}\prod_{j\in J^c}s_j\prod_{j\in J}\tanh l_{j\to a}
\end{aligned}$$

[1] In the coding theory literature it is usual to define the log-likelihood ratios *without* the prefactor 1/2. With the definition adopted here these are exactly the same objects than the "magnetic fields" of statistical mechanics. We refer interchangeably to "log-likelihood ratios" or "magnetic fields" depending what we wish to emphasize.

where the sums over $J \subset \partial a \setminus i$ run over all subsets of $\partial a \setminus i$ including $J \neq \emptyset$ and $J = \partial a \setminus i$. In the last line the only monomials that survive correspond to the subsets $J = \emptyset$ (resp. $J^c = \emptyset$) for the first sum (resp. second sum) and we simply get

$$1 \pm \prod_{j \in \partial a \setminus i} \tanh l_{j \to a}$$

Therefore the ratio (6.4) reduces to the simple form

$$\frac{\widehat{\mu}_{a \to i}(+1)}{\widehat{\mu}_{a \to i}(-1)} = \frac{1 + \prod_{j \in \partial a \setminus i} \tanh l_{j \to a}}{1 - \prod_{j \in \partial a \setminus i} \tanh l_{j \to a}}.$$

Finally taking the logarithm and using $\frac{1}{2} \ln \frac{1+x}{1-x} = \operatorname{atanh} x$ we arrive at the *check node rule*

$$\widehat{l}_{a \to i} = \operatorname{atanh}\Big\{ \prod_{j \in \partial a \setminus i} \tanh l_{j \to a} \Big\}. \tag{6.5}$$

Let us now look at the "marginals" computed from the BP equations. We call them *BP-marginals* and denote them by $\nu_i^{\mathrm{BP}}(s_i)$ to make a clear distinction with the true marginals $\nu_i(s_i)$ of the Gibbs distribution (as repeatedly pointed the two types of marginals are equal on a tree). Adapting (5.11) to the present setting,

$$\nu_i^{\mathrm{BP}}(s_i) = \frac{e^{h_i s_i} \prod_{a \in i} \widehat{\mu}_{a \to i}(s_i)}{e^{h_i} \prod_{a \in i} \widehat{\mu}_{a \to i}(+1) + e^{-h_i} \prod_{a \in i} \widehat{\mu}_{a \to i}(-1)}\,.$$

In order to express the BP marginals in terms of the log-likehood ratios we divide the numerator and denominator by $e^{h_i} \prod_{a \in i} \widehat{\mu}_{a \to i}(+1)$ and use (6.2) to deduce

$$\begin{aligned} \nu_i^{\mathrm{BP}}(s_i) &= \frac{e^{(h_i + \sum_{a \in \partial i} \widehat{l}_{a \to i})(s_i + 1)}}{1 + e^{2(h_i + \sum_{a \in \partial i} \widehat{l}_{a \to i})}} \\ &= 1 + s_i \tanh\big(h_i + \sum_{a \in \partial i} \widehat{l}_{a \to i}\big) \end{aligned}$$

From this BP-marginal one can compute the *BP-magnetization* of the i-th bit (to be distinguished from the true magnetization)

$$m_i^{\mathrm{BP}} = \sum_{s_i = \pm 1} s_i \nu_i^{\mathrm{BP}}(s_i) = \tanh\big(h_i + \sum_{a \in \partial i} \widehat{l}_{a \to i}\big) \tag{6.6}$$

The BP *estimate* for bit i is then found from[2]

$$\widehat{s}_i^{\mathrm{BP}} = \operatorname{sign}(m_i^{\mathrm{BP}})\,. \tag{6.7}$$

The average bit-wise probability of error associated to this estimate is discussed in Section 6.4.

[2] Recall the convention already used in Chapter 3: $\operatorname{sign}(x) = 0$ for $x = 0$, ± 1 for $x > 0, x < 0$.

There is an important statistical mechanical interpretation of (6.6). The BP-magnetization is the same as that of a system constituted by a *single spin* with Gibbs distribution (at $\beta = 1$)

$$\frac{e^{-l_i s_i}}{2\cosh l_i}, \qquad l_i = h_i + \sum_{a \in \partial i} \widehat{l}_{a \to i}$$

This is the distribution of a spin that sees a "local field" l_i equal to the "external magnetic field" h_i plus a "mean field" $\sum_{a \in \partial i} \widehat{l}_{a \to i}$. In the BP framework the mean filed is given by a sum over all "cavity fields" $l_{a \to i}$, $a \in \partial i$. From the point of view of traditional statistical mechanics the BP-magnetization is a "mean field approximation" of the true magnetization.

Summary of BP equations for coding

To summarize, in the case of transmission over a binary input memoryless channel the messages can be represented by a single real quantity. If we choose this quantity to be the log-likelihood ratio (6.2) then the processing rules at variable and check nodes take on a particularly simple form (see (6.3), (6.5))

$$\begin{cases} l_{i \to a} = h_i + \sum_{b \in \partial i \setminus a} \widehat{l}_{b \to i} \\ \widehat{l}_{a \to i} = \operatorname{atanh}\left\{ \prod_{j \in \partial a \setminus i} \tanh l_{j \to a} \right\} \end{cases} \tag{6.8}$$

The BP estimate of a bit is given by (see (6.6), (6.7))

$$\widehat{s}_i^{\mathrm{BP}} = \operatorname{sign}(\tanh(h_i + \sum_{a \in \partial i} \widehat{l}_{a \to i})) \tag{6.9}$$

For the special case of the BEC one can make further simplifications as dicussed in Section 6.3.

6.2 Scheduling on general factor graphs

If the factor graph is a tree, then message-passing starts from the leaf nodes and messages propagate through the graph until a message has been sent on each edge in both directions. However, cycle-free parity-check codes do not perform well. This is true even if we allow optimal decoding. Hence we have to use codes whose graphs have cycles.

Given a factor graph with cycles, the order in which messages are computed has to be defined explicitly and in principle different orders might result in different performance. We call such an order a *schedule*. A naive scheduling which is convenient for the analysis of belief propagation is the *flooding* or *parallel* schedule. In this schedule at each step every outgoing message is updated according to the incoming messages in the previous step.

In more details, every iteration consists of two steps. In the first step we

compute the outgoing messages along each edge at variable nodes and we forward them to the check node side. In the second step we then process the incoming messages at check nodes, and compute for every edge at check nodes the outgoing messsage and send it back to variable nodes. What about the initial condition? At the very beginning, none of the messages except the ones coming from the channel are defined. So in order to get started, we set all "internal" messages to be "neutral" messages, $\widehat{\mu}_{a\to i}(\pm 1) = 1/2$. If we represent messages as log-likelihood ratios, this means that we set all internal messages to $\widehat{l}_{a\to i} = 0$. One can check that for a tree this prescription reduces to the initial conditions dictated by the theory developped in Chapter 5.

Let us formalize the above discussion. Iterations are indexed by "time", a discrete integer $t \geq 1$. At iteration t in the first step we have messages $l^t_{i\to a}$ flowing (in parallel) from variable to check nodes and in the second step we have messages $l^t_{i\to a}$ flowing from check to variable nodes. They satisfy

$$\begin{cases} l^t_{i\to a} = h_i + \sum_{b\in\partial i\smallsetminus a} \widehat{l}^{t-1}_{b\to i} \\ \widehat{l}^t_{a\to i} = \operatorname{atanh}\big\{\prod_{j\in\partial a\setminus i} \tanh l^t_{j\to a}\big\} \end{cases} \tag{6.10}$$

The iterative process is initialized with $l^{(0)}_{i\to a} = \widehat{l}^{(0)}_{a\to i} = 0$. The total estimated likelihood ratio for bit i at time t is

$$l^t_i = h_i + \sum_{a\in\partial i} \widehat{l}^t_{a\to i} \tag{6.11}$$

and the BP estimate of bit i at time t is

$$\widehat{s}^{\mathrm{BP,t}}_i = \operatorname{sign}(\tanh\, l^t_i) = \operatorname{sign}(l^t_i) \tag{6.12}$$

6.3 Message Passing and Scheduling for the BEC

The BEC is a very special binary input memoryless channel. As depicted in Fig. 1.2, the transmitted bit is either correctly received at the channel output with probability $1-\epsilon$ or erased by the channel with probability ϵ and thus, nothing is received at the channel output.[3] The erased bits are denoted by E. For example, if $s_i = 1$ (resp. $s_i = -1$) is transmitted in the BEC, then the set of possible channel observations is $\{1, E\}$ (resp.$\{-1, E\}$). The log-likelihood ratios corresponding to the various channel observations are

$$h_i = \frac{1}{2}\ln\left\{\frac{p(y_i \mid s_i = 1)}{p(y \mid s_i = -1)}\right\} = \begin{cases} \frac{1}{2}\ln(\frac{1-\epsilon}{0}) = +\infty & y = 0, \\ \frac{1}{2}\ln(\frac{\epsilon}{\epsilon}) = 0, & y = E, \\ \frac{1}{2}\ln(\frac{0}{1-\epsilon}) = -\infty, & y = 1. \end{cases}$$

Now, since the initial condition for the internal messages is $\widehat{l}^0_{a\to i} = 0$, the iterations (6.10) imply that at later times $l^t_{i\to a}, \widehat{l}^t_{a\to i} \in \{0, \pm\infty\}$. This allows to further simplify the BP equations.

[3] The position of the erased bit is known and only its value is unknown.

According to the variable-node rule the outgoing message from a variable node is $+\infty$ (or $-\infty$) if at least one incoming message from one of its neighbors is $+\infty$ (or $-\infty$), otherwise it is equal to 0. Note that it is not possible that a variable node receives both $+\infty$ and $-\infty$ simultaneously. This is due to the fact that by assumption the transmitted word is a valid codeword and that the channel never introduced mistakes.

Since $\tanh l_{i\to a} \in \{\pm 1, 0\}$, we can use $\tanh l_{i\to a} = \operatorname{sign}(l_{i\to a})$ to simplify the updating rule of check nodes to the following equation,

$$\operatorname{sign}(\widehat{l}^t_{a\to i}) = \prod_{j\in\partial a\setminus i} \operatorname{sign}(l^t_{j\to a}). \tag{6.13}$$

This discussion shows that on the BEC, knowing the sign of all incoming messages is sufficient to compute outgoing messages, thus we can assume that the set of messages is $\{0, \pm 1\}$ instead of $\{\pm\infty, 0\}$. At check nodes the operation is then simple multiplication. At variable nodes, if at least one of the incoming edges is non-zero, then all non-zero incoming messages must in fact be the same and the outgoing message is this common value. Otherwise, when all incoming messages are 0, the outgoing message is also 0.

For the BEC, but only for the BEC, we can implement the parallel schedule in a more efficient manner. Some thought shows that the messages emitted along a particular edge can only jump once, namely from 0 to either the value $+1$ or -1. After the value has jumped it stays constant thereafter. Further, the message can only jump if at least one of the incoming messages jumped. Therefore, rather than recomputing every message along every edge in each iteration, we can just follow changes in the messages and see if they have consequences. As a consequence, we have to "touch" every edge only once and so the complexity of this algorithm scales linearly in the number of edges.

6.4 Two Basic Simplifications

To analyze the performance of the (l, r)-regular LDPC ensemble over a channel, we pick a code $\mathcal{C}$ uniformly at random from the ensemble of graphs and run the message passing algorithm. For a given code $\mathcal{C}$ and channel parameter ϵ, let $P_{\mathrm{BP,b}}(\mathcal{C}, \underline{s}^{\mathrm{in}}, \epsilon, t)$ denote the average bit-wise error probability of the BP decoder at iteration t, when the input codeword is $\underline{s}^{\mathrm{in}}$. Explicitely,

$$\begin{aligned} P_{\mathrm{BP,b}}(\mathcal{C}, \underline{s}^{\mathrm{in}}, \epsilon, t) &\equiv \frac{1}{n}\sum_{i=1}^{n} \mathbb{P}(\widehat{s}_i^{\mathrm{BP,t}} \neq s_i^{\mathrm{in}}) \\ &= \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2}(1 - \mathbb{E}_{\underline{h}|\underline{s}^{\mathrm{in}}}[s_i^{\mathrm{in}}\widehat{s}_i^{\mathrm{BP,t}}]) \end{aligned} \tag{6.14}$$

where $\mathbb{P}_{\underline{h}|\underline{s}^{\mathrm{in}}}$ and $\mathbb{E}_{\underline{h}|\underline{s}^{\mathrm{in}}}$ are the probability and expectation with respect to channel outputs conditional on the input word (see Chapter 3). We will study the

behavior of $\mathbb{P}_{\mathrm{BP,b}}(\mathcal{C}, \underline{s}^{\mathrm{in}}, \epsilon, t)$ in terms of ϵ and t as a measure of performance of the code $\mathcal{C}$.

For the BEC, we either can decode a bit correctly, or the bit is still erased at the end of the decoding process. Therefore, in this case we typically compute the *bit erasure probability*. If we want to convert this into an *error probability*, then we can imagine that for all erased bits we flip a coin uniformly at random. With probability one-half we will guess the bit correctly and with probability one-half we will make a mistake. Therefore, the bit erasure and the bit error probability are the same up to a factor of one-half. In our calculations it is always more convenient to compute the erasure probability for the BEC (this is simply (6.14) without the factor 1/2). But our language reflect the general case and so we will also talk about error probabilities.

Restriction To The All-One Codeword

In Chapter 3 we showed that the bit-wise MAP error probability is independent of the transmitted codeword as long as the channel is symmetric. Something similar holds for the BP decoder. Therefore we can analyze the error probability of the BP decoder assuming that $s_i^{\mathrm{in}} = 1$, $i = 1, \ldots, n$, was transmitted (the "all-zero" codeword $\underline{x}^{\mathrm{in}} = 0$). In formulas, we claim that (6.14) equals

$$P_{\mathrm{BP,b}}(\mathcal{C}, \epsilon, t) \equiv \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} (1 - \mathbb{E}_{\underline{h}|\underline{1}}[\widehat{s}_i^{\mathrm{BP,t}}]) \tag{6.15}$$

This is true in a more general setting than the present one. In general, for the statement to hold we need two kinds of symmetry to hold: *channel symmetry* (see Chapter 3, Section 3.2) and *decoder symmetry*. Decoder symmetry here means that at check nodes the magnitude of the outgoing message is only a function of the magnitude of the incoming messages, and that the sign of the outgoing message is the product of the signs of the incoming messages. At variable nodes, we require that if the signs of all the incoming messages are reversed then the outgoing message also just changes by a reversal of the sign. Equations (6.10) show this is obviously the case for the BP decoder. But often one implements simplified versions of this decoder for which the symmetry conditions also hold.

For the BEC and BP decoding it is particularly easy to see why (6.15) is true. If we go back to the message-passing rules for this case, we see that both at check nodes as well as at variable nodes we can determine if the outgoing message is an erasure or not by only looking how many of the incoming messages are erasures, and we do not need to know the values of the incoming messages. Therefore, the final erasure probability only depends on the erasure pattern created by the channel, and is independent of the transmitted codeword.

For general symmetric channels (6.15) is proved by using the two symmetry conditions stated above. The proof is the subject of an exercise.

Concentration

The second major simplification stems from the fact that, rather than analyzing individual codes, it suffices to assess the average performance of the code ensemble. When this is true the individual behavior of elements of an ensemble is with high probability close to the ensemble average. More precisely one can prove the following statement.

Concentration of bit-wise BP error probability: Let $\mathcal{C}$, chosen uniformly at random from the Gallager ensemble LDPC(d_v, d_c, n), be used for transmission over a binary-input memoryless symmetric channel. Then, for any given $\delta > 0$, there exists an $\alpha > 0$, $\alpha = \alpha(d_v, d_c, \delta)$, such that

$$\mathbb{P}\{|P_{\mathrm{BP,b}}(\mathcal{C}, \epsilon, t) - \mathbb{E}\left[P_{\mathrm{BP,b}}(\mathcal{C}, \epsilon, t)\right]| > \delta\} \leq \epsilon^{-\alpha n} \tag{6.16}$$

where here $\mathbb{P}$ and $\mathbb{E}$ refer to the code ensemble.

In words, all except an exponentially (in the blocklength) small fraction of codes behave within an arbitrarily small δ from the ensemble average. Therefore, assuming sufficiently large blocklengths, the ensemble average is a good indicator for the individual behavior and it seems a reasonable route to focus one's effort on the design and construction of ensembles whose average performance approaches the Shannon theoretic limit.

6.5 Concept of Computation Graph

Message passing takes place on the local neighborhood of a node. At each iteration, variable nodes send their beliefs $l_{i\to a}$ along their edges toward check nodes, and then check nodes compute the outgoing message $\hat{l}_{a\to i}$ for each of their edges according to the beliefs of incoming edges and send it back to the variable nodes. Afterwards, each variable node updates the outgoing messages along its edges according to beliefs returned back on its edges.

Therefore, after t iterations, the belief of a variable node depends on its initial belief h_i and the beliefs of all the nodes placed within (graph) distance $2t$ or less. The graph consisting of these nodes is called the computation graph of that variable node of height t. Figure 6.1 illustrates, the factor graph of a $(2, 4)$-regular LDPC code and the computation graphs of node 1 with height 1 and height 2. On this example the computation graph of height 1 is a tree because each labelled node appears only once. The computation graph of height 2 is not a tree because some nodes appear more than once. Nevertheless the computation graph is always conveniently "depicted" as a tree.

If a computation graph is a tree, then no node is used more than once in the graph. Therefore the incoming messages of each node are independent. But note that by increasing the number of iterations, the number of nodes in a computation graph grows exponentially and thus in at most $c \log n$ steps, where c is some suitable constant, some node will necessarily be reused. It is clear that

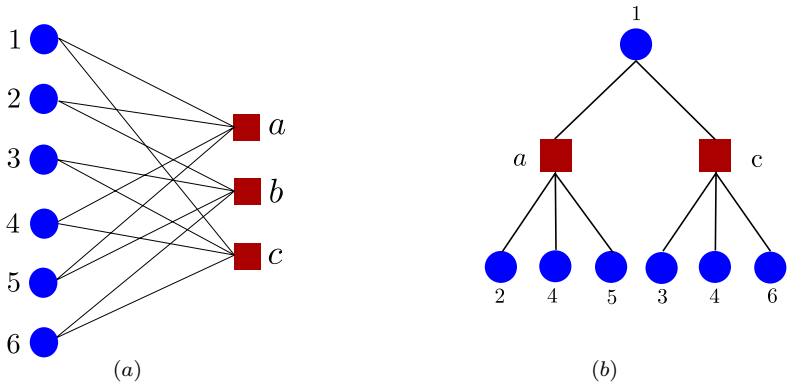


Figure 6.1 The factor graph of a $(2,4)$-regular LDPC code with 6 variable nodes, and the corresponding computation graphs of node 1 for the first iteration and second iterations. For one iteration the computation graph is a tree. It is not a tree for two iterations.

small computation graphs are more likely to be tree-like than large ones and that the chance of having a tree-like computation graph increases if we increase the blocklength.

Let us discuss this last point in more detail. Let T_t denote the computation graph of a variable node chosen uniformly at random from the set of variable nodes of height t in the (d_v, d_c)-regular LDPC ensemble. If the height t is kept fixed then

$$\lim_{n\to\infty} \mathbb{P}\left(T_t \text{ is a tree }\right) = 1. \tag{6.17}$$

We only give a sketch of the proof. We are given the randomly chosen variable node and we construct its computation graph of height t by growing out its "tree" one node at a time, breadth first. We use the principle of *deferred decisions*. This means that rather than first constructing a particular code, then checking if the corresponding computation graph is a tree and then averaging over all codes we perform the averaging over all codes at the same time as we grow the tree, in other words we *defer* the decision of how edges are connected until we look at a particular edge and reveal its endpoints. Note that a computation graph of a fixed height has at most a certain number of nodes and edges. At each step when we reveal how a particular edge is connected there are two possible events. The newly inspected edge is either connected to a node which is already contained in the computation graph. In this case we terminate the procedure since we know that the computation graph is not a tree. Or the edge is connected to a new node, maintaining the tree structure. Since not yet revealed edges are connected uniformly at random to any not yet filled slot, the probability of reconnecting to an already visited node vanishes like $1/n$, where n is the blocklength. By the union bound, and since we only perform a fixed number of steps of order, it

follows that the probability that the computation graph is indeed a tree behaves like $1 - O(1/n)$, which proves the claim.[4]

6.6 Density Evolution

We will now show how to compute the bit error probability under BP decoding. According to (6.15), given a code $\mathcal{C}$ from the ensemble, and a variable node i selected uniformly at random, we should compute the expectation of $\widehat{s}_i^{\mathrm{BP},t}$. A look at (6.12) shows that we should determine the probability distribution of l_i^t. The difficulty here is that this depends on previous iterations in which the messages are not independent. Fortunately the concentration result(6.16) and the local tree like property (6.17) allow to by-pass this problem, at least in the limit where n grows large and t is fixed (but arbitrarily large).

From the concentration (6.16) of the error probability it suffices to compute the average (over the code ensemble) error probability

$$P_{\mathrm{BP,b}}(d_v, d_c, \epsilon, t) \equiv \lim_{n \to +\infty} \mathbb{E}[P_{\mathrm{BP,b}}(\mathcal{C}, \epsilon, t)], \tag{6.18}$$

and since the computation graph T_t of a random vertex of fixed height t is a tree with probability $1 - O(1/n)$ we get

$$P_{\mathrm{BP,b}}(d_v, d_c, \epsilon, t) = \lim_{n \to +\infty} \mathbb{E}[P_{\mathrm{BP,b}}(\mathcal{C}, \epsilon, t) | T_t \text{ is a tree }]. \tag{6.19}$$

Our task is therefore reduced to the computation of the probability distribution of l_i^t on a *tree* T_t. This problem can be handled quite easily, at least in principle, because the incoming messages to each node of this tree are *independent*.

It is common to refer to the iterative equations governing the probability distributions of messages on the tree as the *Density Evolution* (DE) equations. For the BEC these are a simple set of algebraic (polynomial) equations and we first give their derivation in this simple case. For general channels these are integral equations, but as we will see their derivation is (conceptually at least) not much more difficult.

DE equations for the BEC

Consider a computation *tree* T_t with height t. We divide this computation graph into $t + 1$ levels, from 0 to t. We label the levels by $m = 0, \cdots, t$. Level $m = 0$ contains the leaf nodes, level $m = 1$ contains the parent check nodes *and* the grandparent variable nodes of the leaf nodes, etc (see Fig. 6.2). Recall that the messages can be reduced to the alphabet $\{0, \pm 1\}$ where 0 corresponds to an erasure and ± 1 to known a known value of the bit.

Every variable node at the m-th level is the root of a computation tree with height m. Consider the outgoing message $\in \{0, \pm 1\}$ emitted by a variable nodes

[4] A more detailed argument shows that the t dependence of $O(1/n)$ grows as $(d_v d_c)^t$.

towards its parent check node in the $m+1$-th level. It is equal to either an erasure message 0 with probability x_m or a known value ± 1 with probability $1-x_m$. Now consider level $m+1$. Each variable node is connected to d_v-1 check nodes and each check node is connected to d_c-1 variable nodes of m-th level. Consider the outgoing message $\in \{0, \pm 1\}$ emitted by a check node towards its parent variable node in the same level. We call y_m the probability that this message is an erasure 0.

The outgoing message of a check node is an erasure, if at least one of its incoming messages is also an erasure. Since on a tree the incoming messages are independent, the probability that a check node at level $m+1$ sends an erasure message to its parent variable node is

$$y_m = 1 - (1 - x_m)^{d_c - 1} \tag{6.20}$$

The outgoing message from a variable node of $m+1$-th level is an erasure if its initial message from the channel is erasure and all of its children (check nodes) at level $m+1$ also send erasure messages. Moreover on a tree the incoming messages are independent, hence

$$x_{m+1} = \epsilon y_m^{d_c - 1} \tag{6.21}$$

Equations (6.20) and (6.21) are the two *Density Evolution* (DE) equations for the BEC. Of course they can be merged into a single iteration

$$x_{m+1} = \epsilon(1 - (1 - x_m)^{d_c - 1})^{d_v - 1}. \tag{6.22}$$

By definition, the outgoing message at level 0 is an erasure with probability $x_0 = \epsilon$. Thus $x_0 = \epsilon$ serves as the initial condition for DE iterations.

The *erasure probability* of the root of T_t, which is connected to the d_v check nodes of level t, is equal to $\epsilon(1 - (1 - x_{t-1})^{d_c - 1})^{d_v}$. Since for each erased bit we flip a coin to decide if we decode it as a $+1$ or a -1, the average bit-error probability of the BP decoder finally is one-half times the erasure probability,

$$\mathbb{P}_{\mathrm{BP,b}}(d_v, d_c, \epsilon, t) = \frac{\epsilon}{2}(1 - (1 - x_{t-1})^{d_c - 1})^{d_v}. \tag{6.23}$$

Obviously successful decoding corresponds to $\lim_{t \to +\infty} x_t = 0$ since then the error probability tends to zero. In section 6.7 we give the analysis of the density evolution equation and draw conclusions for this bit-wise BP error probability and its threshold behaviour.

DE equations for general BMS channels

Luckily it turns out that exactly the same type of analysis works for general binary memoryless symmetric (BMS) channels. The density evolution equations for the BEC (6.20), (6.21) are *polynomial* equations relating probabilities x_m and y_m of erasure messages. They also involve the channel erasure probability ϵ. For general BMS channels the density evolution equations are *integral* equations relating probability distributions for the messages of type $l_{i \to a}$ and $\widehat{l}_{a \to i}$ after

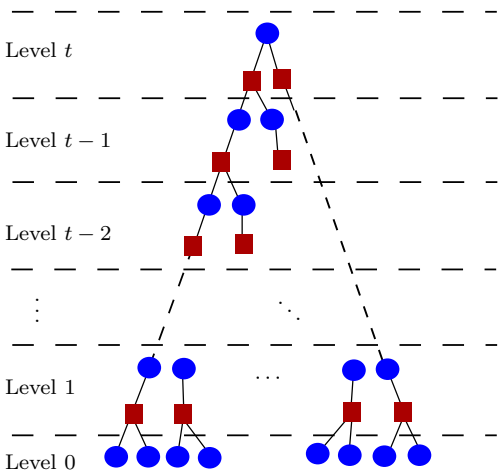


Figure 6.2 A computation graph of $(2,3)$-regular LDPC code with height t. The graph is split into $t+1$ levels.

a certain number of iterations. Besides they involve the channel distribution $\mathrm{c}(h|1)$.[5] There exists a beautiful and helpful algebra of "convolution" operations over probability distributions that allows one to directly transfer all the intuition gained on the BEC. We first define the convolution operations and state their main algebraic properties.

Variable node convolution: The first one is the standard convolution $\otimes$. Let l_1 and l_2 be two independent random variables with distributions $\mathrm{a}_1(l)$ and $\mathrm{a}_2(l)$. Then their sum $l = l_1 + l_2$ is distributed as

$$(\mathrm{a}_1 \otimes \mathrm{a}_2)(l) = \int_{\mathbb{R}^2} dl_1 a(l_1) dl_2 a(l_2) \delta(l - (l_1 + l_2)) \tag{6.24}$$

Check node convolution: The second type of convolution[6] is denoted by $\boxplus$ and is defined via by the distribution of $l = \operatorname{atanh}(\tanh l_1 \tanh l_2)$,

$$(\mathrm{a}_1 \boxplus \mathrm{a}_2)(l) = \int_{\mathbb{R}^2} dl_1 a(l_1) dl_2 a(l_2) \delta(l - \operatorname{atanh}(\tanh l_1 \tanh l_2)) \tag{6.25}$$

Algebraic properties of convolutions: It is clear that $\otimes$ is commutative and associative and that the neutral element is a Dirac mass at the origin $\Delta_0(l)$.[7] We leave it as an exercise to the reader to show that $\boxplus$ is also commutative, associative and that the neutral element is $\Delta_\infty(l)$, the Dirac mass at $+\infty$. Both

[5] We will pretend that all probability distributions have densities. This is not really true and it is important to take into account probability distributions which are convex combinations of densities and Dirac masses. However, practically, this makes no difference in the formalism except for introducing technicalities that only serve to obscure the picture.

[6] If we are willing to bring all random variables to a different domain $\boxplus$ becomes a usual convolution. We do not pursue this further here but it is useful to know that leads to computational efficient ways of performing the check node convolution in practice.

[7] In this context it is customary to use the notation $\Delta_0(l)$ instead of $\delta(l)$.

operations are linear: $(\mathrm{a}_1 + \mathrm{a}_2) \otimes \mathrm{a}_3 = (\mathrm{a}_1 \otimes \mathrm{a}_3) + (\mathrm{a}_2 \otimes \mathrm{a}_3)$ and $(\mathrm{a}_1 + \mathrm{a}_2) \boxplus \mathrm{a}_3 = (\mathrm{a}_1 \boxplus \mathrm{a}_3) + (\mathrm{a}_2 \boxplus \mathrm{a}_3)$. We stress that the two operations do not "mix" well together in the sense that $(\mathrm{a}_1 \otimes \mathrm{a}_2) \boxplus \mathrm{a}_3 \neq \mathrm{a}_1 \otimes (\mathrm{a}_2 \boxplus \mathrm{a}_3)$. Also there is nothing like distributivity in the sense that $(\mathrm{a}_1 \boxplus \mathrm{a}_2) \otimes \mathrm{a}_3 \neq (\mathrm{a}_1 \otimes \mathrm{a}_3) \boxplus (\mathrm{a}_2 \otimes \mathrm{a}_3)$.

We are now ready to derive the density evolution equations. Consider again the computation *tree* T_t with height t, with the division into $t+1$ levels, from 0 to t as before (Fig. 6.2). Look at level $m+1$. At a variable node, the incoming messages are independent (real valued) random variables sent by the $d_v - 1$ children check nodes. Let these messages be $\widehat{l}_1, \ldots, \widehat{l}_{d_v-1}$ and their common distribution $\mathrm{y}_m(\widehat{l})$. The BP equations tell us that the outgoing message from the variable node to the check node (both at level $m+1$) is

$$l = h + \widehat{l}_1 + \cdots + \widehat{l}_{d_v-1}$$

Let $\mathrm{x}_{m+1}(l)$ denote the probability distribution of the outgoing message. Since the outgoing random variable is the sum of independent random variables, the density of the outgoing random variable is the convolution of the densities of the incoming random variables,

$$\mathrm{x}_{m+1} = \mathrm{c} \otimes \mathrm{y}_m^{\otimes d_v - 1}. \tag{6.26}$$

Here we use the notation $\mathrm{y}_m^{\otimes d_v-1}$ for $\mathrm{y}_m \otimes \cdots \otimes \mathrm{y}_m$ convolved $d_v - 1$ times. This equation is the analog of (6.21). Now we seek an equation for y_m in terms of x_m. At check nodes of level $m+1$ the incoming messages are $d_c - 1$ independent random variables coming from the children variable nodes of level m. Call the random messages $l_1, \cdots, l_{d_c-1}$ and denote their probability distribution by $\mathrm{x}_m(l)$. From the BP equations the outgoing message from check nodes to the variable node (both at level $m+1$) is

$$\widehat{l} = \operatorname{atanh}\left(\prod_{i=1}^{d_c-1} \tanh l_i)\right)$$

and we have for the probability densities

$$\mathrm{y}_m = \mathrm{x}_m^{\boxplus d_c - 1}\,. \tag{6.27}$$

As above, we use the notation $\mathrm{x}_m^{\boxplus d_c-1}$ for $\mathrm{x}_m \boxplus \cdots \boxplus \mathrm{x}_m$ convolved $d_c - 1$ times. This equation is the analog of (6.20).

Equations (6.26) and (6.27) are the *DE equations for general BMS channels*. Combining them into a single equation yields

$$\mathrm{x}_{m+1} = \mathrm{c} \otimes (\mathrm{x}_m^{\boxplus d_c - 1})^{\otimes d_v - 1} \tag{6.28}$$

(also traditionally called DE equation). These iterations are initialised with $\mathrm{x}_0(l) = \mathrm{c}(l|1)$.

We can now compute the bit-wise probability of error of the BP decoder. In the final step the BP algorithm computes the log-likelihood ratio associated to

the root node as a sum of all messages incoming from d_v children check nodes plus the one coming from the channel,

$$l = h + l_1 + \cdots l_{d_v} .$$

Since on the computation tree all messages are independent, the distribution of l is equal to $\mathrm{c} \otimes (\mathrm{y}_{t-1})^{\otimes d_v}$, or

$$\mathrm{c} \otimes (\mathrm{x}_{t-1}^{\boxplus d_c - 1})^{\otimes d_v} .$$

From (6.12) and (6.15) we see that the average bit error probability is

$$\mathbb{P}_{\mathrm{BP,b}}(d_v, d_c, \epsilon, t) = \int_{-\infty}^{+\infty} dl \, \frac{1}{2}(1 - \mathrm{sign}(l))(\mathrm{c} \otimes (\mathrm{x}_{t-1}^{\boxplus d_c - 1})^{\otimes d_v})(l) . \tag{6.29}$$

This can also be written as

$$\int_{-\infty}^{0_-} dl \, (\mathrm{c} \otimes (\mathrm{x}_{t-1}^{\boxplus d_c - 1})^{\otimes d_v})(l) + \frac{1}{2} \int_{0_-}^{0_+} dl \, (\mathrm{c} \otimes (\mathrm{x}_{t-1}^{\boxplus d_c - 1})^{\otimes d_v})(l) . \tag{6.30}$$

The second term takes into account the possibility of a Dirac mass at the origin which corresponds to an erasure for which a decoding decision is taken by flipping a coin. Successful decoding corresponds to x_t approaching $\Delta_{+\infty}$ as $t \to +\infty$ since then the error probability approaches zero. This is easily seen at a formal level just by using that $\Delta_{+\infty}$ is a neutral element for $\boxplus$ and an absorbing element for $\otimes$. For mathematically rigorous techniques of analysis the interested reader can consult Section 6.8.

Of course, the DE equation (6.28) and the error probability (6.30) reduce to the BEC expressions. On the BEC and under the all-zero codeword assumption, the messages remain in the alphabet $\{0, +\infty\}$. Thus all densities are parametrized as as $\mathrm{c}(h|1) = \epsilon\Delta_0(h) + (1-\epsilon)\Delta_\infty(h)$ and $\mathrm{x}_m(l) = x_m\Delta_0(l) + (1-x_m)\Delta_\infty(l)$, $\mathrm{y}_m(l) = y_m\Delta_0(l) + (1-y_m)\Delta_\infty(l)$. It is an instructive exercise to recover (6.22), (6.23) from this parametrization.

6.7 Analysis of DE Equations for the BEC

We have seen that the bit probability of error of the BP decoder (6.23) can be computed from the DE recursions (6.22). We will show here that a threshold phenomenon appears. Namely there is a noise threshold ϵ_{BP}, called the BP-threshold, such that for $\epsilon < \epsilon_{\mathrm{BP}}$ the limit of $\mathbb{P}_{\mathrm{BP,b}}(d_v, d_c, \epsilon, t)$ vanishes when the number of iterations $t \to +\infty$, while for $\epsilon > \epsilon_{\mathrm{BP}}$ this limit remains strictly positive.

In order to compute $\lim_{t\to+\infty} \mathbb{P}_{\mathrm{BP,b}}(d_v, d_c, \epsilon, t)$ we have to analyze the recursion $x_t = f(\epsilon, x_{t-1})$ where

$$f(\epsilon, x) = \epsilon(1 - (1-x)^{d_c - 1})^{d_v - 1} \tag{6.31}$$

and the initial condition is $x_0 = 1$.[8] We ask whether the sequence $\{x_t\}$ converges to 0 or not. In case it does, the decoding is successful, otherwise it fails.

Note that the function $f(\epsilon, x)$ is increasing in ϵ and x for $\epsilon, x \in [0, 1]$. This is key to prove the following.

LEMMA 6.1 *Let $2 \leq d_v \leq d_c$ and $0 \leq \epsilon \leq 1$. Let $x_0 = 1$ and $x_t = f(\epsilon, x_{t-1})$, $t \geq 1$. Then (a) The sequence $\{x_t\}$ is decreasing in t; (b) If $\epsilon' \leq \epsilon$ then $x_t(\epsilon') \leq x_t(\epsilon)$.*

Proof Let us first show that the sequence $\{x_t\}$ is decreasing. We use induction. The first two elements of the sequence are $x_0 = 1$ and $x_1 = f(\epsilon, x_0) = \epsilon$, so $x_0 \geq x_1$. Therefore, for $t \geq 2$, we assume $x_{t-1} \leq x_{t-2}$ as the induction hypothesis. Since $f(\epsilon, x)$ is increasing in x, we obtain $f(\epsilon, x_{t-1}) \leq f(\epsilon, x_{t-2})$. The left hand side is equal to x_t, and the right hand side to x_{t-1}, and we deduce that $x_t \leq x_{t-1}$. To prove the second claim, we use induction once more. Assume that $\epsilon' \leq \epsilon$. Then $x_1(\epsilon') = \epsilon' \leq \epsilon = x_1(\epsilon)$. The general statement is deduced as follows,

$$x_t(\epsilon') = f(\epsilon', x_{t-1}(\epsilon')) \leq f(\epsilon, x_{t-1}(\epsilon')) \leq f(\epsilon, x_{t-1}(\epsilon)) = x_t(\epsilon),$$

where the first inequality follows from the fact that $f(\epsilon, x)$ is increasing in ϵ, and the second inequality follows from it being increasing in x together with the induction hypothesis. □

From part (a) of lemma 6.1, it follows that $x_t(\epsilon)$ converges to a limit in $[0, 1]$, $\lim_{t\to+\infty} x_t(\epsilon) = x_\infty(\epsilon)$. From the continuity of the function (6.31) we conclude that the limit of the density eveolution iterations is a solution of the fixed point equation

$$x_\infty(\epsilon) = f(\epsilon, x_\infty(\epsilon)). \tag{6.32}$$

From part (b) of the lemma, it follows that if $x_t(\epsilon) \to 0$ for some ϵ, then $x_t(\epsilon') \to 0$ for all $\epsilon' < \epsilon$. Let $x_\infty(\epsilon) = \lim_{t\to\infty} x_t(\epsilon)$. Then $x_\infty(\epsilon)$, as well as the error probability

$$\lim_{t\to+\infty} \mathbb{P}_{\mathrm{BP,b}}(d_v, d_c, \epsilon, t) = \frac{\epsilon}{2}(1 - (1 - x_\infty(\epsilon))^{d_v - 1})^{d_c}, \tag{6.33}$$

are increasing in ϵ as shown in Figure 6.3. Hence we can define the quantity

$$\epsilon_{\mathrm{BP}} = \sup\{\epsilon : x_\infty(\epsilon) = 0\}$$

which we call *the BP threshold*.

There is a graphical way to characterize this threshold. Note that $x_\infty(\epsilon)$ is a solution of the fixed point equation $x = f(\epsilon, x)$. Thus, if $f(\epsilon, x) - x < 0$ for all $x \in [0, \epsilon]$, then $x_\infty(\epsilon) = 0$. For the converse, as soon as there is a fixed point $f(\epsilon, x) = x$ in the interval $]0, \epsilon]$, we have that $x_\infty > 0$. In fact it is easy to check that this condition can be further simplified since there never can be a fixed

[8] Strictly speaking we should set $x_0 = \epsilon$. But since, if we set $x_0 = 1$ then $x_1 = \epsilon$, we may as well start iterations with $x_0 = 1$.

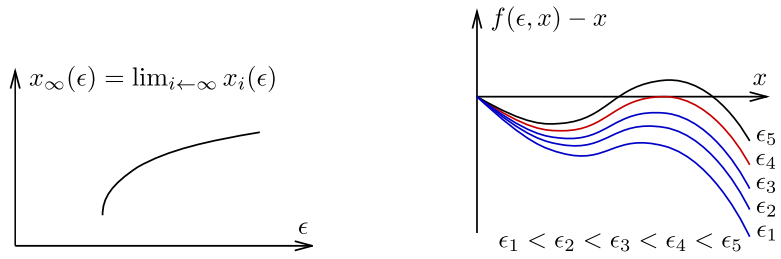


Figure 6.3 *Left*: Monotonicity of x_∞ as a function of ϵ. It is not difficult to show that for $d_v \geq 3$, $d_c > d_v$, x_∞ jumps at the threshold, and for $d_v = 2$, $d_c > d_v$ x_∞ changes continuously at the threshold. *Right*: The threshold ϵ_{BP} is the largest channel parameter so that $f(\epsilon, x) - x < 0$ for the whole range $x \in [0, 1]$. The picture here corresponds to the case $d_v \geq 3$.

point in $]\epsilon, 1]$ as $f(\epsilon, x) < \epsilon$. Therefore, if $f(\epsilon, x) - x < 0$ for all $x \in [0, 1]$, then $x_\infty = 0$. For the converse, as soon as there is a fixed point $f(\epsilon, x) = x$ in the interval $]0, 1]$, we have that $x_\infty(\epsilon) > 0$. This condition is graphically depicted in Figure 6.3.

EXAMPLE 20 For the $(3, 6)$-regular ensemble, we get $\epsilon_{\text{BP}} \approx 0.4294$. Note that the rate of this ensemble is $R = 1 - \frac{d_v}{d_c} = \frac{1}{2}$. Therefore, the fraction 0.4294 has to be compared to the erasure probability that an optimum code (say, a random linear code) could tolerate, which is $\epsilon_{\text{Shannon}} = 1 - R = \frac{1}{2}$. We conclude that already this very simple code, together with this very simple decoding procedure can decode up to a good fraction of the channel capacity. □

6.8 Analysis of DE equations for general BMS channels[9]

The elementary analysis for the BEC can be extended to the class of general symmetric channels. Although the main ideas are the same, the functional nature of the DE equation (6.28), $\mathrm{x}_{t+1} = f(\mathrm{c}, \mathrm{x}_t)$ with

$$f(\mathrm{c}, \mathrm{x}) = \mathrm{c} \otimes (\mathrm{x}^{\boxplus d_c - 1})^{\otimes d_v - 1}, \tag{6.34}$$

makes the analysis technically more challenging. Here we give a brief sketch of the theory.

[9] This section is not needed for the main development and can be skipped in a first reading.

Ordering by degradation of symmetric distributions

The analysis for the BEC rests on the monotonicity in ϵ and x of the function $f(\epsilon, x)$. We will need analogous properties for the functional (6.34) on the right hand side of the DE recursion (6.28). The key is to introduce a partial order relation between distributions.

The DE equations preserve the symmetry property of the initial channel distribution. In other words when we initialize the DE recursion with $\mathrm{x}_0(l) = \mathrm{c}(l|1)$, which satisfies the symmetry condition $\mathrm{c}(l|1) = e^{-2l}\mathrm{c}(-l|1)$, we have for all $t \geq 1$, $\mathrm{x}_{t+1}(l) = e^{-2l}\mathrm{x}_t(-l)$. For this reason, we may restrict ourselves to the space of *symmetric distributions* satisfying

$$\mathrm{a}(l) = e^{-2l}\mathrm{a}(-l)\,. \tag{6.35}$$

Define the moments

$$M_k(\mathrm{a}) = \int dl\, \mathrm{a}(l)(\tanh l)^k.$$

It is not difficult to see that the symmetry condition implies

$$M_{2k-1}(\mathrm{a}) = M_{2k}(\mathrm{a}) \tag{6.36}$$

for all integers $k \geq 1$. Symmetric distributions can be entirely characterized by their even moments: if two symmetric distributions a and b have the *same* set of even moments, $M_{2k}(\mathrm{a}) = M_{2k}(\mathrm{b})$, then they must be equal. Indeed, by the symmetry condition their odd moments are also equal, and since all moments are less than 1, Carleman's criterion[10] is satisfied; thus one can reconstruct a unique measure from the set of even moments, which implies $\mathrm{a} = \mathrm{b}$.

Let us now define *ordering by degradation*. We say that a_2 is degraded with respect to a_1, and write $\mathrm{a}_2 \succ \mathrm{a}_1$ if an only if $M_{2k}(\mathrm{a}_2) \leq M_{2k}(\mathrm{a}_1)$ for all integers $k \in \mathbb{N}^*$. The following example gives the intuitive meaning of this concept.

EXAMPLE 21 Consider the likelihood distribution of the BEC channel $\mathrm{c}_\epsilon(h|1) = \epsilon\Delta_0(h) + (1-\epsilon)\Delta_\infty(h)$. Note that it is symmetric and that the moments are $M_{2k-1} = M_{2k} = 1 - \epsilon$ for $k \geq 1$. Take two channels c_{ϵ_1} and c_{ϵ_2} with $\epsilon_2 > \epsilon_1$. According to our definition we have $\mathrm{c}_{\epsilon_2} \succ \mathrm{c}_{\epsilon_1}$ because $1 - \epsilon_2 < 1 - \epsilon_1$; in other words "$\mathrm{c}_{\epsilon_2}$ is degraded with respect to c_{ϵ_1}" means that "c_{ϵ_2} is more noisy than c_{ϵ_1}". We leave it as an exercise to the reader to show that the same interpretation applies to our other basic symmetric channels, the BSC and BAWGNC. □

As a side remark note that we can associate a BMS channel to any symmetric distribution satisfying (6.35). The idea is to think of the distribution as a "likelihood distribution" for some channel. The transition probability of the channel can be explicitely constructed through the identities $p(y|+1)dy = a(l)dl$, $p(-y|-1) = p(y|1)$ and $l = \frac{1}{2}\ln\frac{p(y|+1)}{p(y|-1)}$. There is an intuitive characterization

[10] Carleman's criterion states that: if a sequence of finite real numbers $\{m_0 = 1, m_1, m_2, \cdots\}$ are the moments of a probability distribution on $\mathbb{R}$, i.e., $m_k = \int dF(x)x^k$, and furthermore $\sum_{k=1}^{+\infty}(m_{2k})^{-1/2k} = +\infty$, then there is a *unique* such probability distribution.

of the relation $\mathrm{a}_2 \succ \mathrm{a}_1$ in terms of the associated channels $p_2(y|s)$ and $p_1(y|s)$. Namely there must exist a channel $q(y|s)$ such that $p_2(z|s) = \sum_y q(z|y)p_1(y|s)$. In other words the channel associated to a_2 is more noisy than the one associated to a_1.

Ordering by degradation is *preserved* under the two convolutions operations $\otimes$ and $\boxplus$. More precisely if $\mathrm{a}_1 \succ \mathrm{a}_2$ and b are symmetric distributions we have:

$$\mathrm{a}_2 \otimes \mathrm{b} \succ \mathrm{a}_1 \otimes \mathrm{b} \quad \text{and} \quad \mathrm{b} \boxplus \mathrm{a}_2 \succ \mathrm{b} \boxplus \mathrm{a}_1 \,.$$

The proof of these assertions is the subject of an exercise.

Entropy distance, entropy functional and moment expansions

For the BEC, besides monotonicity of $f(\epsilon, x)$, another important ingredient was the continuity of the function with respect to ϵ and x. Here we introduce a suitable distance in the space of symmetric distributions that allows to prove analogous statements. We do not wish to introduce sophisticated topological language here and we proceed in a pedestrian way that will be sufficient for our purposes.

For any two symmetric distributions a and b define

$$d(\mathrm{a}, \mathrm{b}) = \sum_{k \geq 1} \frac{|M_{2k}(\mathrm{a}) - M_{2k}(\mathrm{b})|}{2k(2k-1)}. \tag{6.37}$$

It is easy to see that this is a well defined distance, i.e. it is symmetric, satisfies the triangle inequality and vanishes if and only if $\mathrm{a} = \mathrm{b}$. We call it the *entropy distance*.

Let us show that this distance is naturally related to a notion a entropy. We define the *entropy functional*

$$H[\mathrm{x}] = \int dl\, \mathrm{x}(l) \ln(1 + e^{-2l}) \tag{6.38}$$

which is in fact precisely the Shannon conditional entropy $H(X|Y)$ corresponding to a symmetric channel $p(y|s)$ whose likelihood distribution is $\mathrm{x}(l)$ where $l(y) = \frac{1}{2} \ln \frac{p(y|+1)}{p(y|-1)}$ and $y \sim p(y|1)$. Using the expansion

$$\ln(1 + e^{-2l}) = \ln 2 - \ln(1 + \tanh l) = \ln 2 - \sum_{k=1}^{+\infty} \frac{(-1)^{k+1}}{k} (\tanh l)^k$$

and the equality of even and odd moments we get the *moment expansion* of the entropy functional

$$H[\mathrm{x}] = \ln 2 - \sum_{k=1}^{+\infty} \frac{M_{2k}(\mathrm{x})}{2k(2k-1)} \,.$$

Now, by linearity of the entropy functional (6.38)

$$H[\mathrm{a}-\mathrm{b}] = \sum_{k=1}^{+\infty} \frac{M_{2k}(\mathrm{b}) - M_{2k}(\mathrm{a})}{2k(2k-1)}. \tag{6.39}$$

which implies (6.39)

$$d(\mathrm{a},\mathrm{b}) = H[\mathrm{a}-\mathrm{b}], \quad \text{if} \quad \mathrm{a} \succ \mathrm{b}. \tag{6.40}$$

(recall that $\mathrm{a} \succ \mathrm{b}$ was defined as $M_{2k}(\mathrm{a}) < M_{2k}(\mathrm{b})$ for all $k \geq 1$).

All continuity and convergence statements in the next paragraph are base on the following two handy inequalities. For $\mathrm{a} \succ \mathrm{b}$ and any symmetric x

$$\begin{cases} H[\mathrm{x} \otimes (\mathrm{a}-\mathrm{b})] \leq H[\mathrm{a}-\mathrm{b}] = d(\mathrm{a},\mathrm{b}), \\ H[\mathrm{x} \boxplus (\mathrm{a}-\mathrm{b})] \leq H[\mathrm{a}-\mathrm{b}] = d(\mathrm{a},\mathrm{b}) \end{cases}$$

To prove the second inequality we use the moment expansion (6.39) and the fact that moments are multiplicative for the $\boxplus$ operation, $M_{2k}(\mathrm{a} \boxplus \mathrm{b}) = M_{2k}(\mathrm{a})M_{2k}(\mathrm{b})$ (see exercises)

$$\begin{aligned} H[\mathrm{x} \boxplus (\mathrm{a}-\mathrm{b})] &= \sum_{k=1}^{+\infty} \frac{M_{2k}(\mathrm{x} \boxplus \mathrm{b}) - M_{2k}(\mathrm{x} \boxplus \mathrm{a})}{2k(2k-1)} \\ &= \sum_{k=1}^{+\infty} M_{2k}(\mathrm{x}) \frac{M_{2k}(\mathrm{b}) - M_{2k}(\mathrm{a})}{2k(2k-1)} \\ &\leq \sum_{k=1}^{+\infty} \frac{M_{2k}(\mathrm{b}) - M_{2k}(\mathrm{a})}{2k(2k-1)} \\ &= H[\mathrm{a}-\mathrm{b}] \end{aligned}$$

The first inequality is less straightforward because the moments are not mutiplicative for the usual convolution $\otimes$. But we can use the *duality rule* $H((\mathrm{a}-\mathrm{b}) \otimes (\mathrm{a}'-\mathrm{b}')) = -H((\mathrm{a}-\mathrm{b}) \boxplus (\mathrm{a}'-\mathrm{b}'))$ (see exercises) as follows

$$\begin{aligned} H[\mathrm{x} \otimes (\mathrm{a}-\mathrm{b})] &= H((\mathrm{x}-\Delta_\infty) \otimes (\mathrm{a}-\mathrm{b})] \\ &= -H((\mathrm{x}-\Delta_\infty) \boxplus (\mathrm{a}-\mathrm{b})] \\ &= \sum_{k=1}^{+\infty} M_{2k}(\Delta_\infty - \mathrm{x}) \frac{M_{2k}(\mathrm{b}) - M_{2k}(\mathrm{a})}{2k(2k-1)} \\ &= \sum_{k=1}^{+\infty} (M_{2k}(\Delta_\infty) - M_{2k}(\mathrm{x})) \frac{M_{2k}(\mathrm{b}) - M_{2k}(\mathrm{a})}{2k(2k-1)} \\ &\leq \sum_{k=1}^{+\infty} \frac{M_{2k}(\mathrm{b}) - M_{2k}(\mathrm{a})}{2k(2k-1)} \\ &= H[\mathrm{a}-\mathrm{b}] \end{aligned}$$

Analysis of DE recursion and the BP threshold

Let us first prove that the functional $f(c, \mathrm{x})$ on the right hand side of the DE recursions (6.28), is "increasing" with respect to the distributions c and x. Since ordering by degradation is preserved by convolution we obviously have $f(\mathrm{c}_2, \mathrm{x}) \succ f(\mathrm{c}_1, \mathrm{x})$ when $\mathrm{c}_2 \succ \mathrm{c}_1$. Now, notice that if $\mathrm{a}_2 \succ \mathrm{a}_1$ and $\mathrm{b}_2 \succ \mathrm{b}_1$ then $\mathrm{a}_2 \otimes \mathrm{b}_2 \succ \mathrm{a}_1 \otimes \mathrm{b}_2$ and $\mathrm{a}_1 \otimes \mathrm{b}_2 \succ \mathrm{a}_1 \otimes \mathrm{b}_1$, so also $\succ \mathrm{a}_2 \otimes \mathrm{b}_2 \succ \mathrm{a}_1 \otimes \mathrm{b}_1$. Generalizing, for $\mathrm{a}_i \succ \mathrm{b}_i$, $i = 1, \ldots, n$ we have $\mathrm{a}_1 \otimes \cdots \otimes a_n \succ \mathrm{b}_1 \otimes \cdots \otimes \mathrm{b}_n$. The same statements are true if we replace $\otimes$ by $\boxplus$. Thus for $\mathrm{x}_2 \succ \mathrm{x}_1$ we get $\mathrm{x}_2^{\boxplus d_c - 1} \succ \mathrm{x}_2^{\boxplus d_c - 1}$, and then $(\mathrm{x}_2^{\boxplus d_c - 1})^{\oplus d_v - 1} \succ (\mathrm{x}_2^{\boxplus d_c - 1})^{\oplus d_v - 1}$, and finaly $f(c, \mathrm{x}_2) \succ f(c, \mathrm{x}_1)$.

Consider a family of channels c_ϵ parametrized by ϵ (for example a noise level). We say that the *family of channels is ordered by degradation* when $\mathrm{c}_\epsilon \prec \mathrm{c}'_\epsilon$ for $\epsilon < \epsilon'$. The BEC, BEC or BAWGNC are three such families.

We are now ready to prove the analog of Lemma 6.1

LEMMA 6.2 *Let $2 \leq d_v \leq d_c$ and c_ϵ be family of channels ordered by degradation. Let $\mathrm{x}_0 = \Delta_0$ and $\mathrm{x}_t = f(\mathrm{c}_\epsilon, \mathrm{x}_{t-1})$, $t \geq 1$. Then (a) The sequence of distributions $\{\mathrm{x}_t\}$ is decreasing in t in the sense $\mathrm{x}_{t+1} \prec \mathrm{x}_t$; (b) If $\mathrm{c}_\epsilon \prec \mathrm{c}_{\epsilon'}$ then $\mathrm{x}_t(\mathrm{c}_\epsilon) \prec \mathrm{x}_t(\mathrm{c}_{\epsilon'})$.*

Proof We first show the claims by induction. We have $\mathrm{x}_0 = \Delta_0(\dot{)}$ and $\mathrm{x}_1 = f(\mathrm{c}, \mathrm{x}_0) = \mathrm{c}$, so $\mathrm{x}_0 \succ \mathrm{x}_1$. Therefore, for $t \geq 2$, we assume $\mathrm{x}_{t-1} \prec \mathrm{x}_{t-2}$ as the induction hypothesis. Since $f(\mathrm{c}, \mathrm{x})$ is increasing in x, we obtain $f(\mathrm{c}, \mathrm{x}_{t-1}) \prec f(\mathrm{c}, \mathrm{x}_{t-2})$ and we deduce that $\mathrm{x}_t \prec \mathrm{x}_{t-1}$. To prove the second claim assume that $\mathrm{c}_\epsilon \prec \mathrm{c}_{\epsilon'}$. Then $\mathrm{x}_1(\mathrm{c}_\epsilon) = \mathrm{c}_\epsilon \prec \mathrm{c}_{\epsilon'} = \mathrm{x}_1(\mathrm{c}_{\epsilon'})$. The general statement is deduced similarly to the case of the BEC: $\mathrm{x}_t(\mathrm{c}_\epsilon) = f(\mathrm{c}_\epsilon, \mathrm{x}_{t-1}(\mathrm{c}_\epsilon)) \prec f(\mathrm{c}_{\epsilon'}, \mathrm{x}_{t-1}(\mathrm{c}_\epsilon)) \prec f(\mathrm{c}_{\epsilon'}, \mathrm{x}_{t-1}(\mathrm{c}_{\epsilon'})) = \mathrm{x}_t(\mathrm{c}_{\epsilon'})$. □

Statement (a) of the Lemma says that DE iterations give a "decreasing" sequence of probability distributions $\mathrm{x}_0 = \Delta_0 \succ \mathrm{x}_1 = \mathrm{c} \succ \mathrm{x}_2 \succ \cdots \succ \mathrm{x}_t \succ \ldots$. This means that for each $k \geq 1$ we have an increasing sequence of moments $M_{2k}(\mathrm{x}_0) = 0 < M_{2k}(\mathrm{x}_1) = M_{2k}(\mathrm{c}) < M_{2k}(\mathrm{x}_2) < \ldots M_{2k}(\mathrm{x}_t) < \ldots$, and since this sequence is bounded by 1, it converges to a real number in $[0, 1]$. Let m_{2k}^∞ be the limits for each $k \geq 1$. Since even and odd moments are equal, odd moments also converge towards the same set of numbers $m_{2k-1}^\infty = m_{2k}^\infty$. Since $|m_k^\infty|^{-1/k} \geq 1$ Carleman's criterion[11] is satisfied thus the set of numbers $\{m_k^\infty\}$ are the moments of some probability distribution x_∞ with moments $M_{2k-1}(\mathrm{x}_\infty) = M_{2k}(\mathrm{x}_\infty) = m_{2k-1}^\infty = m_{2k}^\infty$. To summarize, we have $\mathrm{x}_t \to \mathrm{x}_\infty$ in the sense $d(\mathrm{x}_t, \mathrm{x}_\infty) \to 0$.

LEMMA 6.3 *The limiting distribution x_∞ is a solution of the DE fixed point equation $\mathrm{x}_\infty = f(\mathrm{c}, \mathrm{x}_\infty)$.*

Proof In the case of the BEC this statement was quite trivially obtained directly from the continuity of $f(\epsilon, x)$. For general channels we use the tools introduced

[11] namely that $\sum_{k=1}^{+\infty} (m_{2k}^\infty)^{-1/2k} = +\infty$

above. It is sufficient to show $d(\mathrm{x}_\infty, f(\mathrm{c}, \mathrm{x}_\infty)) = 0$ because then all moments of x_∞ and $f(\mathrm{c}, \mathrm{x}_\infty)$ are equal and by Carleman's criterion the two distributions must be equal. By the triangle inequality for any t,

$$d(\mathrm{x}_\infty, f(\mathrm{c}, \mathrm{x}_\infty)) \le d(\mathrm{x}_\infty, \mathrm{x}_{t+1}) + d(\mathrm{x}_{t+1}, f(\mathrm{c}, \mathrm{x}_t)) + d(f(\mathrm{c}, \mathrm{x}_t), f(\mathrm{c}, \mathrm{x}_\infty))\,.$$

The second term vanishes because $\mathrm{x}_{t+1} = f(\mathrm{c}, \mathrm{x}_t)$. We now argue that the limits of the first and third terms when $t \to +\infty$ vanish. By construction of x_∞, $\lim_{t\to+\infty} M_{2k}(\mathrm{x}_t) = M(\mathrm{x}_\infty)$, which implies $\lim_{t\to+\infty} d(\mathrm{x}_\infty, \mathrm{x}_{t+1}) = 0$ by dominated convergence. To compute the limit of the third term we recall that $\mathrm{x}_t \succ \mathrm{x}_\infty$ so

$$\begin{aligned}
d(f(\mathrm{c}, \mathrm{x}_t), f(\mathrm{c}, \mathrm{x}_\infty)) &= H(f(\mathrm{c}, \mathrm{x}_t) - f(\mathrm{c}, \mathrm{x}_\infty) \\
&= H(\mathrm{c} \otimes ((\mathrm{x}_t^{\boxplus d_c - 1})^{\otimes d_v - 1} - (\mathrm{x}_\infty^{\boxplus d_c - 1})^{\otimes d_v - 1})) \\
&\le H((\mathrm{x}_t^{\boxplus d_c - 1})^{\otimes d_v - 1} - (\mathrm{x}_\infty^{\boxplus d_c - 1})^{\otimes d_v - 1}) \\
&= H((\mathrm{x}_t^{\boxplus d_c - 1} - \mathrm{x}_\infty^{\boxplus d_c - 1} + \mathrm{x}_\infty^{\boxplus d_c - 1})^{\otimes d_v - 1} - (\mathrm{x}_\infty^{\boxplus d_c - 1})^{\otimes d_v - 1}) \\
&= \sum_{p=1}^{d_v - 1} \binom{d_v - 1}{p} H((\mathrm{x}_t^{\boxplus d_c - 1} - \mathrm{x}_\infty^{\boxplus d_c - 1})^{\otimes p} \otimes (\mathrm{x}_\infty^{\boxplus d_c - 1})^{\otimes d_v - 1 - p}) \\
&\le \sum_{p=1}^{d_v - 1} \binom{d_v - 1}{p} H(\mathrm{x}_t^{\boxplus d_c - 1} - \mathrm{x}_\infty^{\boxplus d_c - 1}) \\
&= (2^{d_v - 1} - 1) H(\mathrm{x}_t^{\boxplus d_c - 1} - \mathrm{x}_\infty^{\boxplus d_c - 1})\,.
\end{aligned}$$

The last entropy is estimated thanks to similar tricks,

$$\begin{aligned}
H(\mathrm{x}_t^{\boxplus d_c - 1} - \mathrm{x}_\infty^{\boxplus d_c - 1}) &= H((\mathrm{x}_t - \mathrm{x}_\infty + \mathrm{x}_\infty)^{\boxplus d_c - 1} - \mathrm{x}_\infty^{\boxplus d_c - 1}) \\
&= \sum_{q=1}^{d_c - 1} \binom{d_c - 1}{q} H((\mathrm{x}_t - \mathrm{x}_\infty)^{\boxplus q} \boxplus \mathrm{x}_\infty^{\boxplus d_c - 1 - q}) \\
&\le (2^{d_c - 1} - 1) H(\mathrm{x}_t - \mathrm{x}_\infty)\,.
\end{aligned}$$

Putting these results together we obtain the simple inequality

$$\begin{aligned}
d(f(\mathrm{c}, \mathrm{x}_t), f(\mathrm{c}, \mathrm{x}_\infty)) &\le (2^{d_v - 1} - 1)(2^{d_c - 1} - 1) H(\mathrm{x}_t - \mathrm{x}_\infty) \\
&= (2^{d_v - 1} - 1)(2^{d_c - 1} - 1) d(\mathrm{x}_t, \mathrm{x}_\infty)
\end{aligned}$$

which implies (by an argument above) $\lim_{t\to+\infty} d(f(\mathrm{c}, \mathrm{x}_t), f(\mathrm{c}, \mathrm{x}_\infty)) = 0$. □

From statement (b) of the lemma, it follows that if $\mathrm{x}_t(\mathrm{c}_\epsilon) \to \Delta_\infty$ (in the sense that $d(\mathrm{x}_t, \Delta_\infty) \to 0$) for a channel c_ϵ, then $\mathrm{x}_t(\mathrm{c}_{\epsilon'}) \to \Delta_\infty$ for a less noisy channel $\mathrm{c}_{\epsilon'} \prec \mathrm{c}_\epsilon$. Hence we can define a *BP threshold* as

$$\epsilon_{\mathrm{BP}} = \sup\{\epsilon : \mathrm{x}_\infty(\epsilon) = \Delta_\infty\}$$

Not surprisingly, with a bit more work, one can show that the DE fixed point

allows to calculate the probability of error

$$\lim_{t\to+\infty} \mathbb{P}_{\mathrm{BP,b}}(d_v, d_c, \epsilon, t) = \int_{-\infty}^{+\infty} dl\, \frac{1}{2}(1-\mathrm{sign}(l))(\mathrm{c}_\epsilon \otimes (\mathrm{x}_\infty^{\boxplus d_c-1})^{d_v})(l), \quad (6.41)$$

For $\epsilon < \epsilon_{\mathrm{BP}}$ we have $\mathrm{x}_\infty = \Delta_\infty$ which yields a vanishing probability of error. It is also possible to show that above ϵ_{BP} this is an increasing function of ϵ.

EXAMPLE 22 If we consider e.g., the BSC, then DE predicts a threshold for the $(3,6)$-ensemble of $\epsilon^{\mathrm{BP}} = 0.084$. This means that as long as the channel introduces fewer than 8.4 percent errors, the BP decoder will with high probability be able to recover the correct codeword from the received word. Note that for rate one-half the maximum number of errors which a capacity-achieving code can tolerate is around 11 percent. So we see that, as for the BEC, the simple $(3,6)$-regular ensemble achieves a good fraction of capacity under BP decoding. □

6.9 Exchange of limits

At this point some readers might be slightly worried. We have defined density evolution by looking at the errors which remain after t iterations when we take the blocklength to infinity. Subsequently we have analyzed DE by looking what happens if we take more and more iterations. In short, we have looked at the limit $\lim_{t\to\infty}\lim_{n\to\infty}$.

This is certainly a valid limit, but if the implication is sensitive to the order in which we take the limit then one might worry how well experiments for "practical" block lengths of lets say thousands to hundreds of thousands of bits and "practical number of iterations" lets say dozens to hundreds of iterations might fit the theory. At least for the BEC there is a fairly simple and straightforward analytic answer: the limit is the same regardless of the order and can also be taken jointly.

We will not prove this result here. The key is to consider the converse limit $\lim_{n\to\infty}\lim_{t\to\infty}$ and to prove that it gives the same result. Note that due to the special nature of the BEC, the performance is monotonically decreasing in the number of iterations (things only can get better if we perform further iterations). From this basic observation we can deduce the following: Let $t(n)$ be any increasing function so that $t(n)$ tends to infinity if n tends to infinity. Then, for any channel parameter ϵ, the error probability under the limit $\lim_{n\to\infty}\lim_{t\to\infty}$ is no larger than the error probability under the joint limit when $t = t(n)$, which in turn is no larger than the error probability under the limit $\lim_{t\to\infty}\lim_{n\to\infty}$. If now we can show that the two extreme cases have the same limit, then any joint limit also has this same limit.

For the BEC the limit $\lim_{n\to\infty}\lim_{t\to\infty}$ can in fact be analyzed. The technique is to use the so-called *Wormald* method, a method which we will encounter soon when we will analyze simple algorithms to solve the K-SAT problem.

For the general case the situation is more complicated. Numerical experiments and analytic arguments show that also in the general case the limit does not depend on the order. But in order to show this rigorously one currently has to impose some further constraints on the ensemble.

6.10 BP versus MAP thresholds

This is a good point to make a small digression on issues treated in detail in part III. In the language of statistical mechanics the BP threshold corresponds to a *dynamical* phase transition in the sense that we have here a sharp change in behaviour of an algorithm. The MAP probability of error also displays a threshold behaviour in the limit of infinite block length: it vanishes for $\epsilon < \epsilon_{\text{MAP}}$ and is strictly positive for $\epsilon > \epsilon_{\text{MAP}}$. Clearly we always have $\epsilon_{\text{BP}} < \epsilon_{\text{MAP}}$ since the MAP decoder is the one among all decoders that minimizes the error probability. There is an important conceptual difference between the two thresholds. The MAP threshold can be shown to be a singularity of the infinite block-length Shannon conditional entropy (3.23) (further averaged on the code ensemble)

$$\lim_{n\to+\infty} \frac{1}{n}\mathbb{E}[H(\underline{X}|\underline{Y})]$$

or, in view of (3.24), of the free energy in thermodynamic limit. This entropy is a continuous convex function of ϵ, which vanishes for $\epsilon \leq \epsilon_{\text{MAP}}$ and is strictly positive for $\epsilon > \epsilon_{\text{MAP}}$. Thus ϵ_{MAP} is a non-analyticity point and corresponds to a *thermodynamic* phase transition in the sense introduced in Chapters 2 and 4. However the infinite block-length Shannon conditional entropy has *no* singularity at the BP threshold. This is an instance of the generic fact that dynamical thresholds related to algorithms are not visible on free energies. As we will see in part III, very interestingly and perhaps surprisingly from the point of view of coding, although the MAP and BP phase transitions are of a different conceptual nature, they are deeply related. In particular, in turns out we can compute the MAP threshold and probability of error from the very same DE equations which determine the BP threshold.

6.11 Notes

Gallager 1963. BEC: Luby et al, BMS Rich Urb, Techniques in Modern book, Exch of limits KU. (and this is what was done in (Luby et al. 1997). see (exchange of limits) **??**) to do

Problems

6.1 Restriction to the all-zero codeword. Use the symmetry of the BP decoder to prove that one can restrict to the all-zero input codeword in Equ. (6.15).

6.2 Algebraic properties of convolutions. Consider the variable node and check node convolution operations, $\otimes$ and $\boxplus$, defined in (6.24) and (6.25). Show the following properties:

(i) $\otimes$ and $\boxplus$ are are commutative and associative.

(ii) Δ_0 is the neutral element for $\otimes$ and Δ_∞ is the neutral element for $\boxplus$. Explicitly, $\Delta_0 \otimes \mathrm{a} = \mathrm{a}$ and $\Delta_\infty \boxplus \mathrm{a} = \mathrm{a}$.

(iii) Δ_∞ is an absorbing element for $\otimes$ and Δ_0 is an absorbing element for $\boxplus$. Explicitly, $\Delta_\infty \otimes \mathrm{a} = \Delta_\infty$ and $\Delta_0 \boxplus \mathrm{a} = \Delta_0$.

(iv) Check the linearity $\mathrm{a}_1 \otimes (\mathrm{a}_2 + \mathrm{a}_3) = \mathrm{a}_1 \otimes \mathrm{a}_2) + (\mathrm{a}_2 \otimes \mathrm{a}_3)$ and $\mathrm{a}_1 \boxplus (\mathrm{a}_2 + \mathrm{a}_3) = \mathrm{a}_1 \boxplus \mathrm{a}_2) + (\mathrm{a}_2 \boxplus \mathrm{a}_3)$.

(v) Find a counterexample to show that the two convolutions do not mix well: $(\mathrm{a}_1 \otimes \mathrm{a}_2) \boxplus a_3 \neq \mathrm{a}_1 \otimes (\mathrm{a}_2 \boxplus a_3)$.

(vi) Show that there is no distributivity: $(\mathrm{a}_1 \boxplus \mathrm{a}_2) \otimes \mathrm{a}_3 \neq (\mathrm{a}_1 \otimes \mathrm{a}_2) \boxplus (\mathrm{a}_2 \otimes \mathrm{a}_3)$.

6.3 From BMS to BEC. Reduce the DE equations (6.28), (6.30) valid for general BMS channels to the polynomial equations (6.22), (6.23) of the BEC case.

6.4 Ordering by degradation. We defined "ordering by degradation" for symmetric distributions by $\mathrm{a}_2 \succ \mathrm{a}_1$ if and only if $M_{2k}(\mathrm{a}_2) \leq M_{2k}(\mathrm{a}_1)$ for all integers $k \geq 1$. Take $\mathrm{a}_2 \succ \mathrm{a}_1$ and show that:

(i) $\mathrm{a}_2 \otimes \mathrm{b} \succ \mathrm{a}_1 \otimes \mathrm{b}$ and $\mathrm{b} \boxplus \mathrm{a}_2 \succ \mathrm{b} \boxplus \mathrm{a}_1$ for any symmetric distribution b.

(ii) There exist a channel $q(y|x)$ such that $p_2(z|x) = \sum_y q(z|y)p_1(y|x)$ where $p_{1,2}(y|x)$ are the transition probabilities associated to $\mathrm{a}_{1,2}$.

6.5 Two useful identities. Show that for any two symmetric densities a and b:

(i) Moments are multplicative under $\boxplus$, $M_{2k}(\mathrm{a} \boxplus \mathrm{b}) = M_{2k}(\mathrm{a})M_{2k}(\mathrm{b})$

(ii) The duality rule holds, $H(\mathrm{a}) + H(\mathrm{b}) = H(\mathrm{a} \otimes \mathrm{b}) + H(\mathrm{a} \boxplus \mathrm{b})$.

(iii) The duality rule implies $H((\mathrm{a}_1 - \mathrm{a}_2) \otimes (\mathrm{a}_3 - \mathrm{a}_4)) = -H((\mathrm{a}_1 - \mathrm{a}_2) \boxplus (\mathrm{a}_3 - \mathrm{a}_4))$ where $\mathrm{a}_1, \mathrm{a}_2, \mathrm{a}_3, \mathrm{a}_4$ are symmetric distributions.

6.6 Belief Propagation for $(3, 6)$ Gallager ensemble and BAWGN Channel. Exercise 1.2 proposed to implement a program which can generate random elements from a regular Gallager ensemble. This can be used together with the BP algorithm to simulate transmission over a BAWGN channel.

Use elements from the $(3, 6)$-ensemble of length $n = 1024$. For every codeword sent, generate a new code in order to get the *ensemble average*. When transmitting with a binary linear code over a symmetric channel, we can in fact assume that the all-zero codeword was sent since the error probability is independent of the transmitted codeword. This simplifies our life since we do not need to implement an encoder. We assume that we send the all-zero codeword over a BAWGN channel. In spin language the input is $s_i^{\mathrm{in}} + 1 = (-1)^0$, $i = 1, \cdots, n$. The channel adds to each component s_i^{in} an independent Gaussian random variable with zero mean and variance σ^2. At the receiver implement the message-passing decoder

in terms of likelihoods. Since a random element from the $(3, 6)$ ensemble typically does not have a tree-like factor graph the scheduling of the messages is important. To be explicit, use a *flooding* schedule. This means: send all *initial* messages from variable nodes to check nodes, then process these messages and send messages back from check nodes to all variable nodes. This corresponds to *one iteration*. For each codeword perform 100 iterations and then make the final decision for each bit.

Plot the negative logarithm (base 10) of the resulting bit error probability as a function of the capacity of the BAWGN channel with variance σ^2. This capacity does not have a closed form but can be computed numerically by means of the integral

$$C_{\text{BAWGNC}} = 1 - \int_{-\infty}^{+\infty} dy \, \frac{e^{\frac{-y^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma} \log_2\bigl(1 + e^{\frac{y}{\sigma^2} - \frac{1}{2\sigma^2}}\bigr)$$

If the code and the decoder where optimal (in the sense that Shannon's capacity is achieved) and the length of the code were infinite, where should we see the threshold (rapid decay of error probability)?

6.7 Density Evolution via Population Dynamics. DE for transmission over the BEC is relatively easy to implement since in this case the "densities" are in fact numbers (erasure probabilities). For general channels, DE is more involved since it really involves the evolution of densities. These are the densities of messages seen at the various iterations when the BP message-passing decoder is implemented on an infinite ensemble for a fixed number of iterations.

A quick and dirty way of implementing DE for general channels is by means of a population dynamics approach. Here is how this works. Assume that transmission takes place over a given BMS channel and that we are using the (d_v, d_c)-regular Gallager ensemble. Random messages are simulated by populations of size N. The larger N the more accurate will be your result but the slower it will be. We have one population $\mathcal{V}_0$ simulating channel outputs, and populations $\mathcal{C}_m$, $\mathcal{V}_m$ simulating messages flowing out of check nodes and variable nodes, and corresponding to the m-th iteration in the following way.

(i) Pick an initial population $\mathcal{V}_0$. This set consists of N i.i.d log-likelihoods associated to the given BMS channel, assuming that the transmitted bit is 1 (we are using spin notation here). More precisely, each sample is created in the following way. Sample Y according to $p(y \mid s = 1)$. Compute the corresponding log-likelihood value $\frac{1}{2}\log(\frac{p(y|s=1)}{p(y|s=-1)})$ and call it H.

(ii) To compute $\mathcal{C}_m$ proceed as follows. Create N samples i.i.d in the following way. For each sample, call it V, pick $d_c - 1$ samples from $\mathcal{V}_{m-1}$ with repetitions. Let these samples be named $U_1, \ldots, U_{d_c-1}$. Compute

$$V = \tanh^{-1}\Bigl(\prod_{a=1}^{d_c-1} \tanh(U_a/2)\Bigr).$$

Note, these are exactly the message-passing rules at a check node.

(iii) To compute $\mathcal{V}_m$ proceed as follows. Create N samples iid in the following way. For each sample, call it U, pick $d_v - 1$ samples from $\mathcal{C}_m$ with repetitions. Let these samples be named $V_1, \ldots, V_{l-1}$. Further, pick a sample from $\mathcal{V}_0$, call it H. Compute

$$U = H + \sum_{i=1}^{d_v - 1} V_i \,.$$

Note, these are exactly the message-passing rules at a variable node.

Think now of each set $\mathcal{V}_m$ and $\mathcal{C}_m$ as a sample of the corresponding distribution. E.g., in order to construct this distribution approximately we might use a histogram applied to the set. Recall, that we assume here the all-zero codeword assumption. Hence, in order to see whether this experiment corresponds to a successful decoding, we need to check whether in $\mathcal{V}_m$ all samples have positive sign and magnitude which converges to infinity as m increases.

Implement the population dynamics approach for transmission over the BAWGN(σ) channel using the $(3, 6)$-regular Gallager ensemble. Estimate the threshold using this method. Plot the threshold on the same plot as the simulation results of problem 6.6. Hopefully this vertical line, indicating the threshold, is somewhere around where the error probability curves show a sharp drop-off.

6.8 Gallager Algorithm A. One of the downsides of BP in a practical application is that it requires the exchange of real numbers. Hence, in any implementation messages are quantized to a fixed number of bits. One way to think of such a quantized algorithm is that the message represents an "approximation" of the underlying message that BP would have sent. Assume that we are limited to exchange messages consisting of a single bit. Recall that for BP a positive message means that our current estimate of the associated bit is $+1$, whereas a negative message means that our current estimate is -1 (the magnitude of the BP message conveys our certainty). So we can think of a message-passing algorithm which is limited to exchange messages consisting of a single bit, as exchanging only the sign of their estimate.

The best known such algorithm (and historically also the oldest) is Gallager's "algorithm A." We assume that the codewords and the received word have components in $\{0, 1\}$ (so think of transmission on the BSC). The message passing rules are:

(i) *Initialization:* In the first iteration send out the received bits along all edges incident to a variable node.

(ii) *Check Node Rule:* At a check node send out along edge e the XOR of the incoming messages (not counting the incoming message along edge e).

(iii) *Variable Node Rule:* At a variable node send out the received value along edge e unless all incoming messages (not counting the incoming message on edge e) all agree in their value. Then send this value.

Assume that transmission takes place over the BSC(ϵ) and that we are using

a $(3, 6)$-regular Gallager ensemble. Write down the density evolution equations for the Gallager algorithm A.

7 Interlude: Message Passing for the Sherrington-Kirkpatrick Spin Glass

This Chapter applies message passing methods to the Sherrington-Kirkpatrick (SK) model of a spin glass. The SK model is a very particular random spin system defined on a complete graph with pair interactions of random i.i.d strengths for each edge (see Section 2.6). The impatient reader can very well jump ahead directly to the next chapter on compressive sensing, but there are good reasons for the present interlude. Certainly, the conceptual and historical role of the SK model in our theoretical understanding of random spin systems cannot be underestimated, however, for us the message passing analysis of this model will serve as a stepping stone towards the technically more involved but related message passing analysis in compressive sensing. In the present chapter we explore message passing within the SK model and will then apply in Chapter 8 what we have learned to compressive sensing.

The application of message passing is similar to coding in its general initial outline. However there is a difference: the SK and compressed sensing models are defined on complete graphs (the graph for compressed sensing is bipartite complete). This is as far as one can get from locally tree-like graphs, so one might think that message passing simply should not work very well for such models and that this should be the end of the story. But in fact the story is much more convoluted and interesting. Belief propagation works well for compressed sensing and it also works for the SK model in its high temperature phase. For the SK model at low temperatures simple message passing does not correctly take into account "long range correlations" and one has to resort to a more sophisticated version of the theory. This new level of sophistication is not really needed for compressed sensing, but rather for the satisfiability problem, so we postpone it to part III.

Because of the denseness of the graph, message passing algorithms a priori involve $\Theta(n^2)$ messages flowing on edges at each iteration step. From the point of view of complexity this is not very good. Recall in coding for sparse graphs the number of message updates at each iteration is $\Theta(n)$. However, as we will see, the denseness of the graph in fact allows to simplify the BP equations and bring down this complexity to linear order. In the SK model the simplified equations one ends up with, are an iterative form of the celebrated Thouless-Anderson-Palmer (TAP) equations. Thouless, Anderson and Palmer initially derived their equations through the analysis of a high temeprature expansion of the free energy

and showed that the Curie-Weiss equation has to be corrected by a so-called "Onsager reaction" term. Here we will discover that the Onsager reaction term just appears automatically within the BP formalism.

An analog of density evolution can be derived from the (iterative) TAP equations. For historical reasons briefly explained in the Notes this goes under the strange name of *replica symmetric* equations. The replica symmetric fixed point equation predicts a threshold behaviour and it is natural to ask what is the relation of this threshold with a thermodynamic phase transition threshold. The necessary tools to answer this difficult question will only be developed in part III. We review in Section 7.5 the main aspects of the exact solution and phase transition of the SK model.

7.1 Sherrington-Kirkpatrick model and belief propagation approach

Sherrington-Kirkpatrick model

The Sherrington-Kirkpatrick model of a spin glass was briefly introduced in the examples of Section 2.6. Recall that the model is defined on a complete graph with n vertices, has binary spin degrees of freedom $s_i = \pm 1$, $i = 1, \dots, n$ attached to the vertices. The Hamiltonian is

$$\mathcal{H}(\underline{s}) = -\sum_{1 \le i < j \le n} J_{ij} s_i s_j - h \sum_{i=1}^{n} s_i, \tag{7.1}$$

where h is a constant magnetic field and J_{ij} are $n(n-1)/2$ i.i.d random variables (the "coupling constants") associated to the edges of the complete graph. In popular versions of the model one chooses $J_{ij} \sim \mathcal{N}(0, J^2/n)$ or $J_{ij} = \pm J/\sqrt{n}$ with i.i.d Bernoulli(1/2) signs and $J > 0$ a constant.

Why are the coupling constants scaled by $1/\sqrt{n}$? That this is the right scaling can be seen by looking at the fluctuations of the Hamiltonian. The mean and variance of $\mathcal{H}(\underline{s})$ are respectively equal to $-h \sum_{i=1}^{n} s_i$ and $(n-1)J^2/2$. Thus for general spin assignements the energy has a standard deviation of $O(\sqrt{n})$ around a mean $O(n)$ and we expect the thermodynamic limit to make sense and be non-trivial. Later on it will often be useful to explicitely extract the scaling by setting $J_{ij} = \tilde{J}_{ij}/\sqrt{n}$ where $\tilde{J}_{ij} \sim \mathcal{N}(0, J^2)$ or $\tilde{J}_{ij} = \pm J$.

The corresponding Gibbs distribution $e^{-\beta \mathcal{H}(\underline{s})}/Z$ is itself random. As is usual for random Gibbs distributions, there are two levels of randomness. The first one associated to quenched or frozen variables, here the coupling constants J_{ij}, and the second one corresponding to the spin assignments distributed according to the Gibbs distribution. We refer back to Chapter 2 for a more extensive discussion of these two levels of randomness.

One of the major achievements of the theory of random spin system is the derivation of an exact formula for the average free energy of the SK model, namely $-\lim_{n \to +\infty} \beta^{-1} \mathbb{E}[\ln Z]/n$, as well as a proof of the concentration of

Add figure of a complete graph and the factor graph with factors attached. Picture with 4 vertices and 6 edges.

Figure 7.1 Complete graph of the SK model and factor graph representation of the Gibbs measure.

$(\ln Z)/n$ as $n \to \infty$. It is perhaps a good idea to stress that the similarity of (7.1) with the Curie-Weiss Hamiltonian should not give the false impression that the path to the solution is easy. Embarking into it at the present stage would distract us too much from our present goal, which is, as explained in the introduction, to concentrate on the message passing approach. A brief comparison of the message passing predictions with the exact statistical mechanics solution is found in Section 7.5.

Belief propagation equations

We now look at BP equations for the SK model. It will shortly become clear that these equations are in fact the same for any Ising model with pairwise interactions (as defined in Sect. 2.1). The specificities related to the SK model are really used only in the next section.

To proceed systematically with the formalism of Chapter 5, we first set up the factor graph formulation (see Figure 7.1). The vertices $i = 1, \dots, n$ of the original (complete) graph play the role of variable nodes. On every edge $(i, j) \equiv a$ we place a factor node with factor $f_a(s_i, s_j) = e^{\beta J_{ij} s_i s_j}$. We then attach extra degree-one factor nodes $\hat{i}$ to each variable node i. The factor associated to $\hat{i}$ is $f_i(s_i) = e^{\beta h s_i}$.

Further, we let $\hat{\mu}_{a\to i}(s_i)$ denote the message which flows from the factor node a to the variable node i. In a similar manner, $\mu_{i\to a}(s_i)$ is the message flowing from variable node i to factor node a. There is also a "trivial" message $\mu_{\hat{i}\to i}(s_i) = f_i(s_i) = e^{\beta h s_i}$ flowing from degree-one factor nodes to variable nodes. Since all messages depend on binary variables $s_i = \pm 1$ we can use the same type of parametrization used for coding in Chapter 6 and set,

$$\hat{h}_{a\to i} = \frac{1}{2\beta} \ln\left\{\frac{\hat{\mu}_{a\to i}(+1)}{\hat{\mu}_{a\to i}(-1)}\right\}, \qquad h_{i\to a} = \frac{1}{2\beta} \ln\left\{\frac{\mu_{i\to a}(+1)}{\mu_{i\to a}(-1)}\right\}. \tag{7.2}$$

Up to the factor β^{-1} these are the usual loglikelihood variables associated to the messages. In the context of spin systems they are also called *cavity fields*. The reason comes from their physical interpretation which will shortly become clear (this interpretation is also the reason why we prefer here the letter "h" instead of "l'' used in Chapter 6).

Add figure to show elimination of degree two factors

Figure 7.2 The factor nodes of degree two can be eliminated and one can work with only one set of messages $h_{i\to j}$ flowing on the original complete graph.

The general BP equations (5.10) read

$$\begin{cases} \mu_{i\to a}(s_i) & = e^{\beta h s_i} \prod_{b\in\partial i\setminus a} \hat{\mu}_{b\to i}(s_i) \\ \hat{\mu}_{a\to i}(s_i) & = \sum_{\sim s_i} e^{\beta J_{ij} s_i s_j} \prod_{j\in\partial a\setminus i} \mu_{j\to a}(s_j) \end{cases} \tag{7.3}$$

An exercise shows that parametrization (7.2) leads to

$$\begin{cases} h_{j\to a} & = h + \sum_{b\in\partial j\setminus a} \hat{h}_{b\to j}, \\ \hat{h}_{a\to j} & = \frac{1}{\beta} \operatorname{atanh}\{\tanh(\beta J_{ij}) \tanh(\beta h_{i\to a})\}. \end{cases} \tag{7.4}$$

Note the similarity with (6.10) in coding theory. Equ. (7.4) reduce to such "coding-like" equations by setting $\beta = 1$ and letting $J_{ij} \to +\infty$ in which case the factor nodes correspond to degree two "parity checks".

There is a special feature of systems with degree two factors that we have not encounterd yet explicitely. The two equations in (7.4) can be conveniently reduced to a single one with messages flowing on the *original* graph. To see this note that because a factor b has degree two, a directed edge $b \to j$ can be identified with a directed edge $i \to j$ on the original graph where i is the unique vertex in $\partial b \setminus j$ (see Figure 7.2). In other words, setting $h_{i\to j} = \hat{h}_{b\to j}$, Equations (7.4) become

$$h_{i\to j} = \frac{1}{\beta} \operatorname{atanh}\big\{(\tanh(\beta J_{ij}) \tanh(\beta(h + \sum_{k\in\partial i\setminus j} h_{k\to i}))\big\}. \tag{7.5}$$

This message passing equation does not refer anymore to the factor graph. Messages flow on the original graph.

The BP-marginal, $\nu_i^{\mathrm{BP}}(s_i)$, at vertex i is determined from the sum of the external and all cavity fields,

$$h + \sum_{a\in\partial i} \hat{h}_{a\to i}, \qquad \text{or equivalently} \qquad h + \sum_{k\in\partial i} h_{k\to i}. \tag{7.6}$$

Explicitly, the normalized marginal is

$$\nu_i^{\mathrm{BP}}(s_i) = \frac{e^{\beta(h+\sum_{k\in\partial i} h_{k\to i}) s_i}}{2\cosh(\beta(h + \sum_{k\in\partial i} h_{k\to i}))}. \tag{7.7}$$

The BP estimate for the magnetization, is by definition the average spin com-

Figure 7.3 A complete graph and the graph with a spin removed and the leftover cavity.

puted from the BP-marginal

$$m_i^{\mathrm{BP}} = \sum_{s_i \in \{\pm 1\}} s_i \nu_i^{\mathrm{BP}}(s_i) = \tanh(\beta(h + \sum_{k \in \partial i} h_{k \to i})). \tag{7.8}$$

We will call m_i^{BP} the BP-magnetization to distinguish it from the (true) thermal equilibrium magnetization $m_i = \langle s_i \rangle$.

Let us pause to give a physical interpretation of these formulas. A single spin s in the presence of a magnetic field h has a Hamiltonian $\mathcal{H}(s) = -hs$ and thus a magnetization $\langle s \rangle = \tanh(\beta h)$. Therefore one interprets $h + \sum_{k \in \partial i} h_{k \to i}$ as an effective magnetic field felt by spin s_i. This is often called the *local field* or also the *mean field*. The local field is the sum of the external field h and the total *cavity field* $h_{i,\mathrm{cav}} \equiv \sum_{k \in \partial i} h_{k \to i}$. The later is an effective field produced by the rest of the system in a "cavity" left out when one removes vertex i from the graph. Such a cavity is illustrated on Figure 7.3. This explains why the messages $h_{k \to i}$, $h_{i \to a}$, $\widehat{h}_{a \to i}$ are generically called "cavity fields".

Flooding schedule

From the perspective of traditional statistical mechanics one would view the BP equations as fixed point equations and try to find all solutions. When multiple solutions arise the important question is: which one to choose? Such issues are deferred to Part III.

Here we take the *algorithmic standpoint*. Recall from Chapter 5, when the underlying graph is a tree the initial conditions and iterations are clearly determined. This is also the situation where the BP equations have a unique solution found by these iterations. But when the graph is not a tree we have to specify initial conditions and a schedule to solve the equations iteratively. Just as in coding we adopt the flooding schedule. A natural initialization is given by the "prior" that we have about the local field. We therefore set

$$h_{i \to j}^t = \frac{1}{\beta} \operatorname{atanh}\big\{(\tanh(\beta J_{ij}) \tanh(\beta(h + \sum_{k \in \partial i \setminus j} h_{k \to i}^{t-1}))\big\}, \quad h_{i \to j}^0 = 0. \tag{7.9}$$

The BP-estimate of the magnetization at time t is,

$$m_i^t = \tanh\big\{\beta(h + \sum_{j \in \partial i} h_{j \to i}^t)\big\}. \tag{7.10}$$

Convergence of the iteration is not guaranteed in general. And even if iterations converge the solution might not be unique, and its relation to the exact statistical mechanical free energy is not obvious.

What is the complexity of this schedule on a complete graph? At each time step t a node i receives $n-1$ messages from which one computes first $h^{t-1}_{i,\,\mathrm{cav}} \equiv \sum_{k\in\partial i} h^{t-1}_{k\to i}$ which counts as $n-1$ additions. Then one computes the $n-1$ outgoing messages as follows

$$h^t_{i\to j} = \beta^{-1}\,\mathrm{atanh}\big\{(\tanh(\beta J_{ij})\tanh(\beta(h + h^{t-1}_{i,\,\mathrm{cav}} - h^{t-1}_{j\to i})\big\},$$

which counts as $n-1$ extra operations (assuming here the operation for each j has unit cost). Since there are n nodes in total this makes $2(n-1)n$ operations. Thus the total complexity is equal to $\Theta(n^2)$ times the number of iterations. In the next section we show that suitable approximations allow to reduce the complexity by one order.

7.2 From belief propagation to Thouless-Anderson-Palmer equations

As just noted above, because the graph is complete, a single BP iteration has quadratic complexity which is costly. Fortunately one can simplify the BP equations and bring the complexity down to order $\Theta(n)$. The key to the simplification is that the coupling constants are weak. Indeed, recall that we have $J_{ij} = \tilde{J}_{ij}/\sqrt{n}$ with fluctuations of $\tilde{J}_{ij} = O(1)$, so we assume in general that the coupling constants J_{ij} are small when $n \to +\infty$, and perform an expansion of the message passing equations. This has to be done with care however and typically one must go beyond the lowest order term in order to obtain correct results. Interestingly, these simplifications of message-passing equations lead to an iterative form of the Thouless-Anderson-Palmer (TAP) equations. These equations have a complexity of $\Theta(n)$ at each iteration. Thus they provide a linear complexity algorithm to compute an algorithmic "TAP-estimate" of the magnetization.

Consider the BP iteration (7.9) at step t. Using the local field

$$\eta_i \equiv h + h_{i,\mathrm{cav}} = h + \sum_{k\in\partial i} h_{k\to i} \tag{7.11}$$

we can rewrite this iteration as

$$h^t_{i\to j} = \frac{1}{\beta}\,\mathrm{atanh}\Big\{\tanh(\beta J_{ij})\tanh(\beta\eta^{t-1}_i - \beta h^{t-1}_{j\to i})\Big\}.$$

Now, since J_{ij} is of order $1/\sqrt{n}$ we Taylor expand the hyperbolic tangent and its inverse. This yields

$$h^t_{i\to j} = J_{ij}\tanh(\beta\eta^{t-1}_i - \beta h^{t-1}_{j\to i}) + O(\beta^2 J^3_{ij}). \tag{7.12}$$

This equation shows that each cavity field is $O(J_{ij})$. On the other hand (7.11)

shows that η_i^{t-1} involves the sum of $n-1$ such cavity fields. Therefore we expect $h_{j\to i}^{t-1}$ to be much smaller than η_i^{t-1}, and we further expand the hyperbolic tangent in (7.12) in powers of the cavity field

$$h_{i\to j}^{t} = J_{ij}\tanh(\beta\eta_i^{t-1}) - \beta J_{ij}h_{j\to i}^{t-1}\Big(1-(\tanh(\beta\eta_i^{t-1}))^2\Big) + O(\beta^2 J_{ij}^3)\,. \quad (7.13)$$

Thanks to (7.10) we can rewrite this equation as,

$$h_{i\to j}^{t} = J_{ij}m_i^{t-1} - \beta J_{ij}h_{j\to i}^{t-1}\big(1-(m_i^{t-1})^2\big) + O(\beta^2 J_{ij}^3)\,. \quad (7.14)$$

Now we seek to express $h_{j\to i}^{t-1}$ on the right hand side of this equation in terms of magnetizations. This will allow to approximate cavity fields entirely in terms of the magnetizations. We note that if we interchange the roles of i and j in (7.14) and use $h_{j\to i}^{t-1} = O(J_{ji})$, we get (since $J_{ij} = J_{ji}$)

$$h_{j\to i}^{t-1} = J_{ij}m_j^{t-2} + O(\beta J_{ij}^2). \quad (7.15)$$

Replacing (7.15) in (7.14) we obtain

$$h_{i\to j}^{t} = J_{ij}m_i^{t-1} - \beta J_{ij}^2 m_j^{t-2}\big(1-(m_i^{t-1})^2\big) + O(\beta^2 J_{ij}^3). \quad (7.16)$$

The first two terms on the right hand side are $O(n^{-1/2})$ and $O(n^{-1})$ while the error term is $O(n^{-3/2})$. Dropping this error term[1] and replacing in in (7.10), we arrive at

$$m_j^t = \tanh\Big\{\beta\Big(h + \sum_{i\in\partial j} J_{ij}m_i^{t-1} - \beta m_j^{t-2}\sum_{i\in\partial j} J_{ij}^2\big(1-(m_i^{t-1})^2\big)\Big)\Big\}\,. \quad (7.17)$$

In the statistical mechanics literature the TAP equations correspond to the fixed point form of (7.17). Their original derivation was obtained by very different means involving expansion methods to compute the free energy.

Discussion of the TAP equations

With the scaling of the coupling constant made explicit the iterative TAP equations are

$$m_j^t = \tanh\Big\{\beta\Big(h + \frac{1}{\sqrt{n}}\sum_{i=1,i\neq j}^{n}\tilde{J}_{ij}m_i^{t-1} - \frac{\beta}{n}m_j^{t-2}\sum_{i=1,i\neq j}^{n}\tilde{J}_{ij}^2\big(1-(m_i^{t-1})^2\big)\Big)\Big\}. \quad (7.18)$$

[1] One may rightly object that dropping $O(\beta^3 J_{ij}^3)$ terms is not harmless because at each iteration these errors accumulate. This difficulty can be ignored if one's goal is to develop the simplest possible *algorithm* to estimate the magnetization. For example one might loose in precision with respect to BP but lower the complexity. Here, it turns out that one lowers the complexity and does not loose precision with respect to BP. The algorithmic estimate so obtained is in a certain sense optimal in a high temperature region of the phase diagram (see Section 7.5).

It is worth pointing out that the TAP equations take their simplest form when $\tilde{J}_{ij} \sim \pm J$ with Ber(1/2) signs. Indeed $\tilde{J}_{ij}^2 = J^2$ so and setting

$$q_{\mathrm{EA}}^{t-1} \equiv \frac{1}{n}\sum_{i=1}^{n}(m_i^{t-1})^2$$

we get

$$m_j^t = \tanh\left\{\beta\left(h + \frac{1}{\sqrt{n}}\sum_{i=1}^{n}\tilde{J}_{ij}m_i^{t-1} - \beta J^2 m_j^{t-2}(1-q_{\mathrm{EA}}^{t-1})\right)\right\}.$$

The parameter q_{EA}^t is an "algorithmic version" of the so-called Edwards-Anderson paremeter $q_{\mathrm{EA}} = \frac{1}{n}\sum_{i=1}^{n}\langle si\rangle^2$ which involves the equilibrium magnetizations, and plays an important role in the exact solution of the model.

Only estimates of the magnetization are involved and there are no messages flowing on edges anymore. At each iteration step $t \geq 1$, magnetization estimates at vertices of the graph are updated and there are n such updates, so the complexity is now $\Theta(n)$ times the number of iterations. An aspect of the iterations which turns out to play a crucial role in numerical implementations is the organisation of the time indices. From a purely algorithmic nothing prevents from trying other arrangements which might work as well or even better in practice. With the approach taken here which starts from the more primary BP equations the arrangement of the time indices comes for free in a principled way. This is a welcome aspect of the approach when one deals with technically more complicated but similar problems such as compressive sensing (see Chapter **??**.

Let us now discuss the issue of initial conditions. Recall that within the BP approach we set $h_{j\to i}^{t=0} = 0$ or equivalently $m_i^{t=0} = \tanh(\beta h)$ (see (7.9), (7.10)). The TAP equations are "second order equations" and require two initial conditions. A look at (7.18) shows that a consistent choice is $m_i^{t=-2} = m_i^{t=-1} = 0$.

The local field in (7.18) is given by the external field h plus a cavity field

$$\begin{aligned} h_{j,\mathrm{cav}}^t &= \frac{1}{\sqrt{n}}\sum_{i=1}^{n}\tilde{J}_{ij}m_i^{t-1} - \frac{\beta}{n}m_j^{t-2}\sum_{i=1}^{n}\tilde{J}_{ij}^2(1-(m_i^{t-1})^2 \\ &= \frac{1}{\sqrt{n}}\sum_{i=1}^{n}\tilde{J}_{ij}m_i^{t-1} - \beta J^2 m_j^{t-2}(1-q_{\mathrm{EA}}^{t-1}) \end{aligned}$$

which is an approximation of the BP cavity field discussed in Sect. 7.1. This is the field created by all spins $i \neq j$ in a cavity left over by removing the spin at j. Each contribution has an interpretation. The first term is the usual Curie-Weiss mean field. But this mean field includes - via the terms m_i - an influence of the spin at j on itself and this "back reaction" should be should be subtracted. This is exactly what the second term $-\beta J^2 m_j(1-q_{\mathrm{EA}})$ does. This is called an *Onsager reaction* term. A back of the envelope heuristic derivation of the Onsager reaction is found in the exercises.

Add a small guided exercise

A parenthesis: the CW model revisited

Recall that the exact solution of the CW model in Chapter 4 led us to the fixed point equation $m = \tanh(\beta(h + Jm))$ Here the local field is just the sum of the external field h and the CW mean field Jm. Why is it that the Onsager reaction term is not needed here? One can repeat the same theory developed in this chapter for (non random) coupling constants $J_{ij} = J/n$. Starting from BP equations and then approximating them to leading orders in coupling constants we obviously find again (7.17). At this point, setting $J_{ij} = J/n$ one easily sees that the Onsager term is $O(n^{-1})$ and can be neglected, so that only the usual CW mean field remains. We are lead to the iterative equations

$$m_j^t = \tanh(\beta(h + \frac{J}{n}\sum_{i=1}^{n} m_i^{t-1}))\,. \tag{7.19}$$

Since the right hand side does not depend on j we can seek a uniform solution $m_j^t = m^t$. Equation (7.19) becomes

$$m^t = \tanh(\beta(h + Jm^{t-1})). \tag{7.20}$$

To summarize, for the CW model the TAP equation reduces to the CW equation because the Onsager reaction term is negligible.

This remark teaches us an important lesson. Recall that the exact solution for the magnetization is found by selecting the fixed point of the CW equation (4.22) which minimises the free energy. The BP approach leads to the iterative form (7.20) which we should solve with the initial condition $m^0 = 0$. Whether the estimate m^t converges to the true magnetisation (computed in Chapter 4) depends on the region of the phase diagram in the (β, h) plane. An analysis of the iterations shows that this is the case outside of the spinodal region, i.e., for $\beta J < 1$ and $(\beta J > 1, |h| > h_{\rm sp})$. Such intimate connections between algorithmic and thermodynamic solutions will be discussed in more depth in Part III, so we close this parenthesis here.

7.3 Replica symmetric equations

The goal of density evolution in coding is to write down an iterative equation that tracks the average behaviour of the algorithm. This can also be done for the TAP iterations under a non-trivial *assumption* of weak correlation for the cavity fields.

Recall expression (7.10) for the BP-magnetization where we set for convenience $h_{j\to i} = f_{j\to i}/\sqrt{n}$

$$m_i^t = \tanh\big\{\beta(h + \frac{1}{\sqrt{n}}\sum_{j\in\partial i} f_{j\to i}^t)\big\} \tag{7.21}$$

where

$$f^t_{j\to i} \approx \tilde{J}_{ij} m^{t-1}_j - \frac{\beta}{\sqrt{n}} m^{t-2}_i \tilde{J}^2_{ij}(1-(m^{t-1}_j)^2)\,. \tag{7.22}$$

within the TAP approximation. A highly non trivial result states that there is a high temperature portion of the (h,T) plane where these messages are only weakly correlated and the central limit theorem applies

$$\lim_{n\to+\infty} \frac{1}{\sqrt{n}} \sum_{j\in\partial i} f^t_{j\to i} \sim \mathcal{N}(0, q_{t-1}). \tag{7.23}$$

where $q_t \equiv \mathbb{E}[(m^t_j)^2]$ (by symmetry this expectation is independent of j).

In the case of coding, as shown in Chapter 6, the assumption of weak correlation of messages is justified in a regime $t \ll n$ because of the locally tree like nature of the factor graph. In fact there, we could condition on the high probability event that a computational tree of finite depth really is a tree, so that messages are independent. For the SK model on a complete graph we cannot rely on the same method to justify the assumption of weak correlations. It turns out that this assumption is true in a "high temperature" portion of the phase diagram, but fails for low temperatures. The Onsager term plays a crucial role: if the CW contribution $\tilde{J}_{ij} m_j$ alone is retained the central limit theorem *cannot* be applied for whatever values of temperature and magnetic field. In effect the Onsager term is responsible for decorrelating the CW terms. A rigorous proof of these important results was given only relatively recently by Bolthauzen and is beyond our scope here. The reader can find evidence for the "miraculous" role of the Onsager term by analysing in detail the first few iterations in Section 7.4. Clear numerical evidence can also be found in a guided exercise.

When (7.23) is satisfied we are in a position to write down "evolution equations" for the average behaviour of the TAP iterations. Set

$$m_t = \mathbb{E}[m^t_i], \qquad \text{and} \qquad q_t = \mathbb{E}[(m^t_i)^2] \tag{7.24}$$

Thanks to (7.23) we can take the expectation of Equ. (7.21) and of the squared version of this equation. This yields

$$\begin{cases} m_t = \int_{-\infty}^{+\infty} du\, \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \tanh\{\beta(h+u\sqrt{q_{t-1}})\}, \\ q_t = \int_{-\infty}^{+\infty} du\, \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \tanh^2\{\beta(h+u\sqrt{q_{t-1}})\}. \end{cases} \tag{7.25}$$

The initial condition (consistent with $m^0_i = \tanh(\beta h)$) is $q_0 = 0$.

The fixed point form of the iterations (7.25) are called the *replica symmetric equations*. This strange name comes from the original derivations, in the statistical mechanics literature, which involved completely different methods (see the notes).

The solutions of the fixed point equations display an interesting threshold behaviour when the temperature is varied on the $h = 0$ axis. Since tanh is an

odd function, the fixed point version of (7.25) becomes

$$m = 0, \quad q = \int_{-\infty}^{+\infty} du \, \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \tanh^2(\beta u \sqrt{q}). \tag{7.26}$$

This equation admits a trivial fixed point $q = 0$ for all β, which is unique and stable for $\beta < 1$. For $\beta > 1$ this fixed point is unstable, and a second stable non-trivial fixed point $q \neq 0$ appears. This is similar to the situation we encountered with the CW fixed point equation and therefore suggests that a phase transition occurs in this model at $(\beta = 1, h = 0)$ when we move along the $h = 0$ axis. This conclusion is correct, but it is only the tip of the iceberg and the phase transition in the SK model turns out to be a very subtle one. For one thing, (7.26) has a unique smoth solution for $h \neq 0$ and therefore does not predict a phase transition for non-zero magnetic fields. It turns out that this is a wrong prediction. The correct picture of the phase diagram in the (β, h) plane is briefly reviewed in Section (7.5).

7.4 First iterations of the TAP equations[2]

We briefly outline how the mean and covariance of the messages $f^t_{j\to i}$ can be computed for the first iterations. We consider the Gaussian model $\tilde{J}_{ij} \sim \mathcal{N}(0, 1)$ which allows to carry out explicit calculations showing that messages

$$f^{(2)}_{j\to i} = \tilde{J}_{ij} m^{(1)}_j - \frac{\beta}{\sqrt{n}} m^{(0)}_i \tilde{J}^2_{ij} (1 - (m^{(1)}_j)^2)$$

have mean $\mathbb{E}[f^{(2)}_{j\to i}] = O(n^{-2})$ and covariance $\mathbb{E}[f^{(2)}_{k\to i} f^{(2)}_{l\to i}] = \delta_{kl} q_1$. This is consistent with (7.23). The algebra clearly shows the crucial role of the Onsager term already at the second iteration $t = 2$. We leave it as an exercise for the reader to carry out similar (but more lengthy) calculations for $t = 3$.

The case $t = 1$ is trivial. With the initialization $m_i^{t=-2} = m_i^{t=-1} = 0$ we have $m^{(0)}_i = \tanh(\beta h)$. Thus (7.22) immediately implies that $f^{(1)}_{j\to i} = \tilde{J}_{ij} \tanh(\beta h)$ are i.i.d Gaussian with $\mathbb{E}[f^{(1)}_{j\to i}] = 0$ and $\mathbb{E}[(f^{(1)}_{j\to i})^2] = (\tanh \beta h)^2 = q_0$.

Let us now consider the second iteration $t = 2$. Now $m^{(1)}_i = \tanh(\beta h + \frac{\beta}{\sqrt{n}} \sum_{j=1, j\neq i}^n \tilde{J}_{ij} m^{(0)}_j)$ is also involved. For the mean we have

$$\mathbb{E}[f^{(2)}_{j\to i}] = \mathbb{E}[\tilde{J}_{ij} m^{(1)}_j] - \frac{\beta}{\sqrt{n}} m^{(0)}_i \mathbb{E}[\tilde{J}^2_{ij} (1 - (m^{(1)}_j)^2)] \tag{7.27}$$

The main trick to evaluate such expressions is to use the integration by parts

[2] This section is not needed for the main development and can be skipped in a first reading.

formula[3]

$$\int dx\, \frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}}\, x f(x) \frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}} = \int dx\, \frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}}\, f'(x)$$

which yields for the first term in (7.27)

$$\begin{aligned}\mathbb{E}[\tilde{J}_{ij} m_j^{(1)}] &= \mathbb{E}[\frac{\partial}{\partial \tilde{J}_{ij}} m_j^{(1)}] = \mathbb{E}[(1-(m_j^{(1)})^2)\frac{\beta}{\sqrt{n}} m_i^{(0)}] \\ &= \frac{\beta}{\sqrt{n}} m_i^{(0)} \mathbb{E}[(1-(m_j^{(1)})^2)]\end{aligned}$$

We see that the result is almost equal to the Onsager term in (7.27). In fact, integrating by parts in that term we get

$$\begin{aligned}\mathbb{E}[\tilde{J}_{ij}^2 (1-(m_j^{(1)})^2)] &= \mathbb{E}[(1-(m_j^{(1)})^2)] + \mathbb{E}[\tilde{J}_{ij} \frac{\partial}{\tilde{J}_{ij}} (1-(m_j^{(1)})^2)] \\ &= \mathbb{E}[(1-(m_j^{(1)})^2)] - \frac{2\beta}{\sqrt{n}} m_i^{(0)} \mathbb{E}[\tilde{J}_{ij} m_j^{(1)} (1-(m_j^{(1)})^2)] \\ &= \mathbb{E}[(1-(m_j^{(1)})^2)] - \frac{2\beta^2}{n} (m_i^{(0)})^2 \mathbb{E}[(1-(m_j^{(1)})^2)^2] \\ &+ \frac{4\beta^2}{n} (m_i^{(0)})^2 \mathbb{E}[(m_j^{(1)})^2 (1-(m_j^{(1)})^2]\end{aligned}$$

A useful rule of thumb to bypass this last calculation is to make the replacement $J_{ij}^2 \to +1$ as if the couplings where Bernoulli random variables, and add a term $O(n^{-1})$ to account for the error. With these results we see that (7.27) becomes

$$\mathbb{E}[f_{j\to i}^{(2)}] = O(n^{-3/2})\,.$$

In particular

$$\mathbb{E}[\frac{1}{\sqrt{n}} \sum_{j\in\partial i} f_{j\to i}^{(2)}] = O(n^{-1})\,.$$

We now turn to the covariance of messages $f_{k\to i}^{(2)}$ and $f_{l\to i}^{(2)}$, and show that asymptotically it equals $\delta_{kl} q_1$. From (7.22) we get four contributions

$$\begin{aligned}\mathbb{E}[f_{k\to i}^{(2)} f_{l\to i}^{(2)}] &= \mathbb{E}[\tilde{J}_{ik}\tilde{J}_{il} m_k^{(1)} m_l^{(1)}] \\ &- \frac{\beta}{\sqrt{n}} m_i^{(0)} \mathbb{E}[\tilde{J}_{ik} \tilde{J}_{il}^2 m_k^{(1)} (1-(m_l^{(1)})^2)] \\ &- \frac{\beta}{\sqrt{n}} m_i^{(0)} \mathbb{E}[\tilde{J}_{il} \tilde{J}_{ik}^2 m_l^{(1)} (1-(m_k^{(1)})^2)] \\ &+ \frac{\beta^2}{n} (m_i^{(0)})^2 \mathbb{E}[\tilde{J}_{ik}^2 \tilde{J}_{il}^2 (1-(m_k^{(1)})^2)(1-(m_l^{(1)})^2)]\,.\end{aligned}$$

[3] For the Bernoulli or other distributions this simple formula cannot be used directly. But with some extra work a similar analysis is also possible.

For $k \neq l$, an integration by parts leads to

$$\begin{aligned}\mathbb{E}[f^{(2)}_{k\to i} f^{(2)}_{l\to i}] = &\frac{\beta^2}{n}(m^{(0)}_i)^2\mathbb{E}[(1-(m^{(1)}_k)^2(1-(m^{(1)}_l)^2] \\ &- \frac{\beta^2}{n}(m^{(0)}_i)^2\mathbb{E}[\tilde{J}^2_{il}(1-(m^{(1)}_k)^2)(1-(m^{(1)}_l)^2)] \\ &- \frac{\beta^2}{n}(m^{(0)}_i)^2\mathbb{E}[\tilde{J}^2_{ik}(1-(m^{(1)}_l)^2)(1-(m^{(1)}_k)^2)] \\ &+ \frac{\beta^2}{n}(m^{(0)}_i)^2\mathbb{E}[\tilde{J}^2_{ik}\tilde{J}^2_{il}(1-(m^{(1)}_k)^2)(1-(m^{(1)}_l)^2)]\,.\end{aligned}$$

Using the rule of thumb alluded to above we find that $\mathbb{E}[f^{(2)}_{k\to i} f^{(2)}_{l\to i}] = O(n^{-2})$ for $k \neq l$. For $k = l$ we have

$$\begin{aligned}\mathbb{E}[(f^{(2)}_{k\to i})^2] &= \mathbb{E}[\tilde{J}^2_{ik}(m^{(1)}_k)^2] - 2\frac{\beta}{\sqrt{n}}m^{(0)}_i\mathbb{E}[\tilde{J}^3_{ik}m^{(1)}_k(1-(m^{(1)}_k)^2)] \\ &\quad + \frac{\beta^2}{n}(m^{(0)}_i)^2\mathbb{E}[\tilde{J}^4_{ik}(1-(m^{(1)}_k)^2)^2] \\ &= \mathbb{E}[(m^{(1)}_k)^2] + O(n^{-1/2})\end{aligned}$$

where the last equality is again obtained by integration by parts. With these results we can conclude

$$\mathbb{E}[(\frac{1}{\sqrt{n}}\sum_{j\in\partial i} f_{j\to i})^2] = q_1 + O(\frac{1}{\sqrt{n}})$$

consistently with (7.23).

7.5 Exact solution of the SK model[4]

In Chapter 4 we solved exactly the CW model and learned that the free energy could be expressed in variational form (4.11). This is also true for the SK model, however the variational expression and its derivation are considerably more subtle. The correct solution was first provided by Parisi using a purely algebraic method called the *replica method*. Since then, the solution has been rederived in a more probabilistic way which goes under the name *cavity method*. The cavity method which will be discussed in Part III can be seen as an "upgraded" message passing method that takes into account long-range correlations that BP neglects.

Here we briefly review the Parisi formula for exact average free energy. This formula was proved much later by Talagrand. From this formula one can infer the existence of a high temperature phase where the replica symmetric fixed point equations are exact, and a low temperature phase where they are not. The phase transition line separating these two phases in the (h, T) plane is called the Almeida-Thouless (AT) line.

Let $x : q \in [0,1] \to x(q) \in [0,1]$ be a non-decreasing cumulative distribution

[4] This section is not needed for the main development and can be skipped in a first reading.

function. Call $\mathcal{Q}$ the space of such cumulative distribution functions. Define the "Parisi functional"

$$f_{\mathrm{P}}(x]) \equiv -\ln 2 - f(0,h;x) - \frac{\beta^2}{2}\int_0^1 q\,x(q)dq \tag{7.28}$$

where $f(q,h;x)$ satisfies the partial differential equation

$$\frac{\partial f}{\partial q} + \frac{1}{2}\frac{\partial^2 f}{\partial h^2} + \frac{x(q)}{2}\big(\frac{\partial^2 f}{\partial h^2}\big)^2 = 0 \tag{7.29}$$

with "final" condition $f(1,h;x) = \ln\cosh(\beta h)$. The Parisi formula for the average free energy of the SK model states

$$-\lim_{n\to+\infty}\frac{\beta^{-1}}{n}\mathbb{E}[\ln Z] = \sup_{x\in\mathcal{Q}} f_P(x)\,. \tag{7.30}$$

To gain some insight into this rather complicated formula let us consider a simple lower bound to (7.30) obtained by taking

$$x(q) = \begin{cases} 0,\ q\in[0,q_0], \\ 1,\ q\in(q_0,1] \end{cases} \tag{7.31}$$

This is the cumulative distribution of the Dirac mass at $\bar{q}$, namely $\delta(q-q_0)$. For (7.31) the Parisi functional $f_{\mathrm{P}}(x)$ reduces to the replica symmetric potential function

$$f_{\mathrm{RS}}(q_0) \equiv -\frac{\beta}{4}(1-q_0)^2 - \beta^{-1}\int_{-\infty}^{+\infty} du\,\frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}}\ln\{2\cosh(\beta h + \beta u\sqrt{q}_0)\}, \tag{7.32}$$

and we have the lower bound

$$-\lim_{n\to+\infty}\frac{\beta^{-1}}{n}\mathbb{E}[\ln Z] \geq \sup_{q_0\in[0,1]} f_{\mathrm{RS}}(q_0). \tag{7.33}$$

The right hand side of this bound is called the "replica symmetric" free energy, and can be proven to be *equal* to the true free energy in the *high temperature phase*. To solve the variational problem on the right hand side of (7.33) we set $f'_{\mathrm{RS}}(q_0) = 0$ which can easily seen to be equivalent to one of the replica symmetric fixed point equations (compare with (7.25))

$$q_0 = \int_{-\infty}^{+\infty} du\,\frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}}\tanh^2\big\{\beta(h+u\sqrt{q_0})\big\} \tag{7.34}$$

So the replica symmetric free energy is equal to $f_{\mathrm{RS}}(q_*)$ where the maximizer q_* is a solution of this fixed point equation. The magnetization can as usual be obtained by differentiating the RS free energy with respect to the magnetic field

Figure 7.4 The Almeida-Thoulessphase transition line in the (h, T) plane separating the high temperature phase where the RS expression for the free energy is exact and the low temperature "glass phase" where it is not.

(compare with Chapter 2, Section 2.4)

$$m = \frac{1}{\beta}\frac{df_{\rm RS}(q_*)}{dh} = \frac{1}{\beta}\frac{\partial f_{\rm RS}(q_*)}{\partial h} + \frac{1}{\beta}\left(\frac{\partial f_{\rm RS}}{\partial q}\right)_{q_*}\frac{dq_*}{dh} = \frac{\partial f_{\rm RS}(q_*)}{\partial h}$$
$$= \int_{-\infty}^{+\infty} du\, \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \tanh\{\beta(h + u\sqrt{q_*})\} \tag{7.35}$$

(compare with (7.25)).

An important breakthrough towards the complete proof of (7.30) was the derivation of the lower bound (7.33) by Guerra and Toninelli and even of teh equality for sufficiently high temperatures. It turns out that this lower bound (7.33) is tight, i.e., the replica symmetric solution is exact, above the "Almeida-Thouless" line given by the equation

$$\beta^{-2} = \int_{-\infty}^{+\infty} du\, \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} (1 - \tanh^2(\beta h + \beta u\sqrt{q_*}))^2 \tag{7.36}$$

and depicted on figure 7.4. The Almeida-Thouless line separates a high temperature phase where the replica symmetric solution is exact, a the low temperature phase where this is not so. Below this line the free energy functional (7.28) has a supremum for non-trivial cdf's $\mathcal{P}(q) \neq \delta(q - q_0)$. This is related to the lack of concentration of the Edwards-Anderson parameter $q_{EA} = \frac{1}{n}\sum_{i=1}^{n}\langle s_i\rangle^2$ which acquires a non-trivial distribution, and to the failure of the assumption (7.23). The full description of the physical nature of the low temperature "glass phase" goes much beyond the free energy formula and still offers interesting challenges (see notes for references).

It is an interesting exercise to show that the RS formula cannot hold at low temperatures even on the $h = 0$ axis. An analysis To see this one considers the entropy obtained from this formula, i.e., $s_{\rm RS} = -\partial f_{\rm RS}/\partial(1/\beta)$ and checks the requirement that it remains non-negative for all temperatures.[5] As exlained in the previous section for $\beta < 1$ and $h = 0$ the only fixed point is $q_0 = 0$. The entropy then becomes $s_{\rm RS} = \ln 2 - \beta^2/4$ which is positive for $\beta < 2(\ln 2)^{1/2}$, and there is no obvious contradiction. However for $\beta > 1$ and $h = 0$ a non-trivial fixed point appears. A careful analysis of (7.34) shows that $q_0 \to 1 - \beta^{-1}\sqrt{2/\pi}$

[5] This is a necessary but not a sufficient condition for the correctness of the free energy.

when $\beta \to +\infty$, which leads to $s_{\rm RS} \to -1/(2\pi)$ for $\beta \to +\infty$, a negative entropy at zero temperature which is unacceptable.

7.6 Notes

The introduction of the SK model has been enormously fruitful in the theory of spin glasses, but also beyond, and nowadays the study of this model connects to algorithmic issues in signal processing, estimation and computer science. The first solution (Sherrington & Kirkpatrick 1975) used the *replica trick*, already introduced by (Edwards & Anderson 1975) in their study of the spin glass model (that bears their name) on a finite dimensional grid. The order parameter $q_{\rm EA} = \frac{1}{n}\sum_{i=1}^{n}\langle s_i\rangle^2$ characterizing the phase transition was already introduced in this work.

The first obstacle one faces in an attempt to compute the average free energy of a random spin system is the inability to switch the expectation and the logarithm in $\mathbb{E}[\ln Z]$. By Jensen's inequality $\mathbb{E}[\ln Z] \leq \ln \mathbb{E}[Z]$. The right hand side yields the *annealed approximation* (see exercises) and (**?**) proved that this is exact for $(\beta < 1, h = 0)$. The replica trick represents $\ln Z$ as $\lim_{n\to 0}(Z^n - 1)/n$, computes the moments $\mathbb{E}[Z^n]$ for integer n, and then somehow makes a "continuation" to obtain the limit $n \to 0$. A somewhat "blind" application of this trick leads to (7.32) - nowadays called *replica symmetric solution* - but already Sherrington and Kirkpatrick noticed that this can not be valid at low temperatures because it leads to a negative entropy. The replica symmetry alluded to here is "simply" the symmetry between the n copies of the system when n is an integer. However, clearly this symmetry is a mysterious concept when $n \to 0$. And even for n integer the physical meaning of the n "replicas" of the system is not so obvious. It was Parisi who in a series of papers first proposed a correct scheme for "breaking the replica symmetry" and obtained the correct free energy and entropies for any values of temperatures and magnetic field (Parisi 1980). The line separating the high temperature phase where the replica symmetric solution is valid, from the low temperature phase where one has to resort to Parisi's full replica symmetry breaking scheme, was first derived in (de Almeida & Thouless 1978). This algebraic approach, although powerful, was not easily accepted at first, and perhaps rightly so since still today it is not mathematically understood. It was soon replaced by a more probabilistic approach - the so called cavity method - that has much in common with the message passing point of view - and which will be explored in more depth in part III for the constraint satisfaction problem. After many efforts this approach has been mathematically established (Talagrand 2000). An account of the initial replica and cavity methods for the SK model is found in (Mézard, Parisi & Virasoro 1987*b*, Nishimori 2001) and the mathematics of the cavity method are found in the treatises (Talagrand 2003, Talagrand 2011, Panchenko 2013). A comprehensive set of references to decades of work is found there. We single

out the *interpolation method* (Guerra 2001, Guerra & Toninelli n.d.) establishing the equality in (7.33) above the AT line and the strict inequality below this line (Toninelli n.d.). The Guerra-Toninelli ideas have found applications well beyond the SK model in coding, compressive sensing and constraint satisfaction and form the subject of Chapter 13.

By its very nature the replica method studies average quantities. The more probabilistic cavity method approach is well suited to study single instances which is of crucial importance when it comes to algorithmic applications. An important step towards the probabilistic approach was taken in a crucial paper by Thouless, Anderson and Palmer who directly studied the free energy of single instances via a diagrammatic expansions (Thouless, Anderson & Palmer 1977). This paper contains a derivation of the Onsager reaction term. Such terms where introduced by (Onsager 1936) in his theory of dielectric properties of molecular liquids, greatly extending a long line of studies in physical chemistry. An account of this complicated history would lead us too far but interested readers can consult the nice book of (Boettcher 1973). This literature is not motivated by an algorithmic approach and in particular the TAP equations are viewed as a set of self-consistent (fixed point) relations that the local magnetizations should satisfy.

Using the TAP equations as an algorithm to estimate "latent variables" goes back to the work of (Opper & Winter 1996*b*, Opper & Winter 1996*a*) in the context of neural networks and is nowadays a widely used idea for systems on dense graphs. The approach taken in this chapter, starting from the belief propagation equations and working out an iterative form of the TAP equations, can be traced back to the work of (Kabashima & Saad 1998, Kabashima 2003) on communications problems. Although the best way to place the time indices in the TAP equations had probably been guessed before, the approach that starts from BP is principled and yields unambiguous results. This idea turns out to be important for algorithmic approaches to more complicated problems. The rigorous study of these TAP iterations was pioneered by (Bolthauzen 2014) who introduced techniques to prove crucial results such as (7.23). These techniques have then been used to control the performance of the the Approximate Message Passing algorithm. More on this is found in Chapter 8.

Problems

7.1 Belief propagation equations for pairwise spin systems. Consider a spin system on a general graph $G = (V, E)$ with Hamiltonian $\mathcal{H}(\underline{s}) = -\sum_{(i,j)\in E} J_{ij} s_i s_j$. Derive in detail the BP equations (7.4) for this system.

7.2 Heuristic derivation of the TAP equations. In this problem we go through a short heuristic argument that corrects the Curie-Weiss mean field equation and yields the TAP equations.

Complete this

7.3 Distribution of cavity fields in the TAP theory. The goal of this exercise is to numericaly justify the assumption (7.23) which forms teh bsis for

the derivation of the RS formula in this chapter. Consider the SK model with i.i.d Bernoulli(1/2) coupling constants $\tilde{J}_{ij} = \pm 1$ or $\tilde{J}_{ij}$ Gaussian with zero mean and unit variance. The TAP approximation to the BP equations reads

$$m_j^t = \tanh\{\beta(h + \sum_{i \neq j} \hat{h}_{i\to j}^t)\}$$

where the update of the cavity fields is

$$\hat{h}_{i\to j}^t = \frac{1}{\sqrt{n}} \tilde{J}_{ij} m_i^{t-1} - \frac{\beta}{n} m_j^{t-1}(1 - (m_i^{t-1})^2)$$

and the initialization $\hat{h}_{i\to j}^0 = 0$.

Take a number $N = 50$ of realizations (coupling constants) of the system of size $n = 500$ or 1000 and an iteration number say $t = 10$. Try values of $(h, T = \beta^{-1})$ in the high temperature regime: the following should be suitable $(h = 0.5, T = 1.2)$ and $(h = 1, T = 0.8)$.

(i) Plot the histogram of the total cavity field

$$\hat{h}_{\text{cav}}^t = \sum_{i \neq j} \hat{h}_{i\to j}^t.$$

This field is equal to a "Curie-Weiss" field to which the "Onsager reaction term" is subtracted. Plot also the histogram of the total Curie-Weiss contribution

$$h_{\text{CW}}^t = \sum_{i \neq j} \frac{1}{\sqrt{n}} \tilde{J}_{ij} m_i^{t-1}.$$

(ii) Check that the Edwards-Anderson parameter

$$q_{\text{EA}}^t = \frac{1}{n} \sum_{i=1}^n (m_i^t)^2.$$

is concentrated on its empirical mean over the N realizations.

(iii) Compare both histograms in (i) with the Gaussian distribution of zero mean and variance equal to the empirical mean of the Edwards-Anderson parameter. You should observe that the histogram of the cavity field agrees with this Gaussian, but not that of the CW field.

7.4 TAP ITERATIONS. Repeat the calculations of section (7.4) for $t = 3$ to show that $f_{j\to i}^{(3)}$ has mean zero and variance $q_2 = \mathbb{E}[(m_i^{(2)})^2]$ in thermodynamic limit.

7.5 REPLICA SYMMETRIC EQUATIONS. Consider the RS equations fixed point equations for m and q. Show rigorously that for $h = 0$ besides the trivial fixed point $m = 0, q = 0$ for $\beta > 1$ there is another non-trivial fixed point $m = 0, q \neq 0$ that is stable. Furthermore show that for $h \neq 0$ there is a unique fixed point for all β.

7.6 Annealed approximation. By Jensen's inequality we have $\mathbb{E}[\ln Z] \leq \ln \mathbb{E}[Z]$. Compute the right hand side for Gaussian couplings and deduce a lower bound for the free energy in the thermodynamic limit.

The expression in the lower bound is called the *annealed approximation* because it treats the quenched couplings on the same level as the spins. It can be shown to be exact for $(h = 0, \beta < 1)$. Check that on this interval it agrees with the replica symmetric solution (see (7.32)) $f_{RS} = -\beta^{-1} \ln 2 - \frac{\beta}{4}$. We stress that the lower bound is strict for $h \neq 0$ so the annealed approximation breaks down even at high temperature as soon as there is a non-zero magnetic field.

7.7 Entropy crisis. The main goal of this exercise is to show that the RS solution cannot be correct at sufficiently low temperatures. The idea is to show that the entropy obtained from $s_{\mathrm{RS}} = -\frac{\partial f_{\mathrm{RS}}}{\partial T}$ becomes negative at low temperatures which is not possible. Consider the RS fixed point equation (**??**) for $h = 0$. Compute the free energy and entropy corresponding to the unique trivial fixed point for $\beta < 1$. You should find that

$$f_{\mathrm{RS}} = -\beta^{-1} \ln 2 - \frac{\beta}{4} \quad \text{and} \quad s_{\mathrm{RS}} = -\ln 2 - \frac{\beta^2}{4}$$

and check that the entropy remains positive for $\beta < 1$. Show through an analysis of the fixed point equation that the non-trivial solution satisfies

$$q \approx 1 - \beta^{-1} \sqrt{\frac{2}{\pi}} \quad \text{for} \quad \beta \gg 1$$

and that the free energy and entropy become to leading order

$$f_{\mathrm{RS}} \approx -\frac{1}{2\pi\beta} \quad \text{and} \quad s_{\mathrm{RS}} \approx -\frac{1}{2\pi}\,.$$

The negativity of s_{RS} shows that the replica symmetric solution cannot be valid at low temperatures.

8 Compressive Sensing: Approximate Message Passing and State Evolution

Recall that a meaningful estimator for the compressive sensing problem is the Least Absolute Shrinkage Selection Operator (LASSO) given by

$$\widehat{\underline{x}}^{\mathrm{LASSO}}(\underline{y}) = \mathrm{argmin}_{\underline{x}} \Big\{ \frac{1}{2} \|\underline{y} - A\underline{x}\|_2^2 + \lambda \|\underline{x}\|_1 \Big\}. \tag{8.1}$$

The use of this estimator can be justified from several points of views as discussed in Chapters 1 and 3. For example one can settle for this estimator because, in the noiseless limit and for a certain range of parameters, the ℓ_1 and ℓ_0 minimization problems are equivalent (Theorem 1.1). Another point of view is the Bayesian one. The zero-temperature limit of the β-MMSE estimator for a Laplacian prior yields the LASSO, and the Laplacian prior is a simple and tractable model for sparse signals with unknown distribution. A justification for using this estimator can also be given in hindsight. We will see that this estimator works well in a fairly general setting. Together with the right structure for the measuring matrix we can even, in some cases, get optimal performance in terms of its asymptotic (in the size) behaviour if we look at the required number of measurements compared to the sparsity of the signal. However it is a long road until we can arrive at this conclusion in Chapter 14, so for the moment we will not worry about this. In the present chapter we simply want to implement the LASSO in an algorithmically efficient manner.

The basic idea to implement the LASSO is straightforward. We first set up a factor graph corresponding to (8.1) and mechanically write down the message-passing rules following the general framework about set out in Chapter 5, no thinking required. Since the LASSO asks for the a minimizer one possible starting point is the min-sum algorithm.[1] Quite surprisingly message passing works although the graph is dense and not at all sparse.

In principle this program only takes a few lines and we could stop at this point. But there are a few issues. For the straightforward message-passing algorithm the number of messages which need to be exchanged in each iteration is of quadratic oder in the graph size. This is true since the graph is dense. The second problem is that the messages are functions and not numbers as was the case for coding. This increases the complexity even further. Fortunately, as we

[1] This is to some degree a matter of convenience. An alternative derivation would start with the sum-product equations for a finite temperature formulation and then look at the zero temperature limit.

will see, one can approximate the original message-passing algorithm to (i) first simplify the messages to numbers, and (ii) bring down the number of messages which need to be exchanged in each iteration to linear order. For this second point we will proceed similarly to the derivation of TAP equations in Chapter 7. The final algorithm we derive is called AMP, where AMP stand for *Approximate Message Passing*

Besides the practical motivation to reduce complexity there is also another, perhaps more important, reason for going through these simplifications. The performance of the resulting AMP algorithm can be rigorously analyzed in detail. Even though the AMP algorithm is an approximation, it works very well and its performance can be caracterized precisely. In the context of coding we were able to assess the performance of belief propagation thanks to density evolution. In the large-size limit the state of belief propagation is given in terms of a distribution or density (of messages). Density evolution then allows to track this state as a function of the iteration. It is possible to develop a similar formalism for the AMP algorithm. In the context of compressive sensing this formalism is called *state evolution* (SE). As we will see, one can derive recursive equations for the mean square error whose average behaviour is tracked by SE.

An important application of SE pin-points an algorithmic phase transition curve in the "sparsity-measurement fraction" plane. Remarkably this curve is independent of the noise level and determines the region of equivalence of the ℓ_1 and ℓ_0 problems. It was first obtained by Donoho and Tanner by completely independent means.

8.1 LASSO for the Scalar Case

We begin with the analysis of a toy problem, namely the estimation of a scalar variable corrupted by noise. This turns out to be not only an interesting non-trivial problem, but also an important ingredient for the solution of the estimation of vector signals. Let then $y = x + z$ where $z \sim \mathcal{N}(0, \sigma^2)$. We assume that x is "sparse" in the sense that it is a random variable with Dirac mass $1 - \kappa$ at $x = 0$ and mass of "small" weight κ distributed (in an unknown way) for $x \neq 0$. More formally, this is the class $\mathcal{S}_\kappa$ of distributions of the form $p_0(x) = (1-\kappa)\delta(x)+\kappa\phi_0(x)$ where $\phi_0(x)$ is an unknown non-negative continuous probability distribution function normalized to one (but we suppose here that κ is known).

The LASSO estimator

$$\text{argmin}_x \Big\{ \frac{1}{2}(y - x)^2 + \lambda|x| \Big\}. \tag{8.2}$$

corresponds to the Hamiltonian $\mathcal{H}(x|y) = \frac{1}{2}(y - x)^2 + \lambda|x|$. Let us check where this Hamiltonian takes on its minimum. For $x > 0$ its derivative with respect to x equals $-(y-x)+\lambda$. Setting this derivative to 0 we find that (8.2) equals $y-\lambda$.

This solution is valid for $y > \lambda$. On the other hand for $x < 0$ the derivative is $-(y-x)-\lambda$. Setting this derivative to 0 we get that (8.2) equals $y+\lambda$. This is valid for $y < -\lambda$). For the remaining case $-\lambda < y < \lambda$ one checks the inequality $\frac{1}{2}y^2 \le \frac{1}{2}(y-x)^2 + \lambda|x|$ which means that (8.2) equals 0. Summarizing, we get the scalar estimator

$$\eta(y\,;\,\lambda) \equiv \begin{cases} y-\lambda, & \text{if } y > \lambda, \\ 0, & \text{if } -\lambda < y < \lambda, \\ y+\lambda, & \text{if } y < -\lambda. \end{cases}$$

This is called the *soft thresholding* estimator or function, and the corresponding graph is shown in Figure 8.1.

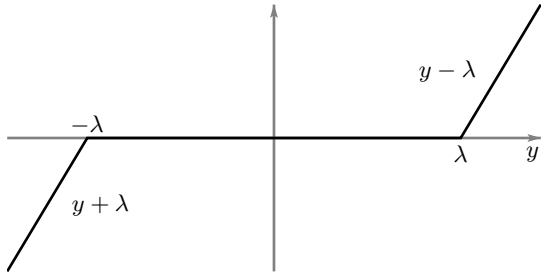


Figure 8.1 Graph of the soft-threshold function $\eta(y\,;\,\lambda)$. The parameter λ is the thresholding parameter.

In the above estimator we need to choose the parameter λ. Since $\phi_0(x)$ is unknown one reasonable criterion is: "choose the best λ for the worst prior in $\mathcal{S}_\kappa$." In mathematical terms we compute the "risk" or minimax-mean-square-error

$$\inf_{\lambda>0} \sup_{p_0(\cdot)\in\mathcal{S}_\kappa} \mathbb{E}[|\eta(Y,\lambda) - X|^2]. \tag{8.3}$$

Writing it explicitly and making the change of variables $y \to x+z$ the minimax-mean-square-error equals

$$\inf_{\lambda>0} \sup_{p_0(\cdot)\in\mathcal{S}_\kappa} \int dx p_0(x) \int dz\; p_0(x) \frac{e^{-\frac{1}{2\sigma^2}z^2}}{\sqrt{2\pi\sigma^2}} \big(\eta(x+z\,,\,\lambda) - x\big)^2. \tag{8.4}$$

It is natural to choose the threshold on the scale of the noise, i.e., we set $\lambda = \alpha\sigma$. We also note that under the change of variables $x \to \sigma x$, $z \to \sigma z$

$$\sigma p_0(\sigma x) = (1-\kappa)\delta(x) + \kappa\phi_0^{(\sigma)}(x) \qquad \text{with} \qquad \phi_0^{(\sigma)}(x) = \sigma\phi_0(\sigma x)$$

is a normalized distribution also belonging to $\mathcal{S}_\kappa$. In other words the ensemble $\mathcal{S}_\kappa$ is *scale invariant*. These important remarks imply that (8.4) equals

$$\sigma^2 \inf_{\alpha\ge 0} \sup_{p_0\in\mathcal{S}_\kappa} \int dx p_0(x) \int dz \frac{e^{-\frac{z^2}{2}}}{\sqrt{2\pi}} \Big(\eta(x+z\,;\,\alpha) - x\Big)^2 \tag{8.5}$$

and that the solution of the minimax problem is essentially independent of the noise level. The only thing that really depends on the noise level is the *overall* scale of the minimax-MSE. It should be clear that this is so because, since $\mathcal{S}_\kappa$ is scale invariant, σ^2 is the *only* scale or "dimensionfull" quantity in the problem. So dimensional analysis immediately tells us that the minimax-MSE must be proportional to σ^2. This is generally not true for the usual MMSE estimator which would be used if the prior were known. A known prior introduces another scale in the problem, besides σ^2.

It turns out that in the scalar case one can compute the worst case distribution and best possible α exactly. Let us set

$$M_{\text{scalar}}(\kappa, \alpha; p_0) \equiv \int dx p_0(x) \int dz \frac{e^{-\frac{z^2}{2}}}{\sqrt{2\pi}} \Big(\eta(x+z\,;\,\alpha) - x \Big)^2$$

and

$$M_{\text{scalar}}(\kappa, \alpha) \equiv \sup_{p_0 \in \mathcal{S}_\kappa} M_{\text{scalar}}(\kappa, \alpha; p_0), \qquad M_{\text{scalar}}(\kappa) \equiv \inf_{\alpha>0} M_{\text{scalar}}(\kappa, \alpha)$$

For fixed α the worst case distribution turns out to be[2]

$$p_{0,\text{worst}}(x) = (1-\kappa)\delta(x) + \frac{\kappa}{2}\delta_{+\infty}(x) + \frac{\kappa}{2}\delta_{-\infty}(x).$$

Using this expression one easily deduces that

$$M_{\text{scalar}}(\kappa, \alpha) = \kappa(1+\alpha^2) + (1-\kappa)\left[2(1+\alpha^2)\Phi(-\alpha) - 2\alpha \frac{e^{-\frac{\alpha^2}{2}}}{\sqrt{2\pi}} \right], \tag{8.6}$$

where $\Phi(\alpha) = \int_{-\infty}^{\alpha} du \, \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}}$ the cumulative distribution of a standardized Gaussian. To find the best possible α we now minimize (8.6) over $\alpha > 0$. Setting the derivative to zero we obtain

$$\kappa = \frac{2\big(\frac{e^{-\frac{\alpha^2}{2}}}{\sqrt{2\pi}} - \alpha\Phi(-\alpha)\big)}{\alpha + 2\big(\frac{e^{-\frac{\alpha^2}{2}}}{\sqrt{2\pi}} - \alpha\Phi(-\alpha)\big)}\,. \tag{8.7}$$

The right hand side of (8.7) is a monotone decreasing function of α, thus given κ there exist a unique $\alpha_{\text{best}}(\kappa)$ found by inverting (8.7). Finally, the minimax-MSE (8.5) for the scalar problem is

$$\sigma^2 M_{\text{scalar}}(\kappa) = \sigma^2 M_{\text{scalar}}(\kappa, \alpha_{\text{best}}(\kappa))\,. \tag{8.8}$$

[2] This was first derived by Donoho and Johnson (). See the notes and exercises for more information.

8.2 The vector case: preliminaries

From the point of view of statistical physics computing (8.1) is equivalent to minimizing the Hamiltonian (or cost function)

$$\mathcal{H}(\underline{x}) = \sum_{a=1}^{m} \frac{1}{2}(y_a - (A\underline{x})_a)^2 + \lambda \sum_{i=1}^{n} |x_i| \tag{8.9}$$

We explained in Chapter 3 that this cost function can be interpreted as a spin-glass Hamiltonian. The matrix A and the observation $\underline{y}$ are random, but once we have a realization they are considered fixed. These are the quenched (or frozen) variables. The degrees of freedom reside in the signal components x_i. These are "continuous spins" since $x_i \in \mathbb{R}$ rather than the usual binary variable $s_i = \pm 1$.

We are looking for the minimum of the Hamiltonian, and while the scalar case could be solved straightforwardly, for the vector case we have to settle for an algorithmic solution. According to the factor graph framework developed in Chapter 5 we use the min-sum algorithm. The underlying factor graph is the complete bipartite graph with variable nodes corresponding to the signal components x_i, and two types of factor nodes corresponding to the factors

$$\frac{1}{2}(y_a - (A\underline{x})_a)^2, \qquad \text{and} \qquad \lambda|x_i|.$$

A straightforward application of message passing rules leads to the following equations involving two types of messages, call them $\widehat{E}_{a\to i}(x_i)$ and $E_{i\to a}(x_i)$, $i = 1, \cdots, n$ and $a = 1, \cdots, m$,[3]

$$\begin{cases} E^{t+1}_{i\to a}(x_i) = \lambda|x_i| + \sum_{b\in\partial i\setminus a} \widehat{E}^t_{b\to i}(x_i), \\ \widehat{E}^{t+1}_{a\to i}(x_i) = \min_{\sim x_i}\Big\{\frac{1}{2}(y_a - (A\underline{x})_a)^2 + \sum_{j\in\partial a\setminus i} E^{t+1}_{j\to a}(x_j)\Big\}. \end{cases} \tag{8.10}$$

In addition we have the initialization

$$\begin{cases} E^0_{i\to a}(x_i) = \lambda|x_i|, \\ \widehat{E}^0_{a\to i}(x_i) = \min_{\sim x_i}\{\frac{1}{2}(y_a - (A\underline{x})_a)^2 + \sum_{j\in\partial a\setminus i} \lambda|x_j|\}. \end{cases} \tag{8.11}$$

The min-sum estimate at time t, call it $\widehat{x}^t_i(\lambda)$, is computed from

$$\widehat{x}^t_i = \operatorname{argmin}_{x_i} E^t_i(x_i), \tag{8.12}$$

where

$$E^t_i(x_i) = \lambda|x_i| + \sum_{b\in\partial i} \widehat{E}^t_{b\to i}(x_i). \tag{8.13}$$

Recall that in Chapter 5 we discussed the BP equations for compressive sensing. As explained there, the min-sum equations (8.10) can be obtained by taking the $\beta \to +\infty$ limit of BP equations. Alternatively one can derive them by a direct application of the distributive law for $(\min, +)$ operations.

[3] Recall that $\min_{\sim x_i}$ means minimisation over all components of $\underline{x}$ except x_i.

We stress here that $\widehat{\underline{x}}^t$ in (8.12) is the *min-sum estimate* - an algorithmic quantity - and although one might hope that as $t \to +\infty$ it converges to $\widehat{\underline{x}}^{\mathrm{LASSO}}(\underline{y})$ this is far from obvious a priori. We will have the occasion to come back to this issue of their comparison in Section 8.8.

Running min-sum on a complete bipartite graph with a bipartition of size n and m respectively, requires to transmit $\Theta(mn)$ messages at each iteration. For large instances this complexity is prohibitive. We will show that we can get away with linear complexity. To be sure, the algorithm which we shall develop is an approximation of the original min-sum message passing.[4] How can we derive such an approximation? The model and the situation is analogous to that of the SK model. Therefore, it should not come as a surprise that the methodology which we follow for the analysis is also similar. We have seen in the previous chapter that for the SK model we can go from the BP equations to the TAP equations by exploiting the smallness of interaction coefficients, more precisely $J_{ij} \sim \mathcal{N}(0, \frac{1}{n})$ or $J_{ij} = \pm\frac{1}{\sqrt{n}}$ with Bernoulli(1/2) signs. In the present case we can also exploit $A_{ai} \sim \mathcal{N}(0, 1/m)$ or is Bernoulli(1/2) with $A_{ai} = \pm 1/\sqrt{m}$ and as shown in section 8.4 this leads to significant simplifications and linear complexity. Both models lead to the same final result and we will use the common notation $A_{ai} \sim 1\sqrt{m}$ to express the order of magnitude of the matrix elements.

Before we tacle this derivation there is one complication we first have to deal with. Contrary to the SK or coding models the "spin variables" (here the signal components) are not binary and therefore the min-sum messages cannot be exactly parametrized by numbers (the cavity fields in the binary case). However it turns out that a *quadratic approximation* of the messages is possible, which approximates each message by a set of two real numbers. This is the subject of the next section.

8.3 Quadratic Approximation

The following is a fairly technical calculation. In a first reading the reader may just look at formulas (8.14) and (8.17) that define a parametrization of messages in terms of four real values $\alpha_{a\to i}$, $\beta_{a\to i}$, $x_{i\to a}$, $\gamma_{i\to a}$, and then skip forward directly to the message passing equations (8.18) and (8.19). These equations are all that is needed for the derivation of the AMP algorithm in Section 8.4.

A simple but crucial observation is that in the message passing expression (8.10) for $\widehat{E}^{t-1}_{a\to i}(x_i)$ the x_i dependence only enters as $A_{ai}x_i$ in $(A\underline{x})_a$. Now since $A_{ai}x_i \sim 1/\sqrt{m}$ this contribution is small as $n \to +\infty$, and we can consider the Taylor expansion of $\widehat{E}^{t+1}_{a\to i}(x_i)$ in powers of $A_{ai}x_i$. Keeping only the lowest order

[4] Further, recall that we are not operating on a tree and so even a full fledged message passing algorithm is not necessarily optimal. There is therefore no reason to insist on an exact implementation of the BP algorithm.

terms

$$\widehat{E}^{t+1}_{a\to i}(x_i) = \widehat{E}^{t+1}_{a\to i}(0) - \alpha^{t+1}_{a\to i}(A_{ai}x_i) + \frac{1}{2}\beta^{t+1}_{a\to i}(A_{ai}x_i)^2 + O((A_{ai}x_i)^3), \quad (8.14)$$

where the Taylor coefficients $\alpha^{t+1}_{a\to i}$ and $\beta^{t+1}_{a\to i}$ are real numbers that we determine later. These are two real valued messages that approximate $\widehat{E}^{t+1}_{a\to i}(x_i)$. Equation (8.14) constitutes the *quadratic approximation* for $\widehat{E}^{t+1}_{a\to i}(x_i)$. Replacing (8.14) in the first message passing equation (8.10) we get

$$\begin{aligned} E^{t+1}_{i\to a}(x_i) &\approx E^{t+1}_{i\to a}(0) + \lambda|x_i| - x_i \sum_{b\in\partial i\setminus a} A_{bi}\alpha^t_{b\to i} + \frac{x_i^2}{2}\sum_{b\in\partial i\setminus a} A^2_{bi}\beta^t_{b\to i} \\ &= E^{t+1}_{i\to a}(0) - \frac{\lambda(a^t_1)^2}{2a^t_2} + \frac{\lambda}{a^t_2}\big\{a^t_2|x_i| + \frac{1}{2}(x_i - a^t_1)^2\big\} \end{aligned} \quad (8.15)$$

where

$$a^t_1 = \frac{\sum_{b\in\partial i\setminus a} A_{bi}\alpha^t_{b\to i}}{\sum_{b\in\partial i\setminus a} A^2_{bi}\beta^t_{b\to i}}, \qquad a^t_2 = \frac{\lambda}{\sum_{b\in\partial i\setminus a} A^2_{bi}\beta^t_{b\to i}}. \quad (8.16)$$

The second equality in (8.15) has been obtained by completing the square. The calculation presented for the scalar case shows that the minimum of the term $\{\dots\}$ is equal to $\eta(a^t_1\,;\,a^t_2)$. Thus, when the right hand side of (8.15) is expanded around its minimum one finds (this is justified below)

$$E^{t+1}_{i\to a}(x_i) = \text{Constant} + \frac{1}{2\gamma^{t+1}_{i\to a}}(x_i - x^{t+1}_{i\to a})^2 + O((x_i - x_{i\to a})^3) \quad (8.17)$$

where

$$x^{t+1}_{i\to a} = \eta(a^t_1\,;\,a^t_2), \qquad \gamma^{t+1}_{i\to a} = \frac{a^t_2}{\lambda}\eta'(a^t_1\,;\,a^t_2) \quad (8.18)$$

where η' is the derivative of η with respect to the first argument. Equation (8.17) constitutes the *quadratic approximation* for $E^{t+1}_{i\to a}(x_i)$.

Why can one hope that it is a good approximation to expand (8.15) near its minimum? One way to understand this is to recall the connection between min-sum and BP. For β large the BP messages are proportional to $e^{-\beta E^{t+1}_{i\to a}(x_i)}$, a weight that is dominated by x_i close to the minimum of the exponent. Once this is accepted, it remains to find this minimum and write down the Taylor expansion around it. For $\lambda \neq 0$ the absolute value is not differentiable at the origin so the derivation involves a few technical subtleties that are worth discussing. [5] In the scalar minimisation problem we learned that the minimum of (8.15) over x_i is attained at $x^t_{i\to a} = \eta(a^t_1\,;\,a^t_2)$. The expansion is best performed by first assuming that $x^t_{i\to a} > 0$, i.e. $x^t_{i\to a} = \eta(a^t_1\,;\,a^t_2) = a^t_1 - a^t_2$. In this case we can set $|x_i| = x_i$ and the first derivative of (8.15) is $(a^t_2 + (x_i - a^t_1))\lambda/a^t_2$ which vanishes at $x^t_{i\to a}$. The second derivative is equal to $\lambda/a^t_2 = \lambda/(a^t_2\eta'(a^t_1\,;\,a^t_2)) = 1/\gamma^t_{i\to a}$. Therefore (8.17) holds when $x^t_{i\to a} > 0$. The reader can work out the case $x^t_{i\to a} < 0$ in

[5] For $\lambda = 0$ we have $\eta(y;\lambda) = y$ so that cubic and higher order terms in (8.17) vanish, and $x^{t+1}_{i\to a} = a^t_1$, $\gamma^{t+1}_{i\to a} = 1/\sum_{b\in\partial i\setminus a} A^2_{ai}\beta^t_{b\to i}$.

a similar way. Finally we consider the singular case $x^t_{i\to a} = 0$, i.e. $\eta(a^t_1\,;\,a^t_2) = \eta'(a^t_1\,;\,a^t_2) = 0$. At the origin the first derivative of $|x_i|$ has a jump, and the second derivative is formally infinite. Therefore we have to take $\gamma^t_{i\to a} = 0$ which is consistent with $\gamma^t_{i\to a} = \eta'(a^t_1\,;\,a^t_2)a^t_2/\lambda$.

The final step is to determine $\alpha^t_{b\to i}$ and $\beta^t_{b\to i}$. For this we replace (8.17) in the second message passing equation (8.10). Then we compare with the expansion (8.14). After a tedious but *exact* algebraic calculations this yields

$$\alpha^t_{a\to i} = \frac{y_a - \sum_{j\in\partial a\setminus i} A_{aj}x^t_{j\to a}}{1+\sum_{j\in\partial a\setminus i} A^2_{aj}\gamma^t_{j\to a}}, \qquad \beta^t_{a\to i} = \frac{1}{1+\sum_{j\in\partial a\setminus i} A^2_{aj}\gamma^t_{j\to a}}. \tag{8.19}$$

Let us summarize these calculations. The quadratic approximation assumes that the expansions (8.14) and (8.17) to second order are good approximations and neglects cubic and higher order terms. The min-sum equations (8.10) then reduce to a set of four equations (8.18), (8.19) for real valued messages $x^t_{i\to a}$, $\gamma^t_{i\to a}$, $\alpha^t_{a\to i}$, $\beta^t_{a\to i}$.

8.4 Derivation of the AMP Algorithm

The scaling $A^2_{ai} \sim 1/m$ induces important simplifications that we now derive heuristically. It is useful to keep in mind the Bernoulli model for which $A^2_{ai} = 1/m$ and many arguments become more transparent. But the end results do not depend on the model as long as the matrix elements are iid with sub-gaussian distribution.

Since we are deriving an algorithm in a sense we don't care about making this paragraph rigorous. I don't think there is a rigorous derivation in the literature. Rigor is only needed in sec 8.6 in principle. Maybe we should spell out this clearly somewhere.

First simplications of (8.18) and (8.19)

Our derivation rests on the assumption that the term in the denominator of (8.19)

$$1+\sum_{j\in\partial a\setminus i} A^2_{aj}\gamma^t_{j\to a}$$

can be treated as independent of a and i. Why might this be true? Note that $A^2_{aj} \sim 1/m$ and that we sum over $n-1$ terms. This sum is therefore up to a negligible term equal to the empirical mean of $\gamma^t_{j\to a}$ over all edges of the graph, and we therefore expect this to concentrate on a value independent of a and i. Thus we set

$$1+\sum_{j\in\partial a\setminus i} A^2_{aj}\gamma^t_{j\to a} \equiv \frac{\theta_t}{\lambda} \tag{8.20}$$

and we treat θ_t as independent of a and i. The update equation for θ_t is discussed later on. We also set

$$r^t_{a\to i} = y_a - \sum_{j\in\partial a\setminus i} A_{aj}x^t_{j\to a}, \tag{8.21}$$

so that (8.19) become

$$\alpha^t_{a\to i} = \frac{\lambda}{\theta_t} r^t_{a\to i}, \qquad \beta^t_{a\to i} = \frac{\lambda}{\theta_t}. \tag{8.22}$$

Let us now look at a^t_1 and a^t_2 in (8.16). From $\beta^t_{b\to i} = \lambda/\theta_t$ we deduce that the denominator of a^t_1 and a^t_2 is equal to

$$\frac{\lambda}{\theta_t} \sum_{b\in\partial i\setminus a} A^2_{bi}$$

Furthermore note that $\sum_{b\in\partial i\setminus a} A^2_{bi} \approx 1$ With these remarks we obtain

$$a^t_1 = \sum_{b\in\partial i\setminus a} A_{bi} r_{b\to i}, \qquad a^t_2 = \theta_t. \tag{8.23}$$

Replacing (8.23) in the first message passing equation (8.18) one finds

$$x^{t+1}_{i\to a} = \eta\big(\sum_{b\in\partial i\setminus a} A_{bi} r^t_{b\to i}\,;\,\theta_t\big). \tag{8.24}$$

So far the message-passing rules boil down to (8.21) and (8.24). But we still need an equation for the updates of θ_t. This is easily obtained by multiplying the second equation in (8.18) by A^2_{ai} and summing over i. We get

$$1 + \sum_{i\in\partial a} A^2_{ai}\gamma^{t+1}_{i\to a} = 1 + \sum_{i\in\partial a} A^2_{ai}\frac{a^t_2}{\lambda}\eta'(a^t_1\,;\,a^t_2) \tag{8.25}$$

which, in the large size limit, becomes equivalent to (using (8.20), (8.23) and $A^2_{ai}\sim 1/m$)

$$\theta_{t+1} = \lambda + \frac{\theta_t}{m}\sum_{i\in\partial a}\eta'\big(\frac{1}{\mu}\sum_{b\in\partial i\setminus a} A_{bi} r^t_{b\to i}\,;\,\frac{\theta_t}{\mu}\big) \tag{8.26}$$

Notice a nice property of the thresholding function: the derivative $\eta' = 0$ when $\eta = 0$ and $\eta' = 1$ when $\eta \neq 0$. This prompts us to introduce a notation for the "0-absolute value" of a real number,

$$|z|_0 = \begin{cases} 1, & \text{if } z \neq 0,\\ 0, & \text{if } z = 0.\end{cases}$$

Thanks to (8.24) the update equation (8.26) can be written in the nice form

$$\theta_{t+1} = \lambda + \frac{\theta_t}{m}\sum_{i\in\partial a} |x^{t+1}_{i\to a}|_0. \tag{8.27}$$

We have simplified the min-sum equations down to (8.21), (8.24) and (8.27) but at this point we still have $\Theta(nm)$ messages to update at each iteration. A further simplification bringing this complexity down to linear order is the subject of the next subsection.

But before we address this issue it is useful to first consider the estimate

obtained by minimizing $E_i(x_i)$ (see Equs. (8.12), (8.13)). Without going into all details of calculations (similar to Section 8.3) the reader should not be surprized that within the quadratic approximation one finds

$$E_i^t(x_i) \approx \frac{1}{2\gamma_i^t}(x_i - \widehat{x}_i^t)^2 + O((x_i - \widehat{x}_i^t)^3), \tag{8.28}$$

where

$$\widehat{x}_i^t = \eta(\tilde{a}_1^t\,;\,\tilde{a}_2^t), \tag{8.29}$$

and

$$\tilde{a}_1^t = \frac{\sum_{b\in\partial i} A_{bi}\alpha_{b\to i}^t}{\sum_{b\in\partial i} A_{bi}^2\beta_{b\to i}^2}, \qquad \tilde{a}_2^t = \frac{\lambda}{\sum_{b\in\partial i} A_{bi}^2\beta_{b\to i}^t}. \tag{8.30}$$

This leads to an estimate at time t of the form

$$\widehat{x}_i^t = \eta(\sum_{b\in\partial i} A_{bi} r_{b\to i}^t\,;\,\theta_t). \tag{8.31}$$

In (8.31) *all* messages $r_{b\to i}^t$ entering nodes i are involved, wheras in (8.24) the message $r_{a\to i}^t$ is ommitted. This is a usual feature of mesage passing.

Finals steps

We are now ready to proceed from (8.21), (8.24), (8.27), to the final steps leading to the AMP algorithm. From (8.24) we have

$$\begin{aligned} x_{i\to a}^{t+1} &= \eta(\sum_{b\in\partial i} A_{bi} r_{b\to i}^t - A_{ai} r_{a\to i}^t\,;\,\theta_t) \\ &\approx \eta(\sum_{b\in\partial i} A_{bi} r_{b\to i}^t\,;\,\theta_t) - A_{ai} r_{a\to i}^t \eta'(\sum_{b\in\partial i} A_{bi} r_{b\to i}^t\,;\,\theta_t) \\ &= \widehat{x}_i^t - A_{ai} r_{a\to i}^t |\widehat{x}_i^t|_0, \end{aligned} \tag{8.32}$$

The second approximate equality above is obtained by a Taylor expansion to first order in $A_{ai} r_{a\to i}^t \sim 1/\sqrt{m}$. A similar step was performed when we derived TAP from BP equations in Chapter 7. This step is crucial and will lead to a sort of Onsager reaction term. The last equality follows by remarking again that that $\eta' = 1$ (resp. $\eta' = 0$) whenever $\eta \neq 0$ (resp. $\eta = 0$) and using (8.31). Replacing (8.32) in (8.21),

$$\begin{aligned} r_{a\to i}^t &= y_a - \sum_{j\in\partial a\setminus i} A_{aj}\widehat{x}_j^{t-1} + \sum_{j\in\partial a\setminus i} A_{aj}^2 r_{a\to j}^{t-1} |\widehat{x}_j^{t-1}|_0 \\ &= y_a - \sum_{j\in\partial a} A_{aj}\widehat{x}_j^{t-1} + \sum_{j\in\partial a} A_{aj}^2 r_{a\to j}^{t-1} |\widehat{x}_j^{t-1}|_0 + A_{ai}\widehat{x}_i^{t-1} - A_{ai}^2 r_{a\to i}^{t-1} |\widehat{x}_i^{t-1}|_0. \end{aligned}$$

We see that $r_{a\to i}^t$ consists of the three first terms which are of order one and are independent of i, and the last two which do depend on i but are of order $1/\sqrt{m}$

and $1/m$. So let us write

$$r^t_{a\to i} = r^t_a + \delta r^t_{a\to i},$$

with

$$r^t_a = y_a - \sum_{j\in\partial a} A_{aj}\widehat{x}^{t-1}_j + r^{t-1}_a \sum_{j\in\partial a} A^2_{aj}|\widehat{x}^{t-1}_j|_0 \tag{8.33}$$

for the $O(1)$ terms, and a rest $\delta r^t_{a\to i} \approx A_{ai}\widehat{x}^{t-1}_i$ for the $O(1/\sqrt{m})$ term. The $O(1/m)$ term is neglected. Using again $A^2_{ai} \sim \frac{1}{m}$ (8.33) yields

$$r^t_a = y_a - \sum_{j\in\partial a} A_{aj}\widehat{x}^{t-1}_j + r^{t-1}_a \frac{\|\underline{\widehat{x}}^{t-1}\|_0}{m}. \tag{8.34}$$

Moreover, replacing $r^t_{b\to i} = r^t_b + \delta r^t_{b\to i} \approx r^t_b + A_{bi}\widehat{x}^{t-1}_i$ in estimate (8.31) we find

$$\begin{aligned}\widehat{x}^t_i &= \eta\Big(\sum_{b\in\partial i} A_{bi}r^t_b + \sum_{b\in\partial i} A^2_{bi}\widehat{x}^{t-1}_i\,;\,\theta_t\Big)\\ &= \eta\Big(\sum_{b\in\partial i} A_{bi}r^t_b + \widehat{x}^{t-1}_i\,;\,\theta_t\Big).\end{aligned} \tag{8.35}$$

Finally retaining the leading term in (8.32) the update equation (8.27) for θ_t becomes

$$\theta_{t+1} = \lambda + \theta_t \frac{\|\underline{\widehat{x}}^t\|_0}{m}. \tag{8.36}$$

Equations (8.34), (8.35) and (8.36) form the AMP algorithm.

To conclude, recall that the current form of AMP has been derived for an unknown sparse prior distribution. With only minor extra effort we can derive a variant of AMP adapted to the case of a known (sparse) prior signal distribution when the MMSE estimator is used instead. This is discussed in Section 8.9.

8.5 AMP algorithm for the LASSO

Let us now collect the fruits of our efforts and discuss the AMP algorithm as well as a practical variant.

The final AMP equations (8.34), (8.35), (8.36) can be written in a somewhat more compact notation

$$\begin{cases}\widehat{x}^t_i = \eta(\widehat{x}^{t-1}_i + (A^T\underline{r}^t)_i\,;\,\theta_t),\\ r^t_a = y_a - (A\underline{\widehat{x}}^{t-1})_a + r^{t-1}_a\frac{\|\underline{\widehat{x}}^t\|_0}{m},\\ \theta_{t+1} = \lambda + \theta_t\frac{\|\underline{\widehat{x}}^t\|_0}{m}.\end{cases} \tag{8.37}$$

Clearly, in (8.37) there are no messages flowing on edges, but instead $\Theta(n)$ *values* updated at each iteration; we have gained one order of complexity with respect to the initial message passing equations. This is similar to the complexity reduction

we encountered when going from BP to TAP equations for the Sherrington-Kirkpatrick model.

There are other ways to update θ_t which are somewhat more heuristic and lead to similar algorithmic performance. Here we discuss a variant of (8.37) with a simpler update of θ_t which has the advantage of lending itself more easily to a theoretical performance analysis (as shown in the next two sections). In the scalar case we saw in Section 8.1 that the threshold in η is naturally set on the scale of the noise, i.e. $\lambda = \alpha\sigma$ (and then the best possible α is determined by solving a minimax problem). In that case, σ was the standard deviation of $y - x$. By analogy, for the vector case it is natural to take θ_t on the scale of the standard deviation of $(A^T \underline{r}^t)_i$ which is the term added to the estimate $\widehat{x}_i^{t-1}$ in the first AMP equation (8.37). A rough guess for this standard deviation is

$$\sqrt{\underline{r}^T \mathbb{E}[AA^T]\underline{r}^t} = \frac{1}{\sqrt{m}} \|\underline{r}^t\|_2.$$

Therefore we take the following heuristic value for the soft threshold at time t

$$\theta_t = \frac{\alpha}{\sqrt{m}} \|\underline{r}^t\|_2. \tag{8.38}$$

This completely defines a useful variant of the AMP algorithm

$$\begin{cases} \widehat{x}_i^t = \eta(\widehat{x}_i^{t-1} + (A^T \underline{r}^t)_i \,;\, \frac{\alpha}{\sqrt{m}} \|\underline{r}^t\|_2), \\ r_a^t = y_a - (A\underline{\widehat{x}}^{t-1})_a + r_a^{t-1} \frac{\|\underline{\widehat{x}}^t\|_0}{m}. \end{cases} \tag{8.39}$$

whose performance we will assess in Section 8.6.

The AMP algorithm (8.39) is almost the same than the much older *Iterative Soft Thresholding* (IST) algorithm,

$$\begin{cases} \widehat{x}_i^t &= \eta(\widehat{x}_i^t + (A^T \underline{r}^t)_i \,;\, \frac{\alpha}{\sqrt{m}} \|r^t\|_2), \\ {r_a}^t &= y_a - (A\underline{\widehat{x}}^{t-1})_a. \end{cases} \tag{8.40}$$

The fundamental difference between IST and AMP lies in the Onsager reaction term, namely $r_a^{t-1} \frac{\|\widehat{x}^{t-1}\|_0}{m}$ which is absent in (8.40). One can run experiments and check that this term is responsible for the improved performance of AMP over IST. One typically obtains a much smaller empirical MSE with much lesser iterations.

One could perhaps hope that, when the IST algorithm is tested numerically, the unthresholded estimate

$$\widehat{x}_i^t + (A^T \underline{r}^t)_i = \widehat{x}_i^t + \frac{1}{\sqrt{m}} \sum_{b=1}^{m} \tilde{A}_{bi} r_b^t$$

has a Gaussian histogram (here we set $A = \frac{1}{\sqrt{m}} \tilde{A}$). It is the subject of an exercise to show that *this is not so*. Correlations between the terms in the sum develop along the trajectory of the IST algorithm and the central limit theorem does not hold. Remarkably, it turns out that when the extra Onsager

correction term is added to r_b^t (so AMP is used) the histogram of this unthresholded estimate becomes Gaussian! The Onsager term has the effect of cancelling the correlations between the terms in the sum. Again, the situation is exactly analogous to the one in the SK model. We saw that the naive Curie-Weiss mean field, $\frac{1}{\sqrt{n}}\sum_{i=1;i\neq j}^{n}\tilde{J}_{ij}m_i^{t-1}$, does not have a Gaussian histogram; whereas when the Onsager correction is added the TAP local field, $\frac{1}{\sqrt{n}}\sum_{i=1;i\neq j}^{n}\tilde{J}_{ij}m_i^{t-1}-\beta m_j^{(t-1)}(1-q^{t-1})$, has a Gaussian histrogram.

8.6 Heuristic Derivation of State Evolution

In coding theory we derived density evolution equations that track the state of the BP algorithm, i.e. the probability distributions of messages. Density evolution then allows to compute the probability of a decoding error and assess the performance of a coding ensemble. There exist a similar formalism called *State Evolution* (SE) that tracks the state of the AMP algorithm (8.39) and allows to calculate its performance. For the "state" at time t we take the square error $\|\underline{\widehat{x}}^t-\underline{x}\|^2$ incurred by the estimate $\underline{\widehat{x}}^t$ for a given input signal $\underline{x}$. State evolution tracks the average behavior of the square error in the large size limit. In other words we seek an update equation for

$$\tau_t=\lim_{n\to+\infty}\frac{1}{n}\|\underline{\widehat{x}}^t-\underline{x}\|^2 \tag{8.41}$$

where the limit is taken at fixed measurement fraction $\mu=m/n$ and sparsity $\kappa=k/n$. A priori this is a random quantity which depends on the quenched variables: measurement matrix, noise and input signal. However we will see that it satisfies a deterministic equation, and although the analysis is semi-heuristic this strongly suggests that (8.41) concentrates. This is indeed true and can be shown rigorously. More precisely with probability one $\tau_t=\lim_{n\to+\infty}\mathbb{E}[\|\underline{\widehat{x}}^t-\underline{x}\|^2]/n$.

What is known about the exchange of t and n limits here ?

The key feature that allows us to derive a closed form equation relating τ_{t+1} to τ_t is the Gaussianity of the untresholded estimate for a given the input signal. As explained in the previous section numerical experiments show that with the Onsager term present, the sum $\frac{1}{\sqrt{m}}\sum_{b=1}^{m}\tilde{A}_{bi}r_b^t$ behaves as if the central limit theorem applied (from now on we set $\tilde{A}=\frac{1}{\sqrt{m}}A$). Effectively, it is equivalent to remove the Onsager term from the algorithm and sample afresh the measurement matrix at each time step so that the law of large numbers applies. This remarkable observation is at the basis of the "conditioning technique"[6] which allows for a rigorous derivation of SE. The rigorous proofs would lead us too far here, and we will simply accept, based on numerical observations, that the Onsager term $r_a^{t-1}\frac{\|\underline{\widehat{x}}^t\|_0}{m}$ can be removed if simultaneously we replace the quenched measurement matrix elements $\tilde{A}_{bi}$ by new iid realizations $\tilde{A}_{bi}^t$ sampled afresh from $\mathcal{N}(0,1)$ or uniformly from $\{-1,+1\}$.

[6] Originally developed by Erwin Bolthauzen in his analysis of TAP iterations for the Sherrington-Kirkparick model ().

In other words we are analyzing the following set of equations (compare with (8.39))

$$\begin{cases} \widehat{x}_i^t = \eta(\widehat{x}_i^{t-1} + \frac{1}{\sqrt{m}}(\tilde{A}^{t\,T}\underline{r}^t)_i\,;\, \frac{\alpha}{\sqrt{m}}\|\underline{r}^t\|_2), \\ r_a{}^t = \frac{1}{\sqrt{m}}(\tilde{A}^t\underline{x})_a + z_a - \frac{1}{\sqrt{m}}(\tilde{A}^t\underline{\widehat{x}}^{t-1})_a. \end{cases} \tag{8.42}$$

where to be consistent we have also replaced the measurements $\underline{y} = \frac{1}{\sqrt{m}}\tilde{A}\underline{x} + \underline{z}$ by "new measurements" at each time step $\underline{y}^t = \frac{1}{\sqrt{m}}\tilde{A}^t\underline{x} + \underline{z}$, $z_a \sim \mathcal{N}(0,1)$. We will shortly show that in thermodynamic limit: (i) the first argument of the thresholding function in (8.42) tends to a Gaussian with mean $\underline{x}$ and variance $(\sigma^2+\mu^{-1}\tau_{t-1}^2)^{1/2}$; (ii) the second argument $\frac{\alpha}{\sqrt{m}}\|\underline{r}^t\|_2$ tends to $\alpha(\sigma^2+\mu^{-1}\tau_{t-1}^2)^{1/2}$. Thus in (8.42) each component x_i^t is distributed as the *random variable*

$$\widehat{x}^t = \eta\left(x + u\sqrt{\sigma^2 + \frac{\tau_{t-1}^2}{\mu}}\,;\, \alpha\sqrt{\sigma^2 + \frac{\tau_{t-1}^2}{\mu}}\right) \tag{8.43}$$

where $u \sim \mathcal{N}(0,1)$ and $x \sim p_0(\cdot)$. Using definition (8.41) and the law of large numbers, we obtain the SE updates

$$\tau_t^2 = \int dx\, p_0(x) \int du\, \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \left\{\eta\left(x + u\sqrt{\sigma^2 + \frac{\tau_{t-1}^2}{\mu}}\,;\, \alpha\sqrt{\sigma^2 + \frac{\tau_{t-1}^2}{\mu}}\right) - x\right\}^2. \tag{8.44}$$

The consequences of SE for the phase diagram the AMP algorithm are discussed in the next section. For completeness we first give a somewhat informal proof of points (i) and (ii) above.

Technical details leading to (8.44)

Let us show point (i). Merging the two equations together in (8.42) the first argument of the thresholding function becomes

$$x_i + \frac{1}{\sqrt{m}}\sum_{b=1}^m \tilde{A}_{bi}^t z_b + \sum_{j=1}^n (\delta_{ij} - \frac{1}{m}(\tilde{A}^{t\top}\tilde{A}^t)_{ij})(\widehat{x}_j^{(t-1)} - x_j) \tag{8.45}$$

We discuss the behavior of each sum in the thermodynamic limit. Clearly, given $\underline{z}$, from the central limit theorem

$$\frac{1}{\sqrt{m}}\sum_{b=1}^m \tilde{A}_{bi}^t z_b \tag{8.46}$$

tends to a Gaussian with zero mean and variance $\frac{1}{m}\sum_{b=1}^m z_b^2 \to \sigma^2$. Next, again by the central limit theorem, one shows that the matrix entries $(\delta_{ij} - \frac{1}{m}(\tilde{A}^{t\top}\tilde{A}^t)_{ij})$ tend to a zero mean Gaussian with variance $1/m$. Looking at the covariance of these entries we see that they are independent to leading order. Thus the term

$$\sum_{j=1}^n (\delta_{ij} - \frac{1}{m}(\tilde{A}^{t\top}\tilde{A}^t)_{ij})(\widehat{x}_j^{t-1} - x_j) \tag{8.47}$$

is also a Gaussian, with zero mean and variance

$$\frac{1}{m}\sum_{j=1}^{n}(\widehat{x}_j^{t-1}-x_j)^2=\frac{\tau_{t-1}^2}{\mu}$$

Finally, one can look at the covariance of the two approximate Gaussian variables in (8.46) and (8.47) and show that they are approximately independent. Let us summarize: we have obtained that in the thermodynamic limit (8.46) is $\mathcal{N}(0,\sigma^2)$, that (8.47) is $\mathcal{N}(0,\tau_{t-1}^2/\mu)$, and that they are independent. Thus their sum is $\mathcal{N}(0,\sigma^2+\tau_{t-1}^2/\mu)$ and the first argument of the thresholding function (8.45) tends to the random variable

$$x+u\sqrt{\sigma^2+\frac{\tau_{t-1}^2}{\mu}} \tag{8.48}$$

where $u\sim\mathcal{N}(0,1)$ and $x\sim p_0(\cdot)$ as announced.

It remains to show point (ii). Using the second equation in (8.42) and expanding the Euclidean norm,

$$\begin{aligned}\frac{\alpha^2}{m}\|\underline{r}\|_2^2&=\frac{\alpha^2}{m}\sum_{b=1}^{m}\left(z_b+\frac{1}{\sqrt{m}}\sum_{i=1}^{n}A_{bi}^t(x_i-\widehat{x}_i^{t-1})\right)^2\\&=\frac{\alpha^2}{m}\sum_{b=1}^{m}z_b^2+\frac{2\alpha^2}{m^{3/2}}\sum_{b=1}^{m}\sum_{i=1}^{n}z_b\tilde{A}_{bi}^t(x_i-\widehat{x}_i^{t-1})\\&\quad+\frac{\alpha^2}{m}\sum_{b=1}^{m}\sum_{i=1}^{n}\sum_{j=1}^{n}\tilde{A}_{bi}^t\tilde{A}_{bj}^t(x_i-\widehat{x}_i^{t-1})(x_j-\widehat{x}_j^{t-1})\end{aligned}$$

Clearly the first term tends to $\alpha^2\sigma^2$. By similar arguments as in point (i) the second term can be shown to tend to zero and the third term to $(\alpha^2/\mu)\tau_{t-1}^2$. Thus in the thermodynamic limit

$$\frac{\alpha}{\sqrt{m}}\|\underline{r}^t\|_2\to\alpha\sqrt{\sigma^2+\frac{\tau_{t-1}^2}{\mu}}. \tag{8.49}$$

as announced.

8.7 Performance of AMP

In this section we derive the phase diagram of AMP in the plane of parameters (κ,μ). Recall $\kappa=k/n$ is the fraction of non-zero components in the signal and $\mu=m/n$ the fraction of measurements.[7]

The phase diagram is deduced from a study of SE updates (8.44), so the first

[7] It is also common in the literature to parametrize the phase diagram in terms of (ρ,μ) where $\rho=k/m=\kappa/\mu$, but then the transition lines look more complicated.

question we should address is to determine the multiplicity of solutions of the corresponding fixed point equation

$$\tau^2 = \int dx p_0(x) \int du \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \left\{ \eta\left(x + u\sqrt{\sigma^2 + \frac{\tau^2}{\mu}}\,;\, \alpha\sqrt{\sigma^2 + \frac{\tau^2}{\mu}} \right) - x \right\}^2 . \quad (8.50)$$

It is the subject of an exercise to show that this equation has a *unique solution* $\tau_*^2(\kappa, \mu, \alpha, p_0, \sigma)$ in the extended real half-line $[\sigma^2, +\infty]$. Therefore SE iterations will tend to this fixed point solution.

It is useful to note for further use the following property

$$\tau_*^2(\kappa, \mu, \alpha, p_0, \sigma) = \sigma^2 \tau_*^2(\kappa, \mu, \alpha, p_0^\sigma, 1), \quad (8.51)$$

where $p_0^\sigma(x) = \sigma p_0(\sigma x) = (1-\kappa)\delta(x) + \sigma p_0(\sigma x)$. To prove (8.51) we set $\tau = \sigma\tau'$ and notice that τ' satisfies the fixed point equation (8.50) with σ and p_0 replaced by 1 and p_0^σ respectively. To see this last point one makes the change of variables $x \to \sigma x$ and uses $\eta(\sigma y\,;\, \sigma\lambda) = \sigma\eta(y\,;\,\lambda)$. We already remarked that $p_0^\sigma \in \mathcal{S}_\kappa$ if $p_0 \in \mathcal{S}_\kappa$, in other word the class of distributions $\mathcal{S}_\kappa$ is scale invariant. This scale invariance property played a crucial role in the scalar case, and not surprisingly we will shortly see that it is also fundamental in the vector case.

Minimax Criterion and noise sensitivity phase transition

We have to make a suitable choice for the parameter α of the AMP algorithm (8.39). Recall, since p_0 is unknown, we must choose the best possible α given the worst possible $p_0 \in \mathcal{S}_\kappa$. Formally we have to compute the minimax-MSE of AMP

$$\inf_{\alpha \geq 0} \sup_{p_0 \in \mathcal{S}_\kappa} {\tau_*}^2(\kappa, \mu, \alpha, p_0, \sigma). \quad (8.52)$$

Using (8.51) and the scale invariance of $\mathcal{S}_\kappa$ we find

$$\begin{aligned} \inf_{\alpha \geq 0} \sup_{p_0 \in \mathcal{S}_\kappa} {\tau_*}^2(\kappa, \mu, \alpha, p_0, \sigma) &= \sigma^2 \inf_{\alpha \geq 0} \sup_{p_0 \in \mathcal{S}_\kappa} {\tau_*}^2(\mu, \rho, \alpha, p_0^\sigma, 1) \\ &= \sigma^2 \inf_{\alpha \geq 0} \sup_{p_0 \in \mathcal{S}_\kappa} {\tau_*}^2(\mu, \rho, \alpha, p_0, 1) \\ &\equiv \sigma^2 M(\kappa, \mu). \end{aligned} \quad (8.53)$$

The quantity $M(\kappa, \mu)$ is the rate of change of the minimax-MSE of AMP under small variations of the noise. It has been called the *noise sensitivity* in the literature.[8].

Remarkably, the noise sensitivity is *independent* of the level of noise and a look at the derivation above shows this is due to the *scale invariance* of the class of

[8] In statistical physics language one would call it a "response function" or a "susceptibility". This is a measure of the response of the state of the system to an external perturbation. Here the external perturbation is the variation of the noise.

sparse distributions. It turns out that scale invariance has more consequences, for example, it allows to derive an explicit formula for the noise sensitivity

$$M(\kappa,\mu) = \begin{cases} \frac{M_{\text{scalar}}(\kappa)}{1-\frac{1}{\mu}M_{\text{scalar}}(\kappa)} & \mu > M_{\text{scalar}}(\kappa) \\ +\infty & \mu < M_{\text{scalar}}(\kappa), \end{cases} \tag{8.54}$$

where $M_{\text{scalar}}(\kappa)$ is given by (8.8). Moreover the saddle point $(p_{0,\text{worst}}, \alpha_{\text{best}})$ is the same as the one for the scalar problem in Section 8.1. Figure 8.2 shows the phase diagram of the AMP algorithm. The curve $\mu = M_{\text{scalar}}(\kappa)$ is an algorithmic threshold line, which separates the (κ,μ) plane in two regions. Below the curve the measurement fraction is too small and the noise sensitivity (as well as minimax-MSE) are infinite. There is no hope to recover the sparse signal with the AMP estimate. Above the curve, the measurement fraction is large enough so that we can recover the signal with finite error.

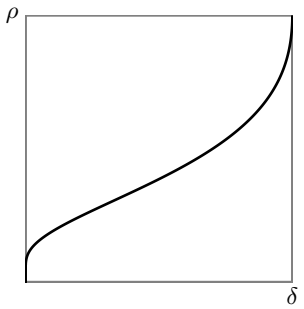


Figure 8.2 Left: the algorithmic noise sensitivity phase transition line in the (κ,μ) plane. Right: the same line in the (μ,ρ) plane.

We point out that the noise sensitivity phase transition line has a rather explicit parametrized form whose derivation is the subject of an exercise.

$$\begin{cases} \mu = \dfrac{2\frac{e^{-\frac{\alpha^2}{2}}}{\sqrt{2\pi}}}{\alpha+2\left(\frac{e^{-\frac{\alpha^2}{2}}}{\sqrt{2\pi}}-\alpha\Phi(-\alpha)\right)} \\ \kappa = \dfrac{2\left(\frac{e^{-\frac{\alpha^2}{2}}}{\sqrt{2\pi}}-\alpha\Phi(-\alpha)\right)}{\alpha+2\left(\frac{e^{-\frac{\alpha^2}{2}}}{\sqrt{2\pi}}-\alpha\Phi(-\alpha)\right)}. \end{cases} \tag{8.55}$$

Derivation of (8.54)

The starting point is again a scaling argument applied to the fixed point equation (8.50). With the change of variables $x \to x\sqrt{\sigma^2+\frac{\tau^2}{\mu}}$ we obtain

$$\tau^2 = (\sigma^2 + \frac{\tau^2}{\mu}) M_{\text{scalar}}(\kappa,\alpha,p_0^\tau) \tag{8.56}$$

with

$$M_{\text{scalar}}(\kappa, \alpha, p_0^\tau) \equiv \int dx p_0^\tau(x) \int du \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \left\{\eta(x+u\,,\,\alpha) - x\right\}^2 \tag{8.57}$$

and $p_0^\tau(x) = \sqrt{\sigma^2 + \frac{\tau^2}{\mu}} p_0(x\sqrt{\sigma^2 + \frac{\tau^2}{\mu}})$. Looking back at the solution of the LASSO for the scalar problem we see that $M_{\text{scalar}}(\kappa, \alpha, p_0^\tau)$ is noting else than the *scalar* MSE for a scaled signal distribution p_0^τ and a noise level $\sigma^2 = 1$. Remark also that scale invariance of $\mathcal{S}_\kappa$ implies

$$\sup_{p_0 \in \mathcal{S}_\kappa} M_{\text{scalar}}(\kappa, \alpha, p_0^\tau) = \sup_{p_0 \in \mathcal{S}_\kappa} M_{\text{scalar}}(\kappa, \alpha, p_0) = M_{\text{scalar}}(\kappa, \alpha) \tag{8.58}$$

where the supremum is attained for $p_{0\,\text{worst}}$.

Suppose the parameters are such that $M_{\text{scalar}}(\kappa, \alpha) > \mu$. Then replacing p_0 by $p_{0,\text{worst}}$ in (8.56) we find that the only solution is $\tau_*(\kappa, \mu, \alpha, p_{0,\text{worst}}, \sigma) = +\infty$. Therefore we necessarily have $\sup_{p_0 \in \mathcal{S}_\kappa} \tau_* = +\infty$ when $M_{\text{scalar}}(\kappa, \alpha) > \mu$. On the other hand if $M_{\text{scalar}}(\kappa, \alpha) < \mu$ we also have $M_{\text{scalar}}(\kappa, \alpha, p_0^\tau) < \mu$ and Equ. (8.56) has a finite solution,

$$\tau_*^2 = \sigma^2 \frac{M_{\text{scalar}}(\kappa, \alpha, p_0^{\tau_*})}{1 - \frac{1}{\mu} M_{\text{scalar}}(\kappa, \alpha, p_0^{\tau_*})} \tag{8.59}$$

This ratio is an increasing function of $M_{\text{scalar}}(\kappa, \alpha, p_0^{\tau_*})$ so it also follows that for $M_{\text{scalar}}(\kappa, \alpha) < \mu$

$$\sup_{p_0 \in \mathcal{S}_\kappa} \tau_*^2 = \sigma^2 \frac{M_{\text{scalar}}(\kappa, \alpha)}{1 - \frac{1}{\mu} M_{\text{scalar}}(\kappa, \alpha)}. \tag{8.60}$$

Now it remains to minimise over α. Recall $\inf_{\alpha > 0} M_{\text{scalar}}(\kappa, \alpha) = M_{\text{scalar}}(\kappa)$. So when α varies over the positive real line, $M_{\text{scalar}}(\kappa, \alpha)$ varies over $[M_{\text{scalar}}(\kappa), +\infty]$. Since the ratio in (8.60) is an increasing function of $M_{\text{scalar}}(\kappa, \alpha)$ which diverges at $M_{\text{scalar}}(\kappa, \alpha) = \mu$ (and remains infinite thereafter), its minimum is attained at $M_{\text{scalar}}(\kappa)$ when $M_{\text{scalar}}(\kappa) < \mu$ and at $+\infty$ when $M_{\text{scalar}}(\kappa) > \mu$. This is precisely the statement of (8.54).

8.8 Relation between AMP and solution of LASSO

We wish to revisit here a few issues that have been swept under the rug. We started by formulating a minimization problem (8.1) which yields the LASSO. We cannot a priori solve this problem analytically (except for the scalar case) so we settled for a min-sum approach. After several natural approximations of the min-sum equations, we where led to the AMP algorithm (8.37) which gives an estimate of the signal parametrized by λ. We switched to a variant of this algorithm, the AMP algorithm (8.39) which gives an estimate parametrized by α instead. The reason for introducing this variant is that its performance can be neatly analyzed thanks to state evolution.

This approach raises two natural questions. First, what is the relation between the two variants of AMP and how do their performance compare? In particular, do they have the same algorithmic phase transition line? Second, what is the relation between the true LASSO and AMP estimate? The first question is a purely algorithmic one, whereas the second really belongs to the third part of the course where we discuss the relations between message passing algorithms and optimal solutions. In the present case we can quite simply obtain at least partial answers which are worth stating immediately.

The Hamiltonian (8.9) is a convex function of $\underline{x} \in \mathbb{R}^n$, so the minima are solutions of the stationnarity condition

$$A^T(\underline{y} - A\underline{x}) = \lambda \underline{v} \tag{8.61}$$

where $v_i = \text{sign}(x_i)$ for $x_i \neq 0$ and $v_i \in [-1, +1]$ for $x_i = 0$.

Take the AMP iterations (8.37) and consider a fixed point $(\widehat{\underline{x}}^*, \underline{r}^*, \theta_*)$. One can see that $\underline{x}^*$ satisfies (8.61) provided we take equation $\lambda = \theta_*(1 - \frac{\|\widehat{\underline{x}}^*\|_0}{m})$. This is also a condition that the fixed point of AMP iterations (8.37) must satisfy. So we conclude that, for any fixed λ, when the updates converge to a fixed point, this fixed point is also a solution of the LASSO minimization problem.

Consider now the α-AMP update equations (8.37), (8.38) and a corresponding fixed point $(\widehat{\underline{x}}^*, \underline{r}^*)$. This time $\underline{x}^*$ satisfies (8.61) provided we take $\lambda = \alpha \frac{\|r\|_2}{\sqrt{m}}(1 - \frac{\|\widehat{\underline{x}_*}\|_0}{m})$. Using the analysis of Section 8.6 (specifically (8.48) and (8.49)) this relation becomes for $m \to +\infty$

$$\begin{aligned}\lambda(\alpha) =& \alpha\sqrt{\sigma^2 + \tau_*^2}\Big\{1 - \\ & \mu^{-1}\int dx p_0(x)\int du \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}}\left[\eta'(x + u\sqrt{\sigma^2 + \tau_*^2}; \alpha\sqrt{\sigma^2 + \tau_*^2})\right]\Big\}.\end{aligned}$$

We conclude that when they converge the α-AMP and $\lambda(\alpha)$-AMP algorithms converge to the same fixed point, and this fixed point is a solution of the LASSO minimization problem.

The two variants of AMP are equivalent in terms of performance in the large size limit. In particular the noise sensitivity phase transition line is the same.

8.9 Approximate Message Passing for the MMSE estimator

Even if this is perhaps a less realistic situation, it is instructive to consider the case of a signal with *known* prior distribution from the class $\mathcal{S}_\kappa$. In other words $p_0(x) = (1 - \kappa)\delta_0(x) + \kappa\phi_0(x)$ for a known $\phi_0(x)$. A good example to keep in mind is a Gaussian distribution $\phi_0(x) = e^{-x^2/2}/\sqrt{2\pi}$; one then refers to $p_0(x)$ as the Bernoulli-Gauss model.

As explained in Chapter 3, in this setting the optimal estimator is the MMSE estimator (3.35). Since we cannot a priori hope to compute it exactly we resort to a message passing calculation. In Chapter 5 we went through the BP

FIGURE

Figure 8.3 The denoiser $\eta_0(y\,;\,\nu)$ for the Bernoulli-Gauss model.

equations in Example 16, and this approach can be systematically developed in order to recursively compute the BP-estimate for the signal. The complexity of the message passing step is again quadratic because the factor graph is bipartite complete; but following the same route as in Sections 8.3 and 8.4, the message-passing equations can be simplified in order to arrive at algorithm that is very similar to (8.39). Instead of embarking in this lengthy route, one can make an educated guess of the form of the new algorithm, just by skimming through the previous results.

In Section 8.5 the AMP algorithm uses the soft thresholding function $\eta(y\,;\,\lambda)$ found by solving the scalar LASSO problem. The reader should not be too surprised that now the updates will involve a *thresholding function given by the MMSE estimator of the scalar case*. Consider a scalar measurement $y = x + z$ of "signal" x affected by Gaussian noise with variance ν^2. The "softer" thresholding function is now

$$\eta_0(y\,;\,\nu) = \mathbb{E}[X|Y=y] = \frac{\int dx\, x\, p_0(x) e^{-\frac{(y-x)^2}{2\nu^2}}}{\int dx\, p_0(x) e^{-\frac{(y-x)^2}{2\nu^2}}}. \tag{8.62}$$

and is also called a *denoiser* (see Figure 8.3). We stress that, contrary to the case of LASSO, here $\eta_0(y\,;\,\nu)$ is not universal and depends on the prior. The corresponding mean square error is the MMSE function (by convention the argument of the MMSE function is a signal-to-noise-ratio, here ν^{-2})

$$\begin{aligned}\mathrm{mmse}(\nu^{-2}) &= \mathbb{E}\big[(X - \mathbb{E}[X|Y])^2\big] \\ &= \int dx\, p_0(x) \int dz\, \frac{e^{-\frac{z^2}{2\nu^2}}}{\sqrt{2\pi\nu^2}} \big\{x - \eta_0(x+z\,;\,\nu)\big\}^2.\end{aligned} \tag{8.63}$$

The AMP updates (for the vector case) are similar to (8.39)

$$\begin{cases} \widehat{x}_i^{t+1} = \eta_0(x_i^t + (A^T \underline{r}^t)_j\,;\,\nu_t), \\ r_a^t = y_a - (A\underline{\widehat{x}}^t)_a^{(t-1)} + b_t r_a^{t-1}. \end{cases} \tag{8.64}$$

with a number of differences that we now discuss. As already pointed out, naturally η_0 replaces η. The Onsager term is also different. In the derivations of Section 8.4 this term can be traced back to a derivative of the soft thresholding

function. We can therefore guess that now

$$b_t = \frac{1}{m}\sum_{i=1}^{n} \eta_0'\big(x_i^{t-1} + (A^T \underline{r}^t)_a \,;\, \nu_t\big). \tag{8.65}$$

Finally recall that for the AMP algorithm (8.39) we expressed in Section 8.6 the threshold level thanks to the MSE through the relation $\theta_t = \alpha\sqrt{\sigma^2 + \frac{\tau_t^2}{\mu}}$. Here the analysis leads to a similar conclusion, namely[9]

$$\nu_t = \sqrt{\sigma^2 + \frac{\tau_t^2}{\mu}}. \tag{8.66}$$

Note that the MMSE problem does not involve any parameter λ or α over which one should optimise. Note also that to run the AMP updates (8.64) one has to precompute τ_t. To do this one has to write down the corresponding SE equations.

The performance analysis follows the same steps than in Section 8.7. The result is a SE recursion with η_0 replacing η

$$\begin{aligned}\tau_{t+1}^2 &= \mathrm{mmse}((\sigma^2 + \frac{\tau_t^2}{\mu})^{-1}) \\ &= \int dx\, p_0(x) \int du\, \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} \left\{ \eta_0\left(x + u\sqrt{\sigma^2 + \frac{\tau_t^2}{\mu}} \,;\, \sqrt{\sigma^2 + \frac{\tau_t^2}{\mu}} \right) - x \right\}^2 .\end{aligned} \tag{8.67}$$

This equation has a nice interpretation: at time $t+1$ the mean square error τ_{t+1}^2 for the AMP estimate is given by the MMSE of a scalar signal (8.63) with effective noise variance $\sigma^2 + \frac{\tau_t^2}{\mu}$ at time t.

Let us summarize. Equation (8.67) gives the evolution of the mean square error of the AMP estimate. Equations (8.64), (8.65), (8.66) define the mmse-AMP algorithm, and allow to compute the estimates for the signal. Note that (8.66) is independent of the input signal and can be precomputed once for all.

We now turn our attention towards the phase diagram. As usual we must get a hold on the solutions of the fixed point equation associated to (8.67). Contrary to the LASSO case where only one solution exists, here the situation is more complicated and multiple solutions can appear. Moreover for the LASSO the solution could be determined rather explicitly because of scale invariance. In the present case there is *no* such scale invariance since $p_0(x)$ is a fixed distribution and typically brings in another scale besides the noise. But it is still possible to make qualitative statements that are valid for a fairly wide class of distributions. Moreover the phase transition line can precisely characterised in a simple manner. For the Bernoulli-Gauss model $\eta_0(y\,;\,s)$ can be explicitly be computed and all statements that follow fairly explicitly checked; this is the subject of an exercise.

[9] Where by definition $\tau_t^2 = \lim_{n\to+\infty} \frac{1}{n}\|\underline{\hat{x}}^t - \underline{x}\|_2$.

FIGURE

Figure 8.4 Phase diagram of mmse-AMP for the Bernoulli-Gauss distribution. The performance is better than LASSO since we exploit prior knowledge of the signal distribution.

Define

$$D(p_0) \equiv \sup_{\nu \geq 0} \left[\nu^{-2} \mathrm{mmse}(\nu^{-2}) \right] \tag{8.68}$$

Note that $\lim_{\nu \to 0} \nu^{-2} \mathrm{mmse}(\nu^{-2}) = \kappa$ so we always have $D(p_0) > \kappa$.

For a measurement rate $\mu > D(p_0)$ there exists only one fixed point solution called $\tau^2_{*,\mathrm{good}}$ such that the "noise sensitivity" $\lim_{\sigma \to 0}(\tau^2_{*,\mathrm{good}}/\sigma^2)$ remains finite. Thus for $\mu > D(p_0)$ the algorithm yields a correct reconstruction in the small noise limit $\sigma \to 0$ (and more generally a finite error for finite noise). Now, decrease the measurement rate in the range $D(p_0) < \mu < \kappa$. One finds two or more stable fixed points (as well as unstable ones) for all $\sigma^2 > 0$. Besides the "good" fixed point which satisfies $\tau^2_{*,\mathrm{good}} = \Theta(\sigma^2)$ there is a "bad" one, i.e. $\tau^2_{*,\mathrm{bad}} = \Theta(1)$.[10] Clearly, initializing iterations with $\tau_0^2 = +\infty$ one is driven to the largest stable fixed point i.e $\tau^2_{*,\mathrm{bad}}$. This means that the noise sensitivity $\lim_{\sigma \to 0} \tau^2_{*,\mathrm{bad}}/\sigma^2$ diverges, and exact reconstruction is not possible even for very small noise.

We therefore conclude that (8.68) is the algorithmic phase transition threshold of AMP for a known prior; a remarkably neat result! It has been called the *information dimension* since it represents the minimum fraction of measurements needed to reconstruct a signal exactly in the noiseless limit. Note that with this interpretation in mind the inequality $D(p_0) > \kappa$ now appears as trivial. This threshold is lower than the Donoho-Tanner curve derived in Section 8.6. This is not surprising since the later concerns the worst case distribution for $p_0 \in \mathcal{S}_\kappa$.

Figure 8.4 illustrates the phase diagram of AMP with known prior in the (κ, μ) plane for the Bernoulli-Gauss model and compares it with AMP for LASSO.

8.10 Notes

to do

Donoho Jonhson scalar case (1994) paper. Montanari et al min-sum to AMP; sate evolution. Paris group other but equivalent approach mmse-AMP. alpha-lambda calibration map. A word about convex optimisation for lasso, and gradient descent, and IST.

[10] When there are more than two stable fixed points we define it as the maximal one.

Problems

8.1 A generalization of IST and its connection to LASSO. The standard Iterative Soft Thresholding algorithm has the form

$$\begin{cases} x_i^{t+1} = \eta(x_i^t + (A^T \underline{r}^t)_i \,;\, \lambda) \\ \underline{r}^t = \underline{y} - A\underline{x}^t \end{cases}$$

starting from the initial condition $x_i^0 = 0$. Consider the following generalization. Let θ_t and b_t be two sequences of scalars (called respectively "thresholds" and "reaction terms") that converge to fixed numbers θ and b. Construct the sequence of estimates according to the iterations

$$\begin{cases} x_i^{t+1} = \eta(x_i^t + (A^T \underline{r}^t)_i \,;\, \theta_t) \\ \underline{r}^t = \underline{y} - A\underline{x}^t + b_t r^{t-1} \end{cases}$$

The goal of the exercise is to prove the following claim. If x^*, r^* is a fixed point of these iterations, then x^* is a stationary point of the LASSO cost function $\mathcal{H}(\underline{x}) = \frac{1}{2}||\underline{y} - A\underline{x}||_2^2 + \lambda||x||_1$ for $\lambda = \theta(1-b)$.

Note that this claim does not say how to specify suitable sequences b_t and θ_t. The point of AMP is that it specifies unambiguously that one should take $b_t = ||x||_0/m$ (for θ_t there is more flexibility). The proof proceeds in three steps.
(i) Show that the stationarity condition for the LASSO cost function is

$$A^T(\underline{y} - A\underline{x}^*) = \lambda \underline{v},$$

where $v_i = \text{sign}(x_i^*)$ for $x_i^* \neq 0$ and $v_i \in [-1, +1]$ for $x_i^* = 0$.
(ii) Show that the fixed point equations corresponding to the iterations above are

$$\begin{cases} x_i^* + \theta v_i & = x_i^* + (A^T \underline{r}^*)_i \\ (1-b)\underline{r}^* & = \underline{y} - A\underline{x}^* \end{cases}$$

(iii) Remark that these two steps imply $\lambda = \theta(1-b)$.

8.2 Statistics of AMP and IST unthresholded estimates. Consider a sparse signal $\underline{x}_0$ with n iid components distributed as

$$(1-\kappa)\delta(x_0) + \frac{\kappa}{2}\delta(x_0 - 1) + \frac{\kappa}{2}\delta(x_0 + 1).$$

Generate m noisy measurements $\underline{y} = \frac{1}{\sqrt{m}}\tilde{A}\underline{x}_0 + \underline{z}$ where $\tilde{A}_{ai} = \pm 1$ are Bernoulli(1/2) and z_a are iid Gaussian zero mean and variance σ^2. Consider the AMP iterations (8.37) with the choice $\theta^t = \alpha\|\underline{r}^t\|_2/\sqrt{m}$. The derivation of state evolution rests on the assumption that the i-th component of the unthresholded estimate

$$\widehat{x}_i^t + \frac{1}{\sqrt{m}}\sum_{b=1}^{m} \tilde{A}_{bi} r_b^t,$$

(given $\underline{x}_0$) has Gaussian statistics. The mean is x_{0i} and the variance $\sigma^2 + \tilde{\tau}^2$ where $\tilde{\tau}^2 = \|\underline{x}^t - \underline{x}_0\|_2^2/n$.

Perform an experiment to check this numerically. Compute also the statistics of the un-thresholded estimate for the IST iterations, i.e. when the Onsager term $r_a^{t-1}\frac{\|\widehat{x}^t\|_0}{m}$ is removed. Compare the two histograms. Indications: Fix a signal realization $\underline{x}_0$. Try $n = 4000$, $m = 2000$, $\kappa = 0.125$ and 40 instances for A and $\underline{z}$. Try various values for σ and α. Look at the i-th components of the un-thresholded estimate for components such that say $x_{0i} = +1$ (or -1, or 0).

8.3 Unicity of solution of SE fixed point equation. Consider the SE fixed point equation (8.50). Show that there is a unique fixed point solution in $[\sigma^2, +\infty]$ (the value $+\infty$ included). Hint: write the fixed point equation for the new variable $\tilde{\tau}^2 = \sigma^2 + \tau^2/\mu$ in the form $\tilde{\tau}^2 = F(\tilde{\tau})$ and show that F is a concave function of $\tilde{\tau}$. Proceed graphically.

8.4 Noise sensitivity phase transition. Derive the parametrised form (8.55) of the noise sensitivity threshold line.

8.5 AMP for a known prior. Give the details of the derivation of the mmse-AMP algorithm (8.64), (8.65), (8.66) and those of the corresponding state evolution equation (8.67).

8.6 Bernoulli-Gauss model. Consider the prior

$$p_0(x) = (1-\kappa)\delta(x) + \kappa\frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}}.$$

(i) Compute the denoiser (8.62) and check,

$$\eta_0(y\,;\,\nu) = \frac{y}{1+\nu^2}\,\frac{\kappa\frac{e^{-\frac{y^2}{2(1+\nu^2)}}}{\sqrt{2\pi(1+\nu^2)}}}{\kappa\frac{e^{-\frac{y^2}{2(1+\nu^2)}}}{\sqrt{1+\nu^2}} + (1-\kappa)\frac{e^{-\frac{y^2}{2\nu^2}}}{\sqrt{\nu^2}}}.$$

(ii) Check also from (8.63)

$$\mathrm{mmse}(\nu^{-2}) = \kappa - \frac{\kappa}{1+\tau^2}\int_{-\infty}^{+\infty} dy\, y^2 \frac{\frac{e^{-\frac{y^2}{2}}}{\sqrt{2\pi}}}{1+\frac{1-\kappa}{\kappa}\sqrt{\frac{1+\tau^2}{\tau^2}}e^{-\frac{y^2}{2\tau^2}}}$$

(iii) Show that $\lim_{\nu\to 0}\nu^{-2}\mathrm{mmse}(\nu^{-2}) = 0$. This implies $D(p_0) \geq \epsilon$.

(iv) Finally analyse the solutions of the mmse-AMP fixed point equation when the undersampling rate satisfies $\mu > D(p_0)$ and $D(p_0) < \mu < \kappa$. Plot the phase transition line $\tilde{\mu}(p_0)$ and compare with the LASSO phase transition line.

9 Random K-SAT: Introduction to Decimation Algorithms

The satisfiability problem is considerably more difficult to analyze than either coding or compressive sensing. One reason for this difficulty is that it is not an inference problem. Indeed, in the regime where a random formula is satisfiable with high probability (i.e., when the number of clauses per Boolean variable is sufficiently small) there are exponentially many solutions contrary to coding or compressive sensing where we typically only have one valid solution. At first we might guess that this makes the problem easy: we are not asking for a particular solution – any solution will do. But in fact it is exactly this lack of uniqueness which makes the problem hard.

Why does this non-uniqueness cause trouble? Pick a specific Boolean variable. From the perspective of this variable there are typically solutions for which this variable takes on the value 0 but also solutions for which it takes on the value 1. In fact, of the exponentially many solutions there are typically roughly equally many of either type. So even if the message-passing algorithm succeeded in computing the marginals of all bits correctly (here we assume that we put a uniform measure on all solutions and compute the marginal with respect to this measure) all these marginals would be roughly uniform and we cannot extract from them a globally valid solution. Therefore a straightforward application of a message-passing algorithm does not work. A new ingredient is needed.

One approach is quite natural given the above description. Assume for a moment that message-passing is capable of accurately computing marginals (or if you prefer assume that they are given by an "oracle"). Then we can proceed as follows. Compute the marginal for one variable. As long as this marginal does not put all mass on either 0 or 1 it means that there are solutions which take on the value 0 as well as solutions which take on the value 1 for this variable. So in this case choose any value for this variable, and reduce the formula by eliminating this variable and all clauses which are now satisfied. This reduction is called the *decimation* step. If the marginal has all its mass on 0, then pick the value 0, and if it has all its mass on 1 then choose the value 1. Again, decimate. It is clear that this procedure succeeds in finding a satisfiable formula if one exists.

The above description assumed that message-passing is capable of exactly computing the marginals (or that we have an oracle). Since this is generally not the case, we proceed slightly differently. Estimate marginals for all variables thanks to a message passing algorithm, say Belief propagation. Then pick a

variable with maximal bias and decimate according to this bias. The hope is that by picking a variable with maximal bias we minimize the chance of making a mistake. This will be true as long as the message-passing algorithm predicts the marginals with reasonable accuracy. The above idea is what is used in *Belief Propagation Guided Decimation* (BPGD). We will talk in more detail about this algorithm in the present chapter. In Chapter 17 when we shall have more concepts and tools at our disposal, we develop an upgraded version of BPGD called *Survey Propagation Guided Decimation* (SPGD).

Unfortunately, currently there does not exist a rigorous analysis for BPGD. It is instructive to first consider the much simpler *Unit Clause Propagation* (UCP) algorithm and show how to analyse it rigorously. Unit clause propagation is a decimation algorithm where we do not decimate according to a good estimate of the marginals but according to a much simpler rule. As long as degree one clauses are present we satisfy them (this is also what belief propagation would "tell us"), while if degree one clauses are not present we select a variable at random and set its value at random. This sort of algorithm has a somewhat mediocre performance, i.e., the threshold $\alpha_{\mathrm{UCP}}(K)$ up to which it works is much below the actual satisfiability threshold $\alpha_s(K)$. But it is relatively easy to analyze and it will give us the excuse of introducing a very powerful general machinery of analyzing such types of graph processes, called the Wormald method. This is our starting point in the next section.

9.1 Analysis of a stochastic process by differential equations

Simple algorithms can often be formulated in terms of a stochastic process and if the state space is sufficiently simple the progress of the algorithm can often be analyzed in terms of a system of differential equations. Here we give an elementary introduction to this method via a very simple toy example. We first treat this example formally and then discuss the Wormald theorem which allows to make the analysis rigorous. Although the Wormald theorem is rather long to state, it has a general applicability and it is often not very hard to verify the hypothesis.

A toy example

Consider n particles in a box of volume V. Think of n and V as large with the initial density of particles $\rho = n/V$ fixed as $n, V \to +\infty$. These particles can annihilate each other according to a simple model. Assume that time is discrete and takes integer values. At each time instant and for each pair of particles (i, j) present, the probability that these two particles annihilate each other is equal to $1/V^2$. How will the number of particles evolve? Let $N(t)$ denote the number of particles which are left at time t, with $N(0) = n$. This is a stochastic process

described by its current state $N(t)$. We have the relationship

$$N(t+1) = N(t) - 2 \sum_{1 \le i < j \le n} \mathbb{1}((i,j) \text{ is annihilated between } t \text{ and } t+1). \quad (9.1)$$

It is easy to write down the expected progress in one time step given the current state $N(t)$ by taking the conditional expectation of (9.1). Using linearity of the expectation, the probability $1/V$ of annihilation, and that there are $N(t)(N(t)-1)/2$ pairs, we obtain

$$\begin{aligned} \mathbb{E}[N(t+1) \mid N(t)] &= N(t) - 2\frac{N(t)(N(t)-1))}{2}\frac{1}{V^2} \\ &= N(t) - \rho^2 \frac{N(t)(N(t)-1))}{n^2}. \end{aligned}$$

This means[1]

$$\mathbb{E}[N(t+1) - N(t) \mid N(t)] = -\rho^2 \frac{N(t)^2}{n^2} + O\big(\frac{1}{n}\big) \quad (9.2)$$

As long as the number of remaining particles is large one may hope that $N(t)$ concentrates on its expectation. If this is the case we can drop the expectation. Wormald's theorem essentially makes this step rigorous. Dropping the expectation in (9.2) and setting $N(t) = nz(t/n)$ we have

$$nz\big(\frac{t}{n} + \frac{1}{n}\big) - nz\big(\frac{t}{n}\big) = -\rho^2 z\big(\frac{t}{n}\big)^2 + O\big(\frac{1}{n}\big) \quad (9.3)$$

The natural time scale is $\tau = t/n$ and for $n \gg 1$ we are lead to consider the differential equation

$$\frac{dz(\tau)}{d\tau} = -\rho^2 z(\tau)^2 \quad (9.4)$$

for the deterministic time evolution of the average particle density. One easily verifies that with the initial condition $z(0) = 1$ (i.e., $N(0) = n$) the solution is $z(\tau) = 1/(\tau\rho^2 + 1)$. If we undo the scalings we see that according to this model the expected number of remaining particles evolves as $n^2/(t\rho^2 + n)$. In this derivation we have replaced the stochastic process (9.1) by a deterministic description (9.4). One might hope that the behaviour of specific instances of $N(t)$ are close to the deterministic solution, at least as long as $N(t)$ is large. Wormald's theorem gives general conditions under which this is indeed correct.

The Wormald Theorem

There are myriads of versions of increasing sophistication. We will be content with stating and applying one particular incarnation.

THEOREM 9.1 *Let $Y_i^{(n)}(t)$ be a sequence (indexed by n) of real valued discrete time random processes, $1 \le i \le k$, where k is fixed, and so that for all $1 \le$*

[1] To get the $O(1/n)$ term one uses $N(t)/n^2 \le N(0)/n^2 \le 1/n$.

$i \le k$, all $0 \le t < m^{(n)}$, and all $n \in \mathbb{N}$ $|Y_i^{(n)}(t)| \le Bn$ for some constant B. Let $I \subset \mathbb{R}^k$ defined as $I = \{(y_1, \dots, y_k) : \mathbb{P}\{Y_i^{(n)}(t) = ny_i, 1 \le i \le k\} > 0,$ for some $n\}$ and let $D \subset \mathbb{R}^{k+1}$ be some open connected bounded set containing the closure of $\{(0, y_1, \cdots, y_k) : (y_1, \cdots, y_k) \in I\}$. Denote by $H(t)$ the history of the processes up to time t, i.e., $H(t) = \{\underline{Y}^{(n)}(0), \underline{Y}^{(n)}(1), \dots, \underline{Y}^{(n)}(t)\}$ where $\underline{Y}^{(n)} = (Y_1^{(n)}, \dots, Y_k^{(n)})$.

Suppose there are functions $f_i : \mathbb{R}^{k+1} \to \mathbb{R}$, $1 \le i \le k$ such that:

1. *[Trend] For all i and uniformly for all $t < t_D^{(n)}$*

$$\mathbb{E}[Y_i(t+1) - Y_i(t) \mid H(t)] = f_i(\frac{t}{n}, \frac{Y_1^{(n)}(t)}{n}, \cdots, \frac{Y_k^{(n)}(t)}{n}) + o(1).$$

2. *[Tail] For all i and uniformly for all $t < t_D^{(n)}$*

$$\Pr(|Y_i^{(n)}(t+1) - Y_i^{(n)}(t)| > n^{1/5} \mid H(t)) = o(n^{-3}).$$

3. *[Regularity] For each i, the function f_i is a Lipschitz continuous on D.*

Then we have:

a. *[Differential equation] For $(0, \widehat{z}_1, \cdots, \widehat{z}_k) \in D$ the system of differential equations*

$$\frac{dz_i}{d\tau} = f_i(\tau, z_1, \cdots, z_k), \;\; 1 \le i \le k,$$

has a unique solution in D with initial condition $z_i(0) = \widehat{z}_i$, $1 \le i \le k$. This solution extends to points arbitrarily close to the boundary of D.

b. *[Concentration] Almost surely*

$$Y_i^{(n)}(t) = nz_i\big(\frac{t}{n}\big) + o(n),$$

uniformly for $0 \le t \le \min\{t_D^{(n)}, n\tau_{max}\}$ and for each i, where $z_i(\tau)$ is the solution in (a) with $\hat{z}_i(0) = Y_i^{(n)}(0)/n$ and where τ_{max} is the maximum time until the solution can be extended before reaching ϵ-close to the boundary of D where ϵ is arbitrary but strictly positive.

In our simple toy example the stochastic process is integer valued $Y^{(n)}(t) = N(t)$ (it is indexed by the initial number of particles n). Obviously there is a "trend" governed by the function $f(t/n, N(t)/n) = -\rho^2 N(t)^2/n^2$ in (9.2). The "tail" condition essentially states that at least for some time the probability that nearly all remaining particles annihilate at once is small. Obviously the square function is Lipshitz and the "regularity" condition is satisfied. This allows to conclude that almost surely $N(t) = nz(t)$ for t finite with respect to n, where $z(t)$ is the solution of (9.4). We leave it as an exercise to check in more detail that all conditions of the theorem are satisfied.

9.2 The Unit-Clause Propagation Algorithm

Suppose that we have an algorithm that allows to find solutions of random K-SAT formulas with uniformly positive probability (uniformly with respect to the size n of the formulas) for some range of densities, say $\alpha < \alpha_{\rm alg}(K)$. Then invoking the threshold behavior (1.12) guaranteed by Friedgut's theorem we conclude that the formula is almost surely satisfiable for $\alpha < \alpha_{\rm alg}(K)$. We also get an *algorithmic lower bound* on the satisfiability threshold, $\alpha_{\rm alg}(K) < \alpha_s(K)$. The challenge is to find an algorithm that is sufficiently simple to analyze rigorously and at the same time finds solutions (with strictly positive probability) in a "decent" range of densities.

The unit clause propagation (UCP) algorithm is a simple and important paradigm among a class of similar algorithms that provably find solutions with positive probability. The analysis of these algorithms is based on the differential equation method. Variants of UCP are discussed in the exercises.

Exercise with variant of simple UCP? The class is teh one of unitary algos where one takes the unit clause step and has a more complicated rule for the free step

Let us briefly recall the setting and notations of Chapter 1. We have n Boolean variables $x_i \in \{0,1\}$ out of which we can construct $2^K\binom{n}{K}$ clauses (disjunctions) containing K variables, where each variable enters as x_i or $\bar{x}_i$ (the negation of x_i). A random formula from the ensemble $\mathcal{F}(n,m,K)$ is sampled by taking m clauses uniformly at random with replacement. We will often think of the number of variables contained in a clause as the *length of a clause*. In particular, initially all clauses have length K and each variable entering in a clause is negated with probability $1/2$.

UCP algorithm

The algorithm is stated in table 1. Here is an informal description.

The algorithm sets the value of one variable at a time according to a rule to be specified and reduces the formula. The clauses that are satisfied by setting the variable are removed from the formula. Those that are not satisfied are shortened, which simply means the variable is removed from the formula. Initially all clauses have length K. As the algorithm proceeds some clauses disappear and others become shorter. Once a variable is fixed, the value stays fixed and is never changed (the algorithm never backtracks). If eventually all clauses disappear, the formula is satisfied and we have found a solution. If a clause of length zero appears then we have failed (indeed a clause of length zero appears it means none of its variables satisfied it).

We still have to specify a rule to set variables one at a time. The rule is the simplest possible and takes advantage of clauses of length one, called *unit clauses*. As long as there are unit clauses present in the formula we set its unique variable to the value that satisfies the unit clause. Note that once a unit clause appears we are anyway forced to do so if we want to satisfy the formula (since we do not backtrack). When there are no unit clauses we simply pick a random a variable at random and set its value at random.

Algorithm 1: Unit Clause Propagation algorithm

1. As long as no zero-length clause appears, iterate the two steps:
   - (Forced step) if the formula contains unit clauses choose one and satisfy it. Reduce the formula.
   - (Free step) if the formula does not contain unit clauses, choose a variable at random and set its value at random. reduce the formula.
2. If a zero-length clause appears output "fail".
3. If no zero-length clause has appeared after n steps output a "solution".

We discuss the application of Wormald's theorem for the analysis of UCP with $K = 3$. Through this analysis one finds the algorithmic threshold $\alpha_{\rm UCP}(3) = 8/3 \approx 2.666$ (compare with the satisfiability threshold $\alpha_s(3) \approx 4.26$). The generalization to any $K \geq 3$ is found in the exercises. We define 'time" t as the number of steps, i.e., the number of variables that have been eliminated. At time t the remaining formula has $n-t$ variables. The number of clauses of length $i = 1, 2, 3$ at time t is denoted $C_i(t)$. The state of the stochastic process associated with UCP is given by $\underline{C}(t) = (C_1(t), C_2(t), C_3(t))$. We do not explicitly indicate the superscript n that indexes the sequence of stochastic processes.

An crucial property of the UCP algorithm that makes the differential equation tractable is the *uniform randomness property*. This property means that at any time step t, given the state $\underline{C}(t)$, the formula belongs to the uniformly random ensemble constructed out of $n-t$ variables and $C_i(t)$ clauses of lengths $i = 1, 2, 3$. Here we only give an intuitive justification of this statement. At a free step we set a variable at random (and reduce the formula) so no information is revealed about the reduced formula. At a forced step the variable is not set at random because it has to satisfy a unit clause. However this unit clause itself is random (it contains the variable or its negation with probability 1/2) so from the point of view of the reduced formula this is equivalent to a free step. In both cases at each UCP step we get no information about the reduced formula which therefore remains uniformly random. In other words the reduced formula could as well have been generated "on the fly" at the current time t from the uniform ensemble with $n-t$ variables and $C_i(t)$ clauses of length $i = 1, 2, 3$. Generating formulas on the fly given the current state of the algorithm is sometimes called the *principle of deferred decisions*.

Differential equations

We first write down the set of "trend" equations. At any time t, a variable is chosen among the $n-t$ remaining ones and is set to some (permanent) value. This will destroy a certain number of clauses, either because they become satisfied or because they are shortened. Clauses of length 3 can only be destroyed. Clauses of length 2 can be destroyed, but also created from the shortening of 3-clauses.

Thus

$$\begin{cases} C_3(t+1) = C_3(t) - \sum_{3\text{-clauses}} \mathbb{1}(3\text{-clause} \ni \text{chosen variable}) \\ C_2(t+1) = C_2(t) - \sum_{2\text{-clauses}} \mathbb{1}(2\text{-clause} \ni \text{chosen variable}) \\ \qquad + \sum_{3\text{-clauses}} \mathbb{1}(3\text{-clause} \ni \text{chosen variable and is not satisfied}) \end{cases}$$

Now take the expectation conditioned on the current state. Because of the uniform randomness property this expectation is with respect to a uniform weight over the current formulas and it does not matter if the chosen variable is set in a free or a forced step. The number of clauses of length $i = 2, 3$ that contain the chosen variable has a binomial distribution $\mathrm{Bin}(C_i(t), i/(n-t))$. So in expectation there are $iC_i(t)/(n-t)$ clauses of length $i = 2, 3$ containing the chosen variable, and among them $iC_i(t)/2(n-t)$ are not satisfied. We obtain the "trend" equations

$$\begin{cases} \mathbb{E}[C_3(t+1) - C_3(t) | \underline{C}(t)] = -\frac{3C_3(t)}{n-t} = \frac{3C_3(t)/n}{1-t/n}. \\ \mathbb{E}[C_2(t+1) - C_2(t) | \underline{C}(t)] = -\frac{2C_2(t)}{(n-t)} + \frac{3C_3(t)}{2(n-t)} = -\frac{2C_2(t)/n}{1-t/n} + \frac{3C_3(t)/n}{2(1-t/n)}. \end{cases} \tag{9.5}$$

At this point we need to check that all the conditions of the Wormald theorem are fulfilled. Obviously the number of clauses is at all times smaller than αn. Also the initial condition is deterministic. Further, changes at each step are small with high probability, so the tail condition is also easily checked. The function giving the trend is Lipschitz for $\tau \in [0, 1[$. In conclusion according to the Wormald theorem almost surely $C_2(t) = nc_2(\tau)$, $C_3(t) = nc_3(\tau)$ with $\tau = t/n$, where $c_2(\tau)$ and $c_3(\tau)$ satisfy differential equations

$$\begin{cases} \frac{\mathrm{d}c_3(\tau)}{\mathrm{d}\tau} = -\frac{3c_3(\tau)}{1-\tau}, \\ \frac{\mathrm{d}c_2(\tau)}{\mathrm{d}\tau} = \frac{3c_3(\tau)}{2(1-\tau)} - \frac{2c_2(\tau)}{1-\tau}, \end{cases} \tag{9.6}$$

for any fixed time strictly bounded away from $\tau = 1$. The initial conditions are $c_3(0) = \alpha$, $c_2(0) = 0$.

It is straightforward to check that the solution of (9.6) is

$$c_3(\tau) = \alpha(1-\tau)^3, \qquad c_2(\tau) = \frac{3\alpha}{2}\tau(1-\tau)^2, \tag{9.7}$$

As $\tau \to 1$ the total number of clauses of length 2 and 3 becomes $o(n)$.

Unit clause process

The reader will have noticed that we did not write a differential equation for $C_1(t)$. Indeed as long as the algorithm does not fail $C_1(t) = O(1)$ and the changes at each time step are of the same order, so that the process $C_1(t)$ does *not* concentrate on the solution of a deterministic differential equation. Also, if at some time $C_1(t) = \Theta(n)$, the algorithm has already failed anyway.

The number of unit clauses is correctly described by a Galton-Walton branching process. The process starts with a free step which generates a cascade of unit clauses. When a variable is set (in a free or forced step) an average of $2C_2(t)/2(n-t)$ new unit clauses are born. Thus, using (9.7), the (birth) rate of the Galton-Walton process is

$$\rho_1(\tau) \equiv \frac{c_2(t)}{1-\tau} = \frac{3\alpha}{2}\tau(1-\tau). \tag{9.8}$$

The maximum of (9.8) is attained at $\tau = 1/2$. Thus this rate remains strictly less than 1 for all times, if and only if $\alpha < 8/3$. The condition $\alpha < 8/3$ ensures that the expected number of unit clauses created during a cascade,

$$1 + \rho_1(\tau) + \rho_1(\tau)^2 + \rho_1(\tau)^3 + \cdots = \frac{1}{1 - \frac{3\alpha}{2}\tau(1-\tau)}, \tag{9.9}$$

remains $O(1)$ for all $\tau \in [0, 1[$. On the other hand when $\alpha > 8/3$ the expected number of unit clauses grows without limit.

It remains to show that for $\alpha > 8/3$ the algorithm almost certainly fails and that for $\alpha < 8/3$ it succeeds with strictly positive probability (i.e., probability not going to zero as $n \to +\infty$). We only give the main arguments, and refer to the notes for references with detailed analysis.

Final steps of the analysis

That the algorithm fails for $\alpha > 8/3$ can be seen as follows.[2] Note $C_2(t)/(n-t)$ is the density of 2-clauses at time t. Therefore from (9.7) if $\alpha > 8/3$ we have $c_2(\tau)/(1-\tau) > 4\tau(1-\tau)$ therefore for times $\tau \approx 1/2$ (i.e. $t \approx n/2$) the density of 2-clauses is above 1. Using the uniform randomness property we have at these times a random 2-SAT formula with density larger than 1 (with possibly additional 3-clauses). It is known that the satisfiability threshold of the 2-SAT ensemble is $\alpha_s(2) = 1$. So such a formula is unsatisfiable with high probability and UCP cannot succeed.

Now let us prove that if $\alpha < 8/3$ then the algorithm succeeds. In this case, as seen from (9.9), at any point in time $C_1(t) = O(1)$ hence the probability that a variable is connected to two unit clauses is negligible and the probability that a 0-clause is created is also negligible. In other words the probability that UCP generates contradictions is negligible. Some care has to be taken to make this argument completely rigorous. In particular, as we discussed we can only guarantee the accuracy of the prediction of differential equations up to a time very close but not equal to $\tau = 1$. So we need in addition an argument which guarantees that the residual formula is satisfiable with high probability. If we look at the solution of the differential equation, we see that if we run the algorithm long enough then there is a time strictly before $\tau = 1$ where the sum of densities

[2] Another way to see that UCP fails for $\alpha > 8/3$ is to use the "birthday problem". One can show that contradictions will be generated with high probability as soon as $C_1(t) = \Theta(\sqrt{n})$. See exercises. write exercise

of 2-clauses and 3-clauses is strictly less than 1.[3] We can now argue as follows. Drop a random variable from each 3-clause. Then the resulting formula is (up to the $O(1)$ unit clauses) a random 2-SAT formula of density strictly less than 1. It is known from the analysis of the 2-SAT ensemble that such a formula is satisfiable with high probability.

9.3 Belief Propagation Guided Decimation

In the preceding section we introduced and analysed the simple UCP algorithm. This analysis established a non-trivial lower bound for the satisfiability threshold. On the downside, the UCP algorithm is not very powerful and so the bound is quite far from $\alpha_s(K)$ (for example $\alpha_s(3) \approx 4.259$ and UCP works up to $8/3 \approx 2.666$).

We now introduce and illustrate a more powerful algorithm, called Belief Propagation Guided Decimation (BPGD). The basic idea is similar to that of the UCP algorithm. At each step we pick a variable and fix its value. This variable belongs to a certain number of clauses. We remove the clauses it satisfies and shorten the ones it does not satisfy, in other words we decimate the formula and get a reduced formula. The difference with UCP lies in how we choose the variable we decimate and how we set its value. In the UCP algorithm, the choice was either forced upon us by the presence of unit clauses or was random when no unit clauses are present. In the BPGD algorithm we use belief propagation to guide the selection and set the value of the variable.

We first introduce a version of the algorithm which is guaranteed to succeed if the factor graph is a tree. We will then apply the algorithm to formulas from the random ensemble $\mathcal{F}(n, m, K)$. A rigorous analysis of BPGD for random formulas is currently out of reach and we therefore have to be assess the performance through experiments. In Chapter 17 we introduce a even more powerful algorithm (the Survey Propagation Guided Decimation algorithm).

Counting and finding solutions by decimation

We briefly recall the formulation of K-SAT in Sections 3.6 and 5.6. Given a formula we can introduce the associated factor graph with the variable nodes connected to factor nodes by full or dashed edges according to whether a variable x_i, $i = 1, \cdots, n$ appears negated or not in a clause $a = 1, \cdots, m$. A sign $J_{ia} = +1$ (resp. -1) is associated to full (resp. dashed) edges for which x_i appears un-negated (resp. negated) in clause a. Recall also that in the spin language $s_i = (-1)^{x_i}$. With these definitions $s_i = J_{ia}$ does not satisfy a, and $s_i = -J_{ia}$

[3] From (9.7) this sum is $c_3(\tau)/(1-\tau) + c_2(\tau)/(1-\tau) = \alpha(1-\tau)(1+\tau/2)$.

satisfies a. From the indicator functions of the clauses,

$$f_a(\underline{s}_{\partial a}) = 1 - \prod_{i \in \partial a} \frac{1}{2}(1 + s_i J_{ia}) \tag{9.10}$$

we form

$$\mathcal{N}_0 = \sum_{\underline{x}} \prod_{a=1}^{m} f_a(\underline{s}_{\partial a}). \tag{9.11}$$

which counts the number of solutions. The "marginal"

$$\mathcal{N}_i(s_i) = \sum_{\sim s_i} \prod_{a=1}^{m} f_a(\underline{s}_{\partial a}) \tag{9.12}$$

counts the number of solutions with s_i fixed. Suppose one is able to compute a marginal for a node i and suppose that $\mathcal{N}_i(s_i) > 0$ for $s_i = +1$, or $s_i = -1$, or both; then we know that the formula is satisfiable. Moreover we also know the total number of solutions $\mathcal{N}_i(+1) + \mathcal{N}(-1) = \mathcal{N}_0$.

Suppose for a moment that all the marginals are given to us by an oracle or that we have some way to compute them. Then the last remarks obviously imply that we know the number of solutions. The following decimation process uses the marginals to *find* solutions.

Algorithm 2: Decimation process

1. Pick an arbitrary variable i and consider the marginal $\mathcal{N}_i(s_i)$.
2. If $\mathcal{N}_i(+1) > 0$ (there exists an assignment with $s_i = +1$ or $x_i = 0$) then:
   - Set $s_i = +1$ in all clauses $a \in \partial i$.
   - Eliminate all clauses where variable i appears negated.
   - Remove s_i from the clauses where it does not appear negated.
3. If $\mathcal{N}_i(+1) = 0$ (there does not exist an assignment with $s_i = +1$), then:
   - Set $s_i = -1$ in all clauses $a \in \partial i$.
   - Eliminate all clauses where variable i does not appear negated.
   - Remove s_i from the clauses where it appears negated.
4. Repeat the process until no variables are left.

Of course many improved variants of this process can be considered. However they are not of practical use since generally we do not know the exact marginals, and this decimation is more of conceptual value. There is one case where this decimation process can be implemented: we saw in Chapter 5 how to find the exact marginals when the factor graph is a *tree*. The following simple example serves as a reminder and as an illustration of the decimation process. In the next paragraph we shall formulate an extension of these ideas to general factor graphs.

EXAMPLE 23 Consider the formula $F = x_1 \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3)$. The corresponding

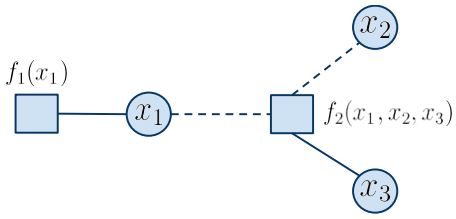


Figure 9.1 Left: factor graph of the equation $F = x_1 \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3)$. Right: reduced formula $F' = \bar{x}_2 \vee x_3$ obtained when we set $x_1 = 1$.

factor graph is shown in Figure 9.1. We express the indicator functions of each factor node as

$$f_a(s_1) = 1 - \frac{1}{2}(1 + s_1), \qquad f_b(s_1, s_2, s_3) = 1 - \frac{1}{8}(1 - s_1)(1 - s_2)(1 + s_3)$$

We want to compute (9.11), (9.12),

$$\mathcal{N}_0 = \sum_{s_1,s_2,s_3} f_a(s_1) f_b(s_1, s_2, s_3), \qquad \mathcal{N}_i(s_i) = \sum_{\sim s_i} f_a(s_1) f_b(s_1, s_2, s_3).$$

The factor graph is a tree and therefore we have access to the exact marginals using BP. We first pick variable node 1 and compute its "marginal" $\mathcal{N}_1(s_1)$. For the convenience of the reader let us recall in detail the use of message passing rules. We initialize leaf node messages, $\mu_{1\to a}(s_1) = f_a(s_1)$ and $\mu_{2\to b}(s_2) = \mu_{3\to b}(s_3) = 1$. To compute all messages that flow in node x_1 one iteration suffices,

$$\mu_{a\to 1}(s_1) = f_a(s_1) = \begin{cases} 0 \text{ if } s_1 = +1, \\ 1 \text{ if } s_1 = -1 \end{cases}$$

and

$$\mu_{b\to 1}(s_1) = \sum_{\sim s_1} f_b(s_1, s_2, s_3)\mu_{2\to b}(s_2)\mu_{3\to b}(s_3) = \begin{cases} 4 \text{ if } s_1 = +1, \\ 3 \text{ if } s_1 = -1 \end{cases}$$

Finally, multiplying the two messages flowing into node x_1 yields the marginal,

$$\mathcal{N}_1(s_1) = \mu_{b\to 1}(s_1)\mu_{a\to 1}(s_1) = \begin{cases} 0 \text{ if } s_1 = +1, \\ 3 \text{ if } s_1 = -1. \end{cases}$$

This already tells us that F has 3 solutions. To find a solution, according to the decimation process, since $\mathcal{N}_1(0) = 0$, we set $s_1 = -1$ (or $x_1 = 1$). The reduced formula then becomes $F' = \overline{x}_2 \vee x_3$. We now pick node 2 and compute

its marginal using BP on the reduced formula,

$$\mathcal{N}_2'(s_2) = \sum_{\sim s_2} (1 - \frac{1}{4}(1 - s_2)(1 + s_3) \times 1$$
$$= \begin{cases} 2 \text{ if } s_2 = +1, \\ 1 \text{ if } s_2 = -1. \end{cases}$$

Since $\mathcal{N}_2'(+1) > 0$ we set $s_2 = +1$. This choice satisfies F' so we remove the clause b', the reduced formula is empty, and for s_3 we thus have two choices $s_3 = \pm 1$. We have found two solutions $(s_1, s_2, s_3) = (-1, +1, +1)$ and $(-1, +1, -1)$. Equivalently $(x_1, x_2, x_3) = (1, 0, 0)$ and $(1, 0, 1)$). Of course, after setting $s_1 = +1$, since $\mathcal{N}_2'(-1) > 0$ for F', we can also look for solutions with $s_2 = -1$. This then yields a reduced formula $F'' = x_3$ where we are forced to take $x_3 = 1$ or $s_3 = -1$ (note $\mathcal{N}_3''(+1) = 0$, $\mathcal{N}_3''(-1) = 1$). Thus the third solution is $(s_1, s_2, s_3) = (-1, -1, -1)$. Equivalently $(x_1, x_2, x_3) = (1, 1, 1)$.

We computed $\mathcal{N}_1(+1) = 0$ and $\mathcal{N}_1(-1) = 3$ which means that there must exist three solutions in total and also that they all have $s_1 = -1$. The reader can check that BP applied to F yields $\mathcal{N}_2(+1) = 2$, $\mathcal{N}_2(-1) = 1$ and $\mathcal{N}_3(+1) = 1$, $\mathcal{N}_3(-1) = 2$. Thus there must exist two solutions with $s_2 = 0$ and one solution with $s_2 = -1$. Also, there must exist one solution with $s_3 = +1$ and two solutions with $s_3 = -1$. This is all consistent with the solutions that we found.

We point out that if one is interested in the *fraction* of satisfying solutions we can just normalize the messages,

$$\nu_i(s_i) = \frac{\mathcal{N}_i(s_i)}{\mathcal{N}_i(+1) + \mathcal{N}_i(-1)}. \tag{9.13}$$

Here we find $\nu_1(+1) = 0$, $\nu_1(-1) = 1$, $\nu_2(+1) = 2/3$, $\nu_2(-1) = 1/3$, $\nu_3(+1) = 2/3$, $\nu_3(-1) = 1/3$.

Of course for such a small formula we can directly obtain solutions and marginals from the truth table 9.1 (without ever using BP). The reader can verify that everything is perfectly consistent.

□

BPGD for general formulas

We now adapt the decimation process in order to turn it into an algorithm applicable to formulas with general factor graphs. However, note that the graphs we have in mind should be sparse.

Over a tree, BP yields exact marginals and we can pick anyone of them in each iteration of the decimation process. But for general graphs marginals computed by BP - call them $\nu_i^{\mathrm{BP}}(s_i)$ - are not exact so it will matter which ones we pick. In order to potentially minimise the effect of the uncertainty of a marginal, in each iteration we pick a node i such that the *bias* $B_i \equiv |\nu_i^{\mathrm{BP}}(+1) - \nu_i^{\mathrm{BP}}(-1)|$

x_1	x_2	x_3	$f_1(x_1)$	$f_1(x_1,x_2,x_3)$	F
0	0	0	0	1	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	0	1	0
1	0	0	1	1	1
1	0	1	1	1	1
1	1	0	1	0	0
1	1	1	1	1	1

Table 9.1 Satisfiability of $F = x_1 \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3)$ for all possible combination of x_1, x_2 and x_3.

is *maximized*.[4] This way, we hope that this node has such a clear bias that its marginals are quite exact despite the graph not being a tree.

The BPGD algorithm is summarized in table 3

Algorithm 3: BPGD algorithm

1. Run BP and calculate all marginals.
2. Pick a node i with maximum bias $B_i = |\nu_i^{\mathrm{BP}}(0) - \nu_i^{\mathrm{BP}}(1)|$ greater than some predefined small number $\delta > 0$. If there are many such nodes pick one at random among them. If all biases are smaller than $\delta > 0$ pick any variable at random.
3. Set s_i to the most likely value, i.e., $s_i = +1$ if $\nu_i^{\mathrm{BP}}(+1) > \nu_i^{\mathrm{BP}}(-1)$ and to $s_i = -1$ otherwise.
4. Eliminate all clauses satisfied by the value of s_i set in the previous step. Remove s_i from the other clauses.
5. Recurse until all variables are eliminated.
6 Output $(s_1, \ldots, s_n)$ and check if it satisfies the formula.

A few remarks are in order. When we run BP on general graphs we have to decide in an ad-hoc way an initialisation of the messages and specify a schedule. A convenient schedule is the flooding schedule. At each iteration step all variable nodes send their messages towards clauses and all clauses send back their message to variable nodes. For the initialization, since at the beginning we have no information whatsoever about the true marginals, it is natural to initialise the (unnormalised) messages uniformly, in other words $\nu_{i\to a}(s_i) = 1/2$ for all $i = 1, \cdots, n$.

We now discuss a few experiments. Before illustrating the performance of the BPGD algorithm itself, we say a few words on the convergence of BP itself. In

[4] The bias is the absolute value of the BP-estimate of the magnetization.

principle, convergence of the BP messages is a prerequisite for computing the maximal bias and decimating the formula. To test this issue we can run BP over many instances and compute the empirical probability that it converges. The resulting probability of convergence as a function of α is shown on figure 9.2. For $K = 3$ we get a convergence threshold $\alpha_{\rm conv}(3) \approx 3.86$ and for $K = 4$ we get $\alpha_{\rm conv}(4) \approx 10.3$.

The empirical probability of success of BPGD computed from runs over many instances is also illustrated on figure 9.2. The probability of success remains strictly positive until $\alpha_{\rm BPGD} \approx 3.86$ for $K = 3$ (this is approximately identical to $\alpha_{\rm conv}(3)$). For $K = 4$ it remains strictly positive until $\alpha_{\rm BPGD} \approx 9.3$ (which is here smaller than $\alpha_{\rm conv}(4)$). These values can be compared to the satisfiability thresholds, $\alpha_s(3) \approx 4.26$ and $\alpha_s(4) \approx 9.93$, predicted by the cavity method.

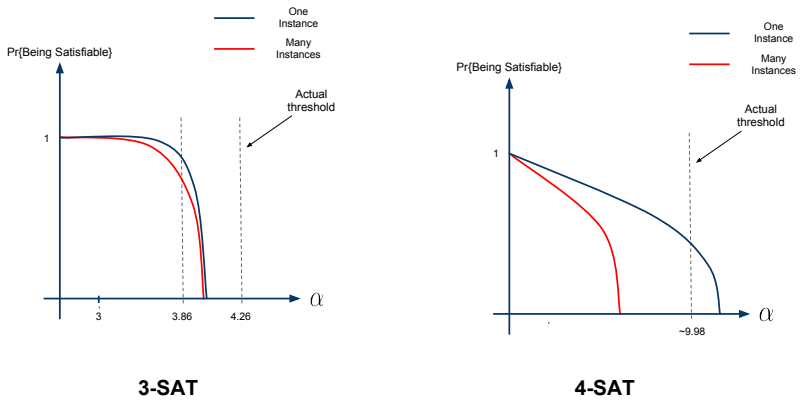


Figure 9.2 Empirical probabilities of convergence of the BP algorithm and of success of BPGD for 3-SAT (left) and 4-SAT (right).

In Chapters 16 and 17 we develop the cavity method and the associated Survey Propagation Guided Decimation algorithm which empirically finds solutions for higher values of α. We will also learn that the limitations of the BPGD algorithm are fundamentally related to the geometry of the solution space in the Hamming hypercube $\{0, 1\}^n$.

Because of Friedgut's theorem (Chapter 1) a strictly positive probability of success of an algorithm implies that the formula is satisfiable with probability one in the limit $n \to +\infty$. Therefore if one would rigorously analyze the BPGD decimation process, just as we did for UCP, the above thresholds would be provable lower bounds for $\alpha_s(K)$. Unfortunately such an analysis is difficult and is currently out of reach for small values of K mainly because the uniform randomness property does not hold and one faces the problem of tracking the evolution of the ensemble of random formulas. But note that for large values of K approximations of BPGD have been analysed (see the notes).

9.4 A convenient parametrization of the BP equations

Since the alphabet is binary we can parametrize the messages in ways similar to coding and Ising type spin models. In K-SAT we have the extra feature that the edges of the factor graph carry a sign $J_{ia} = \pm 1$, and it turns out that a slightly different parametrisation is convenient. This parametrisation will mostly be used in later Chapters but the reader can already use it to implement BPGD in the exercises.

Our parametrization uses the following loglikelihood variables

$$h_{i\to a} = \frac{1}{2}\ln\left\{\frac{\mu_{i\to a}(-J_{ia})}{\mu_{i\to a}(J_{ia})}\right\}, \qquad \widehat{h}_{a\to i} = \frac{1}{2}\ln\left\{\frac{\widehat{\mu}_{a\to i}(-J_{ia})}{\widehat{\mu}_{a\to i}(J_{ia})}\right\}. \tag{9.14}$$

The sum-product equation for messages $\mu_{i\to a}$ flowing from variable to constraint nodes is given by

$$\begin{aligned}\mu_{i\to a}(\pm J_{ia}) &= \prod_{b\in\partial i\setminus a} \hat{\mu}_{b\to i}(\pm J_{ia}) \\ &= \prod_{b\in\partial i\setminus a: J_{ib}=J_{ia}} \hat{\mu}_{b\to i}(\pm J_{ib}) \prod_{b\in\partial i\setminus a: J_{ib}\neq J_{ia}} \hat{\mu}_{b\to i}(\mp J_{ib})\end{aligned}$$

Taking the logarithm of the ratio of these two equations we find

$$\begin{aligned}h_{i\to a} &= \sum_{b\in\partial i\setminus a: J_{ib}=J_{ia}} \hat{h}_{b\to i} - \sum_{b\in\partial i\setminus a: J_{ib}\neq J_{ia}} \hat{h}_{b\to i} \\ &= \sum_{b\in\partial i\setminus a} J_{ia}J_{ib}\hat{h}_{b\to i}\end{aligned} \tag{9.15}$$

This is the first BP equation for K-SAT. Consider now the other sum-product equation for messages $\widehat{\mu}_{a\to i}$ flowing from constraint to variable nodes. This involves the factor

$$f_a(\underline{s}_{\partial a}) = 1 - \prod_{j\in\partial a}\frac{1}{2}(1+s_jJ_{ja}) \tag{9.16}$$

evaluated at $s_i = \pm J_{ia}$. For $s_i = -J_{ia}$ this evaluates to 1 (the clause is satisfied)

$$\begin{aligned}\widehat{\mu}_{a\to i}(J_{ia}) &= \sum_{s_j, j\in\partial a\setminus i}\ \prod_{j\in\partial a\setminus i}\mu_{j\to a}(s_j) \\ &= \prod_{j\in\partial a\setminus i}(\mu_{j\to a}(J_{ja}) + \mu_{j\to a}(-J_{ja})).\end{aligned} \tag{9.17}$$

For $s_i = J_{ia}$ the factor (9.16) is equal to $1 - \prod_{j\in\partial a\setminus i}\frac{1}{2}(1+s_jJ_{ja})$ which implies

$$\widehat{\mu}_{a\to i}(-J_{ia}) = \sum_{s_j, j\in\partial a\setminus i}\left[1 - \prod_{j\in\partial a\setminus i}\frac{1}{2}(1+s_jJ_{ja})\right]\prod_{j\in\partial a\setminus i}\mu_{j\to a}(s_j). \tag{9.18}$$

Combining (9.18) and (9.17) we get

$$\begin{aligned}\widehat{\mu}_{a\to i}(-J_{ia}) &= \widehat{\mu}_{a\to i}(J_{ia}) - \sum_{s_j, j\in\partial a\setminus i}\Big[\prod_{j\in\partial a\setminus i}\frac{1}{2}(1+s_jJ_{ja})\Big]\prod_{j\in\partial a\setminus i}\mu_{j\to a}(s_j)\\ &= \widehat{\mu}_{a\to i}(J_{ia}) - \prod_{j\in\partial a\setminus i}\mu_{j\to a}(J_{ja})\\ &= \widehat{\mu}_{a\to i}(J_{ia})\Big[1-\prod_{j\in\partial a\setminus i}\frac{\mu_{j\to a}(J_{ja})}{\mu_{j\to a}(J_{ja})+\mu_{j\to a}(-J_{ja})}\Big]\end{aligned}$$

Finally, dividing both sides by $\hat{\mu}_{a\to i}(J_{ia})$ and taking the logarithm we obtain

$$\hat{h}_{a\to i} = \frac{1}{2}\ln\Big\{1-\prod_{j\in\partial a\setminus i}\frac{1}{1+e^{2h_{j\to a}}}\Big\}. \tag{9.19}$$

Summarizing, the message passing equations (9.15) and (9.19) for K-SAT can be cast as

$$\begin{cases} h_{i\to a} = \sum_{b\in\partial i\setminus a} J_{ia}J_{ib}\hat{h}_{b\to i}\\ \hat{h}_{a\to i} = \frac{1}{2}\ln\{1-\prod_{j\in\partial a\setminus i}\frac{1}{2}(1-\tanh h_{j\to a})\}\end{cases} \tag{9.20}$$

Let us now work out the expresion of the bias used in the BPGD algorithm. An easy calculation shows that in terms of the "mean field"

$$h_i = \frac{1}{2}\ln\Big\{\frac{\nu_i^{\rm BP}(+1)}{\nu_i^{\rm BP}(-1)}\Big\}$$

the bias is $B_i = |\nu_i^{\rm BP}(+1) - \nu_i^{\rm BP}(-1)| = |\tanh h_i|$. To compute h_i from BP messages we form the ratio,

$$\begin{aligned}\frac{\nu_i^{\rm BP}(+1)}{\nu_i^{\rm BP}(-1)} &= \frac{\prod_{a\in\partial i}\mu_{a\to i}(+1)}{\prod_{a\in\partial i}\mu_{a\to i}(-1)}\\ &= \prod_{a\in\partial i: J_{ia}=+1}\frac{\mu_{a\to i}(J_{ia})}{\mu_{a\to i}(-J_{ia})}\prod_{a\in\partial i: J_{ia}=-1}\frac{\mu_{a\to i}(-J_{ia})}{\mu_{a\to i}(J_{ia})}\\ &= \prod_{a\in\partial i: J_{ia}=+1}e^{-2\hat{h}_{a\to i}}\prod_{a\in\partial i: J_{ia}=-1}e^{2\hat{h}_{a\to i}}.\end{aligned} \tag{9.21}$$

The final expression for the bias is

$$B_i = |\nu_i^{\rm BP}(+1) - \nu_i^{\rm BP}(-1)| = |\tanh(\sum_{a\in\partial i}J_{ia}\hat{h}_{a\to i})| \tag{9.22}$$

9.5 Notes

Wormald method (other names in other disciplines), Wormald proved general thm for graph processes. Application to K sat: review of D.A for details. Uniform randomness and principle of deferred decisions (paper of Kiroussis et al). Nature of computation book. BPGD see also MM. Give a good reference for analysis of

2-SAT (say also in Chap one about 2 SAT). Mention recent progress on analysis of approximations of BPGD for large K

Problems

9.1 Application of Wormald's theorem. Consider the toy model of section 9.1 and check in detail that the hypothesis of Wormald's theorem hold.

9.2 Peferential attachment. The purpose of this problem is to use the Wormald method to study a model for "preferential attachment." Consider n nodes. Initially all nodes have degree 0. Assume that we allow a maximum degree of $d_{\max}$. We proceed as follows. At every step pick two nodes from the set of all nodes which have degree at most $d_{\max}-1$ and "attach" them by an edge. Rather than picking them with uniform probability pick them proportional to their current degree. More precisely, assume that at time t you have $D_i(t)$ nodes of degree i. Then pick a node of degree i with probability

$$P_i(t) = \begin{cases} \frac{D_i(t)}{\sum_{j=0}^{d_{\max}-1} D_j(t)}, & 0 \leq i < d_{\max}, \\ 0, & i = d_{\max}. \end{cases}$$

Initially, we have $D_0(t=0) = n$ and $D_i(t=0) = 0$ for $i = 1, \cdots, d_{\max}$. Note that at time $t = nd_{\max}/2$ all nodes will have maximum degree. Pick $d_{\max} = 4$.

(i) Write down the set of differential equations for this problem. Are the conditions of Wormald's theorem fulfilled?

(ii) Plot the evolution of the degree distribution as a function of the normalized time for $\tau = t/n \in [0, d_{\max}/2]$. Hint: In general one cannot expect to solve the system of differential equations analytically. But it is typically easy to solve them numerically (with Mathematica for example).

9.3 Unit Clause Propagation for $K \geq 3$. We want to generalise the analysis of UCP to any $K \geq 3$.

(i) State the general form of the algorithm.

(ii) Generalize the derivation of the differential equations (9.6).

(iii) Solve analytically the differential equations and obtain the generalised form of (9.7).

(iv) Compute the rate of unit clause production and obtain the generalised form of (9.8).

(v) Deduce from the previous point the large K asymptotics of the UCP threshold $\alpha_{\mathrm{UCP}}(K)$.

9.4 Implementation of Belief Propagation. Implement BP according to the flooding (or parallel) schedule. Initialize the messages uniformly randomly in $[0, 1]$. One iteration means that you send messages from nodes to clauses and back from clauses to variables. Define the following "convergence criterion": declare that the messages have "converged" if there is an iteration number (time) $t_{\mathrm{conv}}(\epsilon)$ such that no messages changes by more than ϵ at $t_{\mathrm{conv}}(\epsilon)$ (take the smallest such time).

(i) Perform the following experiment. Take 100 K-SAT instances of length say $n = 5000$ and 10000 variables and for each instance implement BP as explained above with $\epsilon = 10^{-2}$. If the iterations do not converge stop them at a large time say $t_{\max} \approx 1000$. When they converge, they should do so in a shorter time $t_{\rm conv}(\delta) < t_{\max}$ that does not change much with n.

(ii) Plot as a function of α the empirical probability that the iterations converge. You should see that this probability is large for $\alpha < \alpha_{\rm conv}$ and drops abruptly around some threshold $\alpha_{\rm conv}$. For $K = 3$, $\alpha_{\rm conv} \approx 3.85$ and $K = 4$, $\alpha_{\rm conv} \approx 10.3$.

9.5 Implementation of Belief Propagation Guided Decimation. Implement the BPGD algorithm for finding SAT assignments. It uses BP (implemented as in the previous exercise) as a guide to take decisions on how to fix values for the variables. Once a variable has been fixed the K-SAT formula is suitably reduced - this step is called "decimation" - and BP is run again.

- Initialize with a K-SAT formula $F \in \mathcal{F}(n, m, K)$ of length n.
- For $t = 1, \ldots, \min(t_{\rm conv}(\epsilon), t_{\max})$ do:
  - Run BP on an instance, as in the previous exercise (with the same convergence criterion).
  - If BP does not converge, return "assignment not found" and exit.
  - If BP converges, for each variable i compute its bias B_i (Equ. (9.22))
  - Pick a variable i that has the largest bias B_i.
  - If $\nu_i^{\rm BP}(+1) - \nu_i^{\rm BP}(-1) \geq 0$ fix $s_i = +1$. Otherwise fix $s_i = -1$. (Note that $\nu_i^{\rm BP}(+1) - \nu_i^{\rm BP}(-1)$ is given by Equ. (9.22) without the absolute value.)
  - Replace F by the K-SAT formula obtained by decimating variable i.
- End-For
- Return all fixed variables.

Give for several values of α, the empirical success probability of this algorithm when tested over 100 instances. Compare this empirical success probability with the empirical convergence probability of the previous exercise. You should observe that $K = 3$ and $K = 4$ do not behave on the same way. Try to find an approximate threshold $\alpha_{\rm BPGD}$ beyond which the algorithm does not find SAT assignments.

10 Maxwell Construction

This chapter is an epilogue for the message passing approaches discussed so far. It is also a preview of the sort of questions we address in part III. The analysis of message passing has allowed us to determine algorithmic thresholds for the systems of interest. In coding for example, this threshold is the noise level computed by density evolution which tracks the belief propagation algorithm. For random K-SAT we determined critical clause densities associated to unit clause propagation or to belief propagation guided decimation. For compressive sensing we found that the threshold of approximate message passing algorithm for LASSO is the Donoho-Tanner curve determined by state evolution. These problems typically also have an "optimal" threshold which is independent of the algorithms. In coding this is the maximum posterior threshold beyond which it is impossible to reconstruct the input message even by exhaustive search. For random K-SAT this corresponds to the satisfiability threshold beyond which, with high probability, there are no solutions. These thresholds are optimal in the sense that they have nothing to do with algorithms but are associated to a phase transition *inherent* to the problem. In general, we want to be able to compute such thresholds in order to assess how far our algorithmic thresholds are from the optimal ones.

But is it possible to determine the phase transition (or optimal) thresholds? This is not at all obvious for problems that are computationally hard. The *Maxwell construction* is a method that allows to guess the location of phase transition thresholds by "looking" at the message passing solution through the "correct lenses". Once the Maxwell construction has given us a guess, this can then often be converted into a rigorous statement. The point here is that typically the proof uses the guess as an essential input. For us, the Maxwell construction is a crucial first step in the proof.

Whenever this program works, then this means that the message-passing algorithm is not just a convenient low-complexity algorithm among other ones, but is fact a fundamental algorithm in caracterising the problem. As we will see more extensively in Part III, message passing solutions not only allow to guess the phase transition (optimal) thresholds, but are also intimately related to the full solution behind the problem at hand.

The original Maxwell construction goes back to the 19th century adventure of trying to understand the liquid-vapour phase transition for simple substances

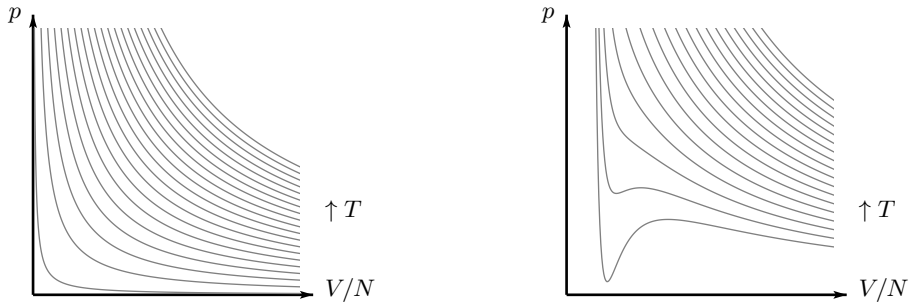


Figure 10.1 Left: isotherms of the ideal gas equation of state. Right: isotherms of the van der Waals equation of state. Note that below a critical temperature, the isotherms are no longer monotone.

(say H_2O). Quite surprisingly, even though at first sight this problem seems to have little to do with coding, compressive sensing or satisfiability, there is a deep analogy between the original Maxwell construction and the one in our cases. It is therefore worth to briefly review the original Maxwell construction. We then point out that we have already seen an example of this construction for the Curie-Weiss model without so far spelling it out explicitly. After these preparations we turn to coding theory and treat in some detail the case of the binary erasure channel where the Maxwell construction is the simplest. A more comprehensive treatment of the Maxwell construction requires new concepts and tools that are developed in Part III.

10.1 The Maxwell construction for the liquid-vapor transition

Assume that we have a gas consisting of N molecules in a container of volume V, at thermal equilibrium at a temperature of T and under a pressure[1] of p. How are these quantities related? The *ideal* gas law states that

$$pV = Nk_BT, \tag{10.1}$$

where k_B is Boltzmann's constant. The left picture in Figure 10.1 shows this relationship at different temperatures T. As one can see from this picture and as we intuitively know, as we decrease the volume, the pressure increases. The ideal gas law is based on several simplifying assumptions. In reality typical molecules interact via forces with a very short range and strong repulsive part, and a weak long range attractive part.[2] Because of the short range strong repulsion it is a good model to assume that the molecules have an "effective volume". The ideal gas law simply *neglects* this effective volume as well as the attractive part of the

[1] At thermal equilibrium pressure is constant throughout the bulk of the system and equals force by unit surface exerted by the system on the walls of the container.

[2] Both repulsive and attractive parts have quantum mechanical origin. At short distance electronic clouds repel each other due to the Pauli principle and at long distances the quantum fluctuations leave out a dipolar Coulombic interaction between molecules.

force (all forces are neglected hence the name ideal). The relation expressed in (10.1) is an *equation of state*, since it relates quantities (p, V, T, N) that define the "thermodynamic equilibrium state" of the system.

In 1873 van der Waals derived a more accurate equation of state taking into account the non-zero effective size of the molecules produced by the strong repulsive forces, as well as the weak long range attracting forces. The van der Waals equation is

$$(p + a\frac{N^2}{V^2})(V - bN) = Nk_BT. \tag{10.2}$$

where a and b are (dimensionfull) constants that characterize the forces between molecules. This equation is very similar in structure to the ideal gas law, but both the volume as well as the pressure terms are modified. The constant b takes into account the strong repulsion equivalent to an effective finite size for each molecule. Due to this finite size the *effective volume of the box* which is available to the N molecules shrinks from V to $V - bN$. The constant a takes into account attractive forces between molecules. It is assumed that these attractive forces act only between molecules of the gas but not between the wall of the container and gas molecules. Therefore, close to the boundary of the container, a molecule has more neighbors away from the boundary than towards the boundary and this creates an effective force "inwards," reducing the pressure of the gas. The reduction is proportional to N^2 because each molecule close to the wall feels the effect of approximately N other molecules and there are of the order of N molecules close to the wall. To obtain an intensive quantity[3] we have to divide by V^2, which gives a reduction in pressure by an amount $-aN^2/V^2$. Taking into account the reduction in effective volume and pressure the ideal gas law (10.1) is modified to

$$p = NkT/(V - bN) - a\frac{N^2}{V^2}.$$

This is equivalent to (10.2).[4]

The right-hand side picture in Figure 10.1 shows the van der Waals isotherms for some choice of constants a and b and for various choices of T. Comparisons with measurements show that the predictions of the van der Waals equation are for the most part more accurate compared to the predictions of the ideal gas equation. But a closer look at Figure 10.1 shows a somewhat curious and non-physical behavior. Below a "critical" temperature, the isotherms are no longer relating the pressure p and the volume per molecule V/N in a monotone fashion. Below this critical temperature, there is a range where a decrease in volume leads

[3] Pressure is intensive, i.e., independent of system size.

[4] In terms of the density $\rho = N/V$ the van der Waals equation of state reads $p = \rho k_{\mathrm{B}}T/(1 - b\rho) - a\rho^2$, and reduces to the ideal gas law for $\rho \to 0$. The constants a and b must be determined by experiment. On the other hand statistical mechanics computes equations of state starting from microscopic Hamiltonians involving the intermolecular forces, which allows to relate a and b to the expressions of the forces. Thus one can learn information about the forces from experimental measurements of the equation of state.

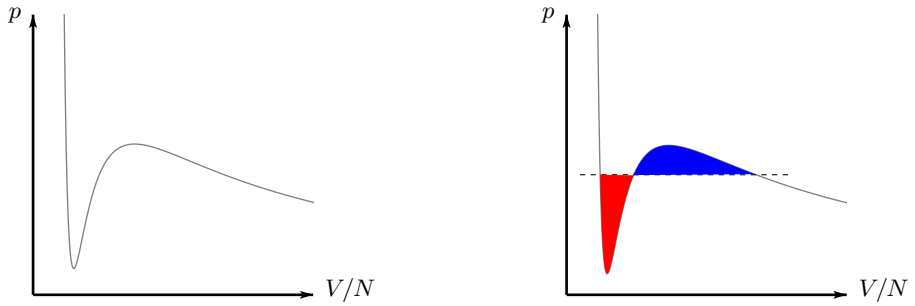


Figure 10.2 The Maxwell construction. Left: one isotherm of the van der Waals equation of state. Right: the same isotherm, where a part of the curve is replaced by a horizontal line which is placed so that the two enclosed areas are in balance.

to a *decrease* in pressure. Clearly, the thermodynamic equilibrium state is not described accurately in this range.

It was Maxwell who in 1875 suggested a modification of the van der Waals isotherms to account for this unphysical behavior. Consider Figure 10.2. The picture on the left shows one van der Waals isotherm which has a non-physical non-monotonous behavior. The Maxwell construction modifies this curve by replacing part of the curve with a horizontal line, so as to obtain a monotonous isotherm. This line is placed in such a way that the two areas above and below the line are in balance. Note that these two areas represent a "work" since they are equal to a pressure times a volume, i.e., a force times a length. The basic thermodynamic argument to justify the equality of the two areas is, roughly speaking, as follows. Suppose one slowly compresses the system starting at large volumes and then slowly relaxes the volume, so that at any instant the system remains in thermal equilibrium at the same temperature. The work *released* to the system by compressing the gas along the curved van der Waals isotherm and the work *gained* by relaxing the volume along the straight line back to its original value should be equal because the system has returned to its initial state. Indeed no net work should be gained or done in the process otherwise we would have a a way to extract energy from a single heat bath at constant temperature, in contradiction to the second principle of thermodynamics.[5]

The horizontal line segment corresponds to a situation where the system coexists in two phases, namely as liquid and as vapor. Along this line the percentage of each component changes from all vapor at the right-most point to all liquid at the left-most point. Note that as soon as all the gas is in liquid form, any further decrease in volume leads to a very large increase in pressure.

It is important to realize that for this physical system neither the ideal gas equation, nor the van der Waals equation, and not even the isotherms modified by the Maxwell construction describe the system *exactly*. These are all increasingly better and more accurate descriptions, taking into account more and more physical effects (and one can of course go further by sophisticated statistical

[5] Equivalently we would have a perpetual mobile.

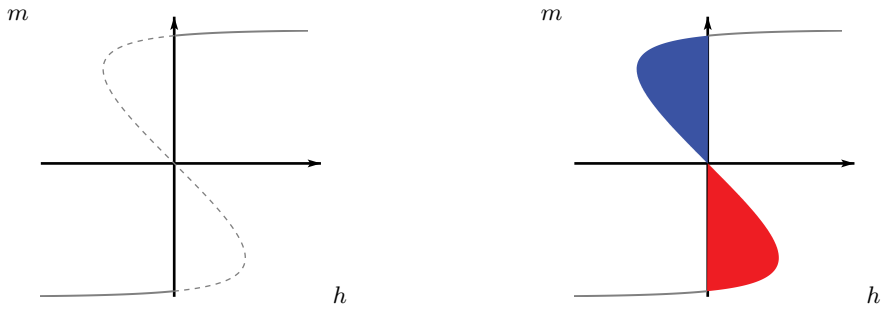


Figure 10.3 Phase transition in Curie-Weiss model when $\beta J > 1$ as a function of h (here $\beta J = 2$). The phase transition is at $h = 0$.

mechanical calculations of the equation of state). The point is that they agree reasonably well with physical experiments. For our applications we are in a conceptually much easier situation. Our aim is not to find a correct theoretical description for a real physical system. Rather, we start with a *model* and therefore, in such a situation we can even hope that the Maxwell construction gives us a *mathematically exact* result. Still, it is quite remarkable that this is the case for models about problems of engineering relevance.

10.2 Maxwell construction for the Curie-Weiss model

For the Curie-Weiss model we have already encountered a simple form of the Maxwell construction! In Chapter 4 we solved the model and computed the exact relationship between the magnetization m and the external magnetic field h for a particular temperature. We saw in Section 4.2 that the magnetisation takes on a value which minimizes the potential function

$$\Phi(m) = -\frac{J}{2}m^2 - hm - \beta^{-1}h_2\big(\frac{1+m}{2}\big). \tag{10.3}$$

Recall that if we take the derivative of the above expression, we find that m is a solution of the fixed-point equation

$$m = \tanh(\beta J m + \beta h). \tag{10.4}$$

For $\beta J < 1$, this fixed-point equation has a single solution for each h, but for $\beta J > 1$ it has up to three solutions, depending on the magnitude of h. When many solutions are present, we have to choose the one that minimises the free energy function (10.3). The left picture in Figure 10.3 shows the resulting relationship between h and m for $\beta J = 2$. The dashed part of the curve are points (h, m) which are solutions to the fixed-point equation but where m is not the minimizer of (10.3). Equation (10.4) is the analog of the van der Waals equation of state with magnetisation and magnetic field being the analogs of volume per particle and pressure. This is vividly visible if one compares the van der Waals

isotherms Figure 10.3. The left picture on this figure shows the exact solution of the Curie-Weiss model obtained through the minimisation of (10.3). This exact solution confirms the Maxwell construction which here consists of correcting the picture by drawing a vertical line separating the two equal areas on each side of the curve. To summarise, we see that for the Curie-Weiss model the Maxwell construction is exact.

So far message passing did not enter our considerations. Indeed on one hand it is not relevant for the liquid-vapor transition and on the other hand we know how to solve the Curie-Weiss model by the exact computation of the partition function, as seen in Chapter 4.

But recall in the end of Section 7.2 we remarked that the simplifications of BP equations applied to the Curie-Weiss model lead to a message passing form of the Curie-Weiss equation, namely Equ. (7.20). There is an important lesson to be drawn here. For this model message passing leads to a correct equation of state except for the fact that it does not provide a principled way of selecting the correct solution of the fixed point equation when multiple ones exist. Indeed in the message passing world we do not know that we "should" minimize the potential function (10.3). From the message passing perspective we start with a particular value of m and then we iterate Equ. (7.20), which eventually yields the "van der Waals" isotherm depicted on the left Fig. 10.3. The Maxwell construction provides the missing step. It allows to "correct" the message passing solution in order to recover the exact solution. Moreover it also gives a way to determine the location of the phase transition threshold, namely $h = 0$ for the present model. It might be argued that this last point is somewhat trivial because it follows from the symmetry of the model. While this is true, at the same time, we have here a powerful principle that we shall apply in much less trivial situations with no symmetry present.

Let us pause to see where we are. We have seen the Maxwell construction for two examples, but so far it is perhaps not very convincing. For the liquid-vapor system the Maxwell construction might appear like a kludge – a rough fix for an obvious problem. For the Curie-Weiss model, on the other hand, it might appear like a very lucky coincidence.

It would be much more compelling if we could start with the belief propagation equations for a non-trivial system and then from these equations could prove that the actual equation of state and phase transition threshold have to be of the form predicted by the Maxwell construction. In particular, this will be compelling if the actual equation of state and phase transition threshold are difficult to compute directly. In the next section we discuss exactly such a case – namely the case of coding on the binary erasure channel. Here the Maxwell construction does indeed give the correct prediction for the MAP threshold and is the starting point for a rigorous derivation of this quantity.

10.3 Coding: the Maxwell construction for the BEC

Let us now consider coding, using Gallager's (d_v, d_c)-regular LDPC ensemble, transmission over the BEC, and BP decoding. For this case we will see how we can determine the MAP threshold exactly. First we explain how, thanks to the Maxwell construction, the MAP threshold can be guessed from the DE analysis of the BP algorithm. In the next two sections we go through the key ideas that allow to prove that the guess is indeed correct.

Contrary to a fluid or a magnetic system considered in the previous examples, it is less clear which are the correct analogous "state variables" and "isotherms" to which we should apply Maxwell's equal area construction. This is the first question we have to clarify.

We saw in Chapter 6 that the analysis of BP lead to the DE fixed point equation for the erasure probability

$$x = \epsilon(1-(1-x)^{d_c-1})^{d_v-1}\,. \tag{10.5}$$

By looking at (10.4) and (10.5) one might be tempted to make the following analogies: $\epsilon \leftrightarrow p$ or h the "control" parameters and $x \leftrightarrow V/N$ or m the "state" variables. The "van der Waals isotherms" would then be the curves $(\epsilon(x), x)$ where

$$\epsilon(x) = \frac{x}{(1-(1-x)^{d_c-1})^{d_v-1}} \tag{10.6}$$

We hasten to say that this is *not* the correct way to proceed. To find the correct analogy, let us have a closer look at the message passing "solution" of the Curie-Weiss model in Section 7.2. If one traces the derivations leading to Equs (7.19)-(7.20) we notice that the BP-magnetization at a vertex i is a function of incoming messages that flow on the edges $j \to i$ from *all* neighbors $j \in \partial i$. In density evolution the analogous quantity is

$$h_{\rm BP}(x) \equiv (1-(1-x)^{d_c-1})^{d_v}, \tag{10.7}$$

the erasure probability of a randomly chosen variable node, obtained using only all "internal" messages and ignoring the directly received observation. Let us compare this expression to teh right hand side of (10.5). Since we ignore the direct observation the factor ϵ is missing; on the other hand we have a power of d_v in this expression and not just $(d_v - 1)$ as in the density evolution equation since we take all internal inputs into account.[6] The correct curve to which one should apply the Maxwell construction is

$$(\epsilon(x), h_{\rm BP}(x)) \quad \text{for} \quad 0 < x \leq 1 \quad \text{and} \quad (\mathbb{R}, 0) \quad \text{for} \quad x = 0. \tag{10.8}$$

This parametric curve (with parameter $0 \leq x \leq 1$) is known in the literature

[6] The notation $h_{\rm BP}$ expresses the fact that this probability is intimately related to the entropy of a BP-decoded bit on the BEC. On the BEC if the decoded bit is not erased then its entropy is zero while if its erased its entropy is $\ln 2$. Thus the total entropy of a bit under BP decoding is $\epsilon\, h_{\rm BP}(x) \ln 2$.

as the BP EXIT *curve*.[7]. It constitutes the correct analog of the van der Waals isotherm. Note that here we have only *one* isotherm since there is no explicit temperature parameter (equivalently we work with $\beta^{-1} = 1$; see Chapter 3 for a discussion of this point).

We now illustrate the Maxwell construction with an example.

EXAMPLE 24 (Graphical Characterization of Thresholds) The left-hand side of Figure 10.4 shows the BP EXIT curve associated to the $(3, 6)$-regular ensemble. This is the curve given by (10.8). For all regular ensembles with $d_v \geq 3$ this curve has the characteristic "C" shape starting at the point $(1, 1)$ for $x = 1$ and then moves downwards and extends to $(+\infty, 0)$ for $x \to 0$. For $x = 0$ we add to this curve the horizontal axis $\mathbb{R}$. Note that $\epsilon(x) > 1$ cannot interpret it as a channel erasure probability. Nevertheless, in the Maxwell construction we must take into account the whole curve. One inserts a vertical line at that unique spot so that

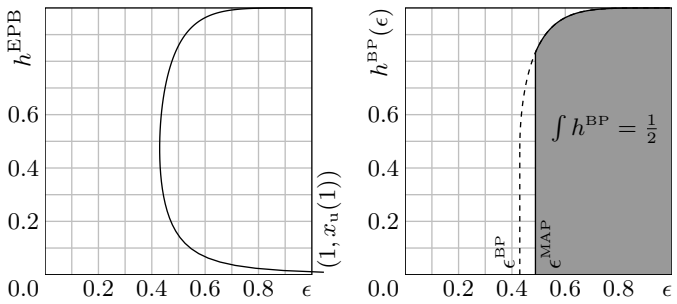


Figure 10.4 Left: the BP EXIT curve of the $(d_v = 3, d_c = 6)$-regular ensemble. The curve goes "outside the box" and tends to infinity and "comes back" on the horizontal axis. Right: the Maxwell construction places a vertcal line such that the two shaded areas balance. The position of this line yields ϵ_{MAP}.

the two shaded areas have the same area. One can prove that the location of this line is precisely equal to ϵ_{MAP}. Note that the BP threshold is also found from the BP EXIT curve as the location of a vertical line tangent to the "C" shape.

10.4 Formal statement of the Maxwell construction and related definitions

The Maxwell construction only gives us a guess of the MAP threshold, and to prove this conjecture needs more work. This section formalises the conjecture and introduces a few key definitions that are useful for the proof (in section **??**).

[7] Here EXIT stands for extrinsic information transfer. This is the information acquired when one takes into account all incoming messages except the one coming from the channel observation. This parametric curve has also been called "extended EXIT curve" in the literature to distinguish it from EXIT *functions* introduced below.

DEFINITION 10.1 (Trial entropy and area threshold) We define the *trial entropy* as

$$\begin{aligned} A(x) &= \int_0^x d\epsilon(x')\, h_{\mathrm{BP}}(x') \\ &= x + \frac{d_v}{d_c}(1-x)^{d_c-1}(d_v + x(d_v d_c - d_v - d_c)) - \frac{d_v}{d_c}. \end{aligned} \tag{10.9}$$

This integral is easily computed from (10.7) and (10.6). Note that $A(x)$ is the area under the EXIT curve, from the point $(\epsilon(0), h_{\mathrm{BP}}(0)) = (+\infty, 0)$ for until the point $(\epsilon(x), h_{\mathrm{BP}}(x))$, as indicated in Figure 10.5. By definition $A(0) = 0$. The

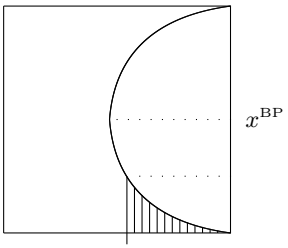


Figure 10.5 The trial entropy $A(x)$ measures the shaded area *below* the BP EXIT curve.

function $A(x)$ is decreasing for $0 \le x \le x_{\mathrm{BP}}$ where x_{BP} is the unique parameter such that $\epsilon_{\mathrm{BP}} = \epsilon(x_{\mathrm{BP}})$. For $x_{\mathrm{BP}} \le x \le 1$ it is increasing, and $A(1) = 1 - \frac{d_v}{d_c} > 0$. It follows that there is a unique value of x in the region $[x_{\mathrm{BP}}, 1]$, call it x_A, such that $A(x_A) = 0$. We call $\epsilon(x_A)$ the *area threshold*, and write $\epsilon_A = \epsilon(x_A)$.

The Maxwell construction conjectures that the MAP threshold is equal to the area threshold, namely

$$\epsilon_{\mathrm{MAP}} = \epsilon_{\mathrm{A}}. \tag{10.10}$$

Thus the MAP threshold can be computed from the BP EXIT curve, which itself results from the density evolution equation.

In the previous paragraph we explained that the Maxwell construction applies to the BP EXIT curve parametrized by $0 \le x \le 1$. Here we introduce two closely related objects: EXIT *functions* which are well defined functions of ϵ. Take a code from the (d_v, d_c)-regular LDPC ensemble of length n. Let $\underline{X}$ denote the codeword, chosen uniformly at random from the set of all codewords and let $\underline{Y}$ be the received word when we transmit over a BEC with parameter ϵ. We set $Y_{\sim i} = (Y_1, \dots Y_{i-1}, Y_{i+1}, \dots Y_n)$ the observation vector *without* Y_i. Consider, $\widehat{x}_i^{\mathrm{BP}}(Y_{\sim i})$ and $\widehat{x}_i^{\mathrm{MAP}}(Y_{\sim i})$, the BP and bit-MAP estimates when bit i is not observed or erased.

DEFINITION 10.2 (BP and MAP EXIT functions) We define the BP EXIT

function as

$$h_{\mathrm{BP}}(\epsilon) = \lim_{t\to\infty} \lim_{n\to\infty} \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} \mathbb{P}[\widehat{x}_i^{\mathrm{BP},t}(Y_{\sim i}) = E]\right]. \tag{10.11}$$

where the expectation is over the code ensemble. This limit exists and is given by density evolution. It is explicitly given by a formula similar to the BP bit error probability, Equ. (6.23) *without the prefactor* ϵ because we do not take into account the observation Y_i. Similarly we define the MAP EXIT *function* as

$$h_{\mathrm{MAP}}(\epsilon) = \limsup_{n\to\infty} \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} \mathbb{P}[\widehat{x}_i^{\mathrm{MAP}}(Y_{\sim i}) = E]\right]. \tag{10.12}$$

where the expectation is again over the code ensemble. It can be shown that the limit exists, however this is not obvious and this is why the definition involves a limsup.

As the notations and terminology suggest the notions of BP EXIT curve and function are very closely related. This is best seen on figure 10.6. The "envelope" of the EXIT *curve* is precisely equal to the EXIT *function* $h^{\mathrm{BP}}(\epsilon)$ as a function of ϵ. Moreover, we can give an alternative definition of the area threshold directly

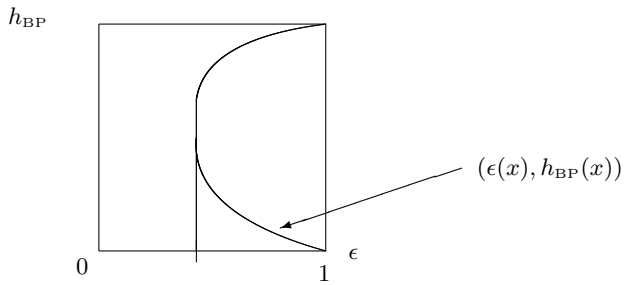


Figure 10.6 The BP EXIT curve $(\epsilon(x), h_{\mathrm{BP}}(x))$ and its envelope $h_{\mathrm{BP}}(\epsilon)$. The BP EXIT function vanishes for $0 \le \epsilon \le \epsilon_{\mathrm{BP}}$ and follows the branch of the BP EXIT curve corresponding to the non trivial stable fixed point for $\epsilon \le \epsilon_{\mathrm{BP}}$

in terms of the BP EXIT function. Indeed the area threshold is the solution of

$$\int_{\epsilon_A}^{1} d\epsilon\, h_{\mathrm{BP}}(\epsilon) = 1 - \frac{d_v}{d_c} \tag{10.13}$$

The proof of these facts is left as an exercise.

The two EXIT functions defined above satisfy the obvious but important inequality

$$h_{\mathrm{MAP}}(\epsilon) \le h_{\mathrm{BP}}(\epsilon). \tag{10.14}$$

Indeed, the MAP decoder is optimal in the sense that it minimizes the bit-error probability. In particular it has a smaller probability of error than the BP decoder for t iterations. This means $\mathbb{P}[\widehat{x}_i^{\mathrm{MAP}}(Y_{\sim i}) = E] \le \mathbb{P}[\widehat{x}_i^{\mathrm{BP},t}(Y_{\sim i}) = E]$.

The inequality then follows by summing over i, taking limits $n \to +\infty$ first and $t \to +\infty$ second.

Key to the proof of (10.10) is the following alternative represenation of the MAP EXIT function.

LEMMA 10.3 (Alternative formula for the MAP EXIT function) *Consider transmission over the BEC and let $\underline{X}$ and $\underline{Y}$ be the input and output vectors. The MAP EXIT function defined by* (10.12) *satisfies*

$$h_{\mathrm{MAP}}(\epsilon) = \limsup_{n\to+\infty} \frac{1}{n}\frac{d}{d\epsilon}\mathbb{E}\left[H(\underline{X}|\underline{Y}(\epsilon))\right] \tag{10.15}$$

Here the expectation is over the code ensemble.

Proof Representation (10.15) follows directly from the claim

$$\frac{1}{n}\frac{d}{d\epsilon}H(\underline{X}|\underline{Y}) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{P}[\widehat{x}_i^{\mathrm{MAP}}(Y_{\sim i}) = E]\,. \tag{10.16}$$

Note that this equation does not involve any average over the code ensemble so the lemma is in fact valid for a fixed finite length code. Now, to prove (10.16) assume for a moment that each bit i is transmitted over a BEC with independent parameter ϵ_i. We have

$$\begin{aligned}\frac{1}{n}\frac{d}{d\epsilon}H(\underline{X}|\underline{Y}(\epsilon)) &= \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial\epsilon_i}H(\underline{X}|\underline{Y}(\epsilon_1,\cdots,\epsilon_n))\big|_{\epsilon_i=\epsilon}\\ &= \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial\epsilon_i}H(X_i|\underline{Y}(\epsilon_1,\cdots,\epsilon_n))\big|_{\epsilon_i=\epsilon}\\ &= \frac{1}{n}\sum_{i=1}^{n}\mathbb{P}[\widehat{x}_i^{\mathrm{MAP}}(\underline{Y}_{\sim i}) = E]\end{aligned} \tag{10.17}$$

To get the second line we used the chain rule

$$\begin{aligned}H(\underline{X}\mid\underline{Y}) &= H(X_i\mid\underline{Y}) + H(\underline{X}_{\sim i}|X_i,\underline{Y})\\ &= H(X_i\mid\underline{Y}) + H(\underline{X}_{\sim i}|X_i,\underline{Y}_{\sim i}),\end{aligned} \tag{10.18}$$

where in (10.18) we dropped Y_i in $H(\underline{X}_{\sim i}|X_i,\underline{Y})$ since the channel is memoryless. Further, note $H(\underline{X}_{\sim i}|X_i,\underline{Y}_{\sim i})$ does not depend on ϵ_i so that this term does not contribute to the partial derivative with respect to ϵ_i. The partial derivative of $H(X_i\mid\underline{Y})$ with respect to ϵ_i then yields (10.17). To see this note that

$$\begin{aligned}H(X_i\mid\underline{Y}) &= \mathbb{P}[\widehat{x}_i^{\mathrm{MAP}}(\underline{Y}) = E)]\\ &= \mathbb{P}[Y_i = E]\mathbb{P}[\widehat{x}_i^{\mathrm{MAP}}(\underline{Y}) = E\mid Y_i = E] + \mathbb{P}[Y_i \neq E]\times 0\\ &= \epsilon_i\mathbb{P}[\widehat{x}_i^{\mathrm{MAP}}(Y_{\sim i}) = E]\end{aligned} \tag{10.19}$$

and also that $\mathbb{P}[\widehat{x}_i^{\mathrm{MAP}}(Y_{\sim i}) = E]$ is independent of ϵ_i. This settles (10.16). □

10.5 Proof of the Maxwell construction for the BEC: key ideas

We first have to clarify a fine point concerning various possible definitions of the MAP threshold. From an operational point of view the MAP threshold is the greatest possible channel parameter below which a MAP decoder can decode with high probability. Mathematically

$$\epsilon_{\text{MAP}} = \sup\{\epsilon \in [0,1] : \limsup_{n\to+\infty} \frac{1}{n}\mathbb{E}[\sum_{i=1}^{n} \mathbb{P}(\widehat{x}_i^{\text{MAP}}(\underline{Y}) \neq X_i)] = 0\}$$

Instead, we will use a definition of the MAP threshold that directly uses the entropy and has the advantage to connect more readily to the quantities that appear in our analysis.

DEFINITION 10.4 (MAP Threshold) The *MAP threshold* of the (d_v, d_c)-regular ensemble is defined by

$$\epsilon_{\text{MAP}} = \sup\{\epsilon \in [0,1] : \limsup_{n\to\infty} \frac{1}{n}\mathbb{E}[H(\underline{X} \mid \underline{Y}(\epsilon))] = 0\}.$$

□

Are the two definitions equivalent? From Fano's inequality one can show that if one transmits above the MAP threshold of definition 10.4 then the bit-MAP error probability is strictly positive. But, it is *not generally true* that transmitting below the MAP threshold defined in 10.4 implies the bit-MAP error probability vanishes as $n \to +\infty$. However one can show for the codes considered here that this *is in fact true*. Therefore for our purposes definition 10.4 and the operational one are in fact equivalent. At this point the reader may just accept these remarks. They are discussed further in the exercises.

Derivation of the upper bound: $\epsilon_{\text{MAP}} \leq \epsilon_{\text{A}}$

From inequality (10.14) and Lemma (10.3) we have

$$\limsup_{n\to+\infty} \frac{1}{n}\frac{d}{d\epsilon}\mathbb{E}\left[H(\underline{X}|\underline{Y})\right] \leq h_{\text{BP}}(\epsilon) \tag{10.20}$$

The essential idea of the proof is to integrate (10.20) over ϵ.

First we integrate the left hand side on an interval $[\bar{\epsilon}, 1]$ with $\bar{\epsilon} > \epsilon_{\text{BP}}$.

$$\begin{aligned}\int_{\bar{\epsilon}}^{1} d\epsilon \limsup_{n\to+\infty} \frac{1}{n}\frac{d}{d\epsilon}\mathbb{E}\left[H(\underline{X}|\underline{Y}(\epsilon))\right] &\geq \limsup_{n\to+\infty} \frac{1}{n}\int_{\bar{\epsilon}}^{1} d\epsilon \frac{d}{d\epsilon}\mathbb{E}\left[H(\underline{X}|\underline{Y}(\epsilon))\right] \\ &= \limsup_{n\to+\infty}\left\{\frac{1}{n}\mathbb{E}\left[H(\underline{X}|\underline{Y}(\epsilon=1))\right] - \mathbb{E}\left[H(\underline{X}|\underline{Y}(\bar{\epsilon}))\right]\right\} \\ &= (1-\frac{d_v}{d_c}) - \liminf_{n\to+\infty}\frac{1}{n}\mathbb{E}\left[H(\underline{X}|\underline{Y}(\bar{\epsilon}))\right] \end{aligned} \tag{10.21}$$

The lower bound follows from Fatou's lemma by observing that the integrand

is bounded. To get the last line we note that $\frac{1}{n}H(\underline{X}|\underline{Y}(\epsilon = 1))$ is equal to the logarithm of the number of codewords normalized by the blocklength. It is intuitive that the limit $n \to \infty$ of this quantity, averaged over the ensemble, is equal to the "design rate" of the code which is $1 - \frac{d_v}{d_c}$. The proof of this claim is purely combinatorial and we skip the steps here, but the result is valid for all (d_v, d_c)-regular ensembles with $2 \le d_v \le d_c$.

Let us now integrate the right hand side of (10.20). By the definition 10.1 of the trial entropy we find (recall that $x_\infty(\epsilon)$ is fixed point attained by DE iterations discussed in Section 6.7),

$$\begin{aligned}\int_{\bar\epsilon}^{1} d\epsilon\, h_{\text{BP}}(\epsilon) &= \int_{x_\infty(\bar\epsilon)}^{x_\infty(1)} d\epsilon(x)\, h_{\text{BP}}(x) \\ &= A(x_\infty(1)) - A(x_\infty(\bar\epsilon)) \\ &= (1 - \frac{d_v}{d_c}) - A(x_\infty(\bar\epsilon)) \end{aligned} \tag{10.22}$$

Putting together (10.20), (10.21) and (10.22) we obtain

$$\liminf_{n\to+\infty} \frac{1}{n}\mathbb{E}\left[H(\underline{X}|\underline{Y}(\bar\epsilon))\right] \ge A(x_\infty(\bar\epsilon)) \tag{10.23}$$

Finally take any $\bar\epsilon > \epsilon_A$. Then $A(x(\bar\epsilon)) > 0$, thus $\liminf_{n\to+\infty} \frac{1}{n}\mathbb{E}\left[H(\underline{X}|\underline{Y}(\bar\epsilon))\right] > 0$, which means $\bar\epsilon > \epsilon_{\text{MAP}}$. This implies $\epsilon_A \ge \epsilon_{\text{MAP}}$ as announced.

The last inequality (10.23) is a special case of a general inequality valid on any binary input memoryless symmetric channels. It is also closely related to a similar inequality that we will obtain by the interpolation method in Chapter 13.

Brief discussion of the lower bound: $\epsilon_{\text{MAP}} \ge \epsilon_{\text{A}}$

So far we have seen that the threshold given by the Maxwell construction is an upper bound on the MAP threshold. There are several ways of proving the reverse inequality. For the specific case at hand, namely transmission over the BEC, one can give a purely combinatorial proof. The idea is to prove that when $\epsilon < \epsilon_A$, with high probability, the matrix which we get if we start with the parity-check matrix and remove all columns which correspond to non-erased bits has rank equal to the number of erased bits. This shows that with high probability the codeword can be reconstructed by solving the corresponding linear system of equations, i.e., with high probability the MAP decoder succeeds. Since this proof is very specific to the erasure channel we skip it. There is a second more conceptual approach using spatial coupling and the interpolation technique which applies to general BMS channels. We will get back to this approach in part III.

10.6 Generalization to BMS channels[8]

The Maxwell construction can be generalized to BMS channels but proving that it yields the correct MAP threshold requires to develop more powerful tools as well considerable more work. We will come back to this question in part III. For the moment we give natural generalizations of the concepts of BP and MAP EXIT functions which allow us to state the conjecture. The formalism of Section 6.8 is used.

We consider transmission over "smooth" families of BMS channels c_ϵ parametrized by a noise level ϵ. Examples of such families are the BEC(ϵ), BSC(ϵ) or the BIAWGNC(ϵ). We will not formalise this notion of smoothness here except for saying that, roughly speaking we demand $H(\mathrm{c}_\epsilon \otimes \mathrm{x})$ is continuously differentiable in ϵ for all reasonable symmetric densities x (here H is the entropy functional given by Equ. (6.38)). Note that we ask for continuous differentiability with respect to the explicit ϵ dependence of the channel and not any other implicit ϵ dependence possibly hidden in x. Indeed in our applications the ϵ dependencies in the DE fixed points densities yield discontinuities.

Let us begin with the right notion of MAP EXIT *function*. The definition (10.12) is really only useful for the BEC and doesn't generalise well. At the same time, the identity (10.16) is only valid for the BEC. For more general channels it turns out that the identity (10.15) yields the good definition that we seek.

DEFINITION 10.5 (MAP EXIT function: general definition) Consider transmission over a smooth family of BMS channels. Let $\underline{X}$ denote the codeword, chosen uniformly at random from the set of all codewords and let $\underline{Y}$ be the received word. The general MAP EXIT function is defined as

$$g_{\mathrm{MAP}}(\epsilon) = \limsup_{n\to+\infty} \frac{1}{n}\frac{d}{d\epsilon}\mathbb{E}\left[H(\underline{X}|\underline{Y})\right] \tag{10.24}$$

where $\mathbb{E}$ is the average over the code ensemble. For the BEC this reduces to (10.12) by virtue of Lemma 10.3.

The MAP threshold is defined in 10.4 (this definition is valid in general provided we relax the range of the noise to $\epsilon \geq 0$). We immediately see $g_{\mathrm{MAP}}(\epsilon) = 0$ for $\epsilon < \epsilon_{\mathrm{MAP}}$. It is also intuitively clear that this function is monotone increasing for $\epsilon \geq \epsilon_{\mathrm{MAP}}$. For $d_v \geq 3$ it is discontinuous at ϵ_{MAP} which therefore represents a first order phase transition point.

We now introduce a BP EXIT function. A little thought shows that for the BEC this function is equal to

$$\lim_{t\to+\infty}(1-(1-x_t)^{d_c-1})^{d_v} = \lim_{t\to+\infty}\frac{\partial}{\partial\epsilon}H(\mathrm{c}_\epsilon \otimes (\mathrm{x}_t^{\boxplus d_c-1})^{\otimes d_v})$$

where x_t are the density evolution iterations initialised with $x_0 = 1$ (or $x_0 = \epsilon$) and $\mathrm{x}_t = x_t\Delta_0 + (1-x_t)\Delta_\infty$. In this expression the partial derivative indicates

[8] This section is not needed for the main development and can be skipped in a first reading.

that the derivative acts only on the *explicit* ϵ dependence in c_ϵ and not on the *implicit* dependence in x_t. The natural generalisation is

DEFINITION 10.6 (BP EXIT function: general definition) Consider transmission over a smooth family of BMS channels. The general BP EXIT function is defined as

$$g_{\mathrm{BP}}(\epsilon) = \lim_{t\to+\infty} \frac{\partial}{\partial\epsilon} H(\mathrm{c}_\epsilon \otimes (\mathrm{x}_t^{\boxplus d_c - 1})^{\otimes d_v}) \tag{10.25}$$

As above the partial derivative acts only on c_ϵ. For the BEC this reduces to (10.11).

Recall that at the end of Section 6.8 we defined the BP threshold as $\epsilon_{\mathrm{BP}} = \sup\{\epsilon : \mathrm{x}_\infty(\epsilon) = \Delta_\infty\}$. It is not difficult to see that Lemma 6.2 implies $g_{\mathrm{BP}}(\epsilon) = 0$ for $\epsilon < \epsilon_{\mathrm{BP}}$ and is monotone increasing for $\epsilon > \epsilon_{\mathrm{BP}}$ (for $d_v \geq 3$ the function is discontinuous at ϵ_{BP}). Thus, the easiest way to define the area threshold is to consider (10.13).

DEFINITION 10.7 (Area threshold for regular codes and BMS channels) Consider transmission over a smooth family of BMS channel with codes from the (d_v, d_c) regular ensemble with $2 \leq d_v < d_c$. We define the area threshold ϵ_A as that unique solution of the equation

$$\int_{\epsilon>\epsilon_A} d\epsilon\, g_{\mathrm{BP}}(\epsilon) = 1 - \frac{d_v}{d_c} \tag{10.26}$$

We are now ready to discuss the Maxwell construction in this general setting. This construction conjectures that $\epsilon_{\mathrm{MAP}} = \epsilon_A$ and therefore provides a constructive way to calculate the MAP threshold. We will see in Chapter 13 that, similarly to the case of the BEC, the upper bound $\epsilon_{\mathrm{MAP}} \leq \epsilon_A$ follows from a Lemma proved using a "correlation inequality" from statistical mechanics.

LEMMA 10.8 (Comparison of general EXIT functions) *The generalised BP and MAP EXIT functions satisfy*

$$g_{\mathrm{BP}}(\epsilon) \leq g_{\mathrm{MAP}}(\epsilon) \tag{10.27}$$

For the lower bound $\epsilon_{\mathrm{MAP}} \leq \epsilon_A$ we cannot resort to the combinatorial method used on the BEC. We will have to use the ideas of "interpolation method" and "spatial coupling" and developed in Chapters 13 and 14, 15. With these tools in our hands we will be able to sketch the proof of the following theorem.

THEOREM 10.9 (Maxwell construction) *Consider transmission over a smooth family of BMS channel with codes from the (d_v, d_c) regular ensemble with $2 \leq d_v < d_c$. For high enough but fixed degrees we have $\epsilon_{\mathrm{MAP}} = \epsilon_A$.*

It is believed that this theorem has much wider generality and in particular, as stated here, is valid for all degree pairs such that $d_c \geq 3$. We will also see in Chapter 15 that the proof yields much more. It provides us with a rather explicit variational expression for the conditional entropy $H(\underline{X}|\underline{Y}(\epsilon))$. This expression is

analogous to "one-letter" formulas of information theory and to "replica formulas" of spin glass theory.

10.7 Discussion

Besides the original liquid-vapor physical system, we have given two explicit examples of the Maxwell construction. For the CW model, the Maxwell construction appears somewhat like a coincidence (at least if one forgets about the physical interpretation). We first computed the exact relationship between average magnetization and the external field and then we computed the same relationship from a message-passing perspective. Comparing the two expressions we see that they are related by a Maxwell construction, just like in the original construction for an ideal gas.

Even more interesting is the situation if we cannot (or do not know how to) in fact compute the exact free energy expression but, starting with the message-passing formulation, can derive it using a Maxwell construction. This was the case for our second example, namely coding. There is currently essentially no other way of computing the MAP threshold. We have seen that the Maxwell construction gives us a guess of where this phase transition appears and we have also sketched some proof ideas. In the third part of these notes we will see how we can complete these proofs using the concepts of spatial coupling and the so-called interpolation method. So in this case, the Maxwell construction, together with further techniques, allows us to solve, what from a classical perspective seems to be a hard problem, namely rigorously compute the MAP threshold.

This is a general theme that applies to numerous other phase transitions. But, there is no trivial recipe for how to apply the Maxwell construction and how to prove that it is indeed correct. Each case requires some slightly different tricks and techniques. In fact, it is easy to construct examples (like K-SAT with belief propagation guided decimation) where the predictions given by a naive Maxwell construction are not even correct. But with a little bit of experience the Maxwell construction is a powerful paradigm.

10.8 Notes

The thesis of van der Waals in 1873 represented a fundamental step in the long and difficult process leading from the earlier thermodynamic and kinetic theories of gases of Boyle, Bernoulli, Gay-Lussac, Clausius and others to the description of states of matter by the statistical mechanical laws of atoms and molecules. Earlier experiments in the 19th century had already established that vapour can be changed into a liquid by continuous as well as discontinuous processes (second and first order phase transitions) and this prompted van der Waals to extend the gas theory of Clausius to the liquid state. He managed to correctly take into

account, at least qualitatively, the intermolecular forces, thereby also clarifying their nature and role. An english translation of the original work (Van der Waals 1873) and an account on its consequences on the physics of liquids well into the 20th century is found in (Van der Waals & Rowlinson 1988). Maxwell recognized the importance of this thesis and "corrected" it with his equal area construction. This construction appears very explicitly in (Maxwell 1875) and his explanation is still essentially unchanged in the modern textbooks. As always history is much more complicated, rich and fascinating and the interested reader may find an excellent account of it in (Brush 1983).

In a classic work of modern statistiscal mechanics (Lebowitz & Penrose 1966) the van der Waals-Maxwell has been shown to be rigorously exact for systems of particles interacting through forces containing a hard core part and an attractive part in the limit where the attraction becomes infinite range and at the same time infinitely weak. This is often called the Kac limit. Surprizingly similar limits also seem to be of relevance in coding theory and compressive sensing and we come to this question in Chapter 14.

The relevance of the Maxwell construction in coding theory was first discovered in (Méasson, Montanari & Urbanke 2004, Méasson, Montanari & Urbanke 2008). For Low-Density Parity-Check codes used for transmission over the binary erasure channel, a constructive relationship between the belief progation and MAP decoders is established through a new decoding algorithm called the "Maxwel decoder" and it is shown that the MAP threshold can be found from an equal area construction on the EXIT function (Equ. (10.10)). This approach was partly generalized to more general channels in (Méasson, Montanari, Richardson & Urbanke 2009) which introduced many useful tools such as generalized EXIT functions and proved Lemma 10.8. A different proof using correlation inequalities was provided in (Macris 2007*b*). The complete proof of Theorem 10.9 relies on more advanced tools introduced in part III and was finally carried out only quite later (Giurgiu, Macris & Urbanke 2016). The main ideas are sketched in Chapters 14 and 15.

Problems

10.1 Van der Waals equation, first and second order phase transitions. The goal of this exercise is to illustrate that phase transitions predicted by the van der Waals equation are perfectly analogous to the ones for ferromagnets predicted by the Curie-Weiss equation analysed in Chapter 4. Express the pressure p in Equ. (10.2) as a function of T and $v = V/N$ (volume per particle). This equation is analogous to the one expressing h as a function of T and m (see e.g., (4.24)).

(i) Let $v_L = V_L/N$ and $v_G = V/N$ the two extreme points of the Maxwell plateau of an isotherm. These are the volume per particle occupied by the liquid and vapour states when they coexist. Compute the critical parameters T_c, p_c, v_c such that the width of the plateau $v_G - v_L$ vanishes, in other words when the

van der Waals isotherm has a flat inflexion point. One finds $T_c = 8a/(27bk_B)$, $p_c = a/(27b^2)$, $v_c = 3b$.
(ii) Show that for $T > T_c$ and given p there is a single solution for v as a function of p. The volume per particle is a smooth function of p and there is no phase transition.
(iii) Show that for $T < T_c$ there are three solutions for the volume. According to the Maxwell construction as p increases (at fixed temperature) the volume is reduced discontinuously from v_G to v_L. This is a first order phase transition: the transition between the liquid and vapour states is discontinuous.
(iv) Show that as $T \to T_c$ from below $v_G - v_L \propto |T - T_c|^{1/2}$. This a second order phase transition behavior: the transition between the liquid and vapour states is continuous. The critical exponent is equal to $1/2$ and is the same as the one of the Curie-Weiss theory.
(v) Define the reduced variables $\bar{T} = T/T_c$, $\bar{p} = p/p_c$, $\bar{v} = v/v_c$. Show that the van der Waals equation becomes $(\bar{p} + 3/\bar{v}^2)(3\bar{v} - 1) = 8\bar{T}$. Although for each fluid the coefficients a and b are different and have to be determined experimentally, the van der Waals equation displays a universality which is well observed experimentally. In particular the critical behaviour at the second order phase transition is independent of the fluid.

10.2 Enveloppe of the BP EXIT curve. Starting from definition 10.11 for the BP EXIT function prove that it is the enveloppe of the parametric BP EXIT curve (10.8) as indicated on Figure 10.6. Moreover show that Definitions 10.1 and 10.13 for the area threshold are equivalent.

10.3 Details of proof of Lemma 10.3. If you have never done it before prove the chain rule used in (10.18). Starting from the definition of the MAP estimate in Chapter 3 for the BEC show in detail that $H(X_i|\underline{Y}) = \mathbb{P}[\widehat{x}_i^{\mathrm{MAP}}(\underline{Y}) = E]$, used in (10.19).

10.4 On the definition of the MAP threshold I. The goal of this exercise is to show that if we are transmitting above the MAP threshold defined according to 10.4 then the average bit error probability is non vanishing. Set $\mathrm{P}_i = \mathbb{P}(\widehat{x}_i^{\mathrm{MAP}}(\underline{Y}) \neq X_i)]$. Note that by the Fano inequality we have $H(X_i|\underline{Y}) \leq h_2(P_i)$ where h_2 is the binary entropy function. Use concavity of h_2 and subadditivity of $H(\underline{X}|\underline{Y})$ to show that

$$\frac{1}{n}\mathbb{E}\big[H(\underline{X}|\underline{Y})\big] \leq h_2\bigg(\mathbb{E}\big[\frac{1}{n}\sum_{i=1}^{n}\mathrm{P}_i\big]\bigg).$$

Deduce that if $\epsilon > \epsilon_{\mathrm{MAP}}$ there exist $\delta > 0$ independent of n such that

$$\mathbb{E}\big[\frac{1}{n}\sum_{i=1}^{n}\mathrm{P}_i\big] \geq h_2^{-1}(\delta).$$

In other words the average bit error probability remains strictly positive.

10.5 On the definition of the MAP threshold II. In general we cannot

conclude that if we are transmitting below the MAP threshold defined according to 10.4 then the average bit error probability vanishes. In other words from $\mathbb{E}[H(\underline{X} \mid \underline{Y})/n] \leq \delta$ does not in general imply that the average bit error probability is small. However this is the case if we have the slightly stronger condition $\mathbb{E}[\sum_{i=1}^{n} H(X_i \mid \underline{Y})/n] \leq \delta$. Let $\mathrm{P}_i = \mathbb{P}(\widehat{x}_i^{\mathrm{MAP}}(\underline{Y}) \neq X_i)]$. The goal of this exercise is to show that under the stronger condition we have $\frac{1}{n}\mathbb{E}[\sum_{i=1}^{n} \mathrm{P}_i] \leq \delta/2$.

First, convince yourself that under MAP decoding the error probability conditioned that we observed $\underline{y}$ is equal to $\min_x p(x \mid \underline{y})$. Then, prove the following statements

$$\begin{aligned}\frac{1}{n}\mathbb{E}[\sum_{i=1}^{n} H(X_i \mid \underline{Y})] &= \frac{1}{n}\mathbb{E}[\sum_{i=1}^{n} \mathbb{E}_{\underline{Y}}[h_2(\min_x p(x \mid \underline{Y}))]] \\ &\geq \frac{1}{n}\mathbb{E}[\sum_{i=1}^{n} \mathbb{E}_{\underline{Y}}[2\min_x p(x \mid \underline{Y})]] \\ &= \frac{2}{n}\mathbb{E}[\sum_{i=1}^{n} \mathrm{P}_i].\end{aligned}$$

The claim follows.

10.6 Large degree limit of the area threshold: BEC. Consider the limit of large degrees $d_v, d_c \to +\infty$ with a fixed ratio d_v/d_c. This means that the design rate $R = 1 - \frac{d_v}{d_c}$ of the code ensemble is fixed. Use (10.13) to show that in the large degree limit $\epsilon_A \to 1 - R$. Note that this means the area threshold tends to the Shannon threshold (or capacity) of the BEC in this limit.

10.7 Large degree limit of the area threshold: BMS channels. Using the tools introduced in Section 6.8 it can be shown that if x_n is a sequence of symmetric densities ordered by degradation (i.e. $\mathrm{x}_{n+1} \succ \mathrm{x}_n$) which tends to x_* in the sense $d(\mathrm{x}_n, \mathrm{x}_*) = H(\mathrm{x}_n - \mathrm{x}_*) \to 0$ as $n \to +\infty$, then

$$\lim_{n\to+\infty} \frac{\partial}{\partial\epsilon} H(\mathrm{c}_\epsilon \otimes \mathrm{x}_n) = \frac{\partial}{\partial\epsilon} H(\mathrm{c}_\epsilon \otimes \mathrm{x}_*)$$

provided c_ϵ is sufficiently "smooth" (e.g. see the next exercise). It follows that if x_∞ is the limit of density evolution iterations

$$g_{\mathrm{BP}}(\epsilon) = \frac{\partial}{\partial\epsilon} H(\mathrm{c}_\epsilon \otimes (\mathrm{x}_\infty^{\boxplus d_c - 1})^{\otimes d_v}).$$

Now, consider the large degree limit as in the previous exercise and show that $\mathrm{x}_\infty \to \Delta_0$ as $d_v, d_c \to +\infty$ with fixed ratio d_v/d_c. Deduce that the area threshold satisfies

$$\lim_{d_v, d_c \to +\infty} H(\mathrm{c}_{\epsilon_A}) = 1 - R.$$

Verify this means ϵ_A tends to the Shannon threshold of the channel (equivalently $1 - H(\mathrm{c}_\epsilon)$ is the capacity of a BMS).

10.8 A smoothness condition on degraded channel families. Let c_ϵ

a degraded channel family, i.e., $\mathrm{c}_{\epsilon'} \succ \mathrm{c}_\epsilon$ for $\epsilon' > \epsilon$. We want to show that the regularity condition (κ a strictly positive numerical constant)

$$H((\mathrm{c}_{\epsilon'} - \mathrm{c}_\epsilon)^{\otimes 2})^{1/2} \leq \kappa|\epsilon' - \epsilon|$$

is sufficient for

$$\lim_{n\to+\infty} \frac{\partial}{\partial\epsilon} H(\mathrm{c}_\epsilon \otimes \mathrm{x}_n) = \frac{\partial}{\partial\epsilon} H(\mathrm{c}_\epsilon \otimes \mathrm{x}_*)$$

to hold where $\mathrm{x}_{n+1} \succ \mathrm{x}_n$ and $\lim_{n\to+\infty} H(x_n - x_*) = 0$.

Use the duality rule, the moment expansion (see Section 6.8) and the Cauchy-Schwarz inequality to show that

$$H((\mathrm{c}_{\epsilon'} - \mathrm{c}_\epsilon) \otimes (\mathrm{x}_n - \mathrm{x}_\infty)) \leq H((\mathrm{c}_{\epsilon'} - \mathrm{c}_\epsilon)^{\otimes 2})^{1/2} H((\mathrm{x}_n - \mathrm{x}_\infty)^{\otimes 2})^{1/2}.$$

Deduce the claim from this inequality.

10.9 BP EXIT FUNCTION FOR GENERAL BMS CHANNELS. Consider a smooth family of degraded channels where smoothness is defined as in the previous exercise. Consider the definition 10.6 of the BP EXIT function and show that the limit: (i) equals $\frac{\partial}{\partial\epsilon} H(\mathrm{c}_\epsilon \otimes (\mathrm{x}_\infty^{\boxplus d_c - 1})^{\otimes d_v})$; (ii) vanishes for $\epsilon < \epsilon_{\mathrm{BP}}$; (iii) is monotone increasing for $\epsilon > \epsilon_{\mathrm{BP}}$ (use the moment expansion).

Part III

From Algorithms to Optimality

11 The Bethe Free Energy

We have discussed how we can analyze the performance of various low-complexity algorithms, in particular algorithms of message-passing type. We have seen that in the limit of infinite system size, such algorithms have thresholds and we were able to characterize these thresholds quantitatively. Such thresholds are often called *dynamical* thresholds since they are associated to the *dynamics* of a process (for us this process is the algorithm).

But there is typically also a *static* phase transition. This corresponds to a phase transition which describes a change in the properties of the system itself, independent of any algorithmic question. For example, in coding we can ask how much noise we can add so that with high probability there is a unique codeword which is "compatible" with the received information. In communicatons language, this corresponds to the MAP threshold. For compressive sensing we can ask how the number of measurements has to scale with the number of unknowns so that with high probability there is a unique sparse vector which is compatible with the measurements. Finally, in K-SAT we can ask how many constraints we can have per Boolean variable so that with high probability a random formula is satisfiable. This is usually refered to as the SAT-UNSAT threshold.

Why are we interested in these quantities? Some systems are given to us and we cannot change them (e.g., K-SAT). In this case it is important to know how well a computationally unbounded system could in principle do in order to gauge how well our algorithm is performing. But often we are actually in control of the system itself. Think of the coding problem or also compressive sensing. It is typically us who designs the code or the measurement matrix. So in these cases it is important to know that the system itself is designed in such a way that at least in principle (if we had unbounded computational resources at our disposal) it has a good performance (comparable to the optimal one). E.g., in coding we can then compare the MAP threshold to the ultimate limit, namely the Shannon threshold and hopefully these two thresholds are close.

As we will see, there are two basic themes which appear. First, static phase transition thresholds are in general much harder to compute than the dynamical ones. In a few cases we will be able to derive rigorous quantitative statements. In some other ones, we will have to be content with computations which are believed to yield the correct value but fall short of a mathematical proof. The second, perhaps more surprising theme is that the analysis of the static phase

transition threshold can often be done by looking at the behavior of the message-passing algorithm! Why message-passing, a sub-optimal algorithm, should have any bearing on the behavior of the optimal algorithm is at first glance puzzling.

As we will see, the key object which connects these two themes is the so-called Bethe free energy. It is an "approximation" to the true free energy which itself depends on the fixed points of the message-passing algorithm. In some instances the static thresholds predicted by the Bethe free energy can be shown to be indeed correct.

Let us discuss this in more detail. Computing the true free energy for typical statistical mechanics models (or general graphical models) is an impossible task. An important approximation philosophy is the so-called "mean-field theory." In this theory, when looking at the interactions of a "spin" with the rest of the system, we only take into account very close neighbors exactly, while we model influences of the remaining system simply by a "mean field," i.e., a field which accounts for the average influence of the remaining of the system. For models defined on sparse graphs that are locally tree-like, a very good form of mean field theory was developed by Bethe and Peierls. This leads to the so-called Bethe free energy approximation. We note that this is already a "sophisticated" version of the most basic mean field theory.

As we will see the Bethe-Peierls theory involves fixed point equations that are the same as those occurring in Belief-Propagation. Their use, and to some extent interpretation, are however different. Note that the clash of initials (BP) is solely due to an historical accident. We hope that this will not cause major confusions.

In this chapter we treat in complete detail the case of graphical models with a discrete alphabet $\mathcal{X}$. As a direct application we will look more closely at the cases of coding and K-SAT. For models with a continuous alphabet such as those occuring in the context of compressive sensing the ideas are conceptually the same, but the calculations have to be slightly adapted. We consider a general Gibbs measure of the form

$$\mu\left(\underline{x}\right) = \frac{1}{Z}\prod_{a} f_a\left(x_{\partial a}\right), \tag{11.1}$$

where the variables $x_i \in \mathcal{X}$, $i = 1, \ldots, n$ and f_a, $a = 1, \ldots, m$ are kernel functions associated to check nodes which depend on $x_{\partial a} = \{x_i, i \in \partial a\}$. In Chapter 5 we discussed the sum-product algorithm that computes *BP-marginals* for such measures. Recall when the graph is a tree these are the *exact marginals*. Similarly we will see that on a tree the free energy

$$f = -\frac{1}{\beta n}\ln Z, \tag{11.2}$$

can be expressed exactly in terms of the marginals of the measure. This is the starting point of the formalism developped in this chapter.

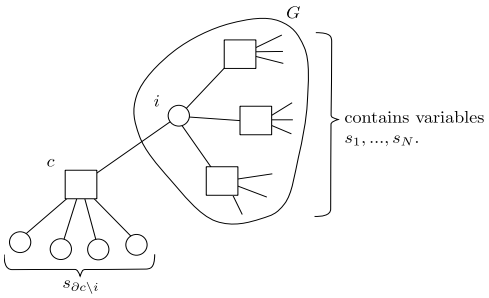


Figure 11.1 Induction procedure: G is the original tree to which we add factor c connected to i such that the new graph is a tree

11.1 The Gibbs measure on trees

Consider the (exact) marginals

$$\nu_i(x_i) = \sum_{\sim x_i} \mu(\underline{x}), \qquad \nu_a(x_{\partial a}) = \sum_{\sim x_{\partial a}} \mu(\underline{x}).$$

As explained in Chapter 5 on a tree these can be computed exactly by the sum-product algorithm. More is true.

LEMMA 11.1 *The Gibbs measure on a tree can be expressed in terms of its marginals as follows,*

$$\mu(\underline{x}) = \prod_a \nu_a(x_{\partial a}) \prod_i (\nu_i(x_i))^{1-d_i} \tag{11.3}$$

where d_i is the degree of variable node i.

Proof We prove (11.3) by induction over number the number m of factor nodes. For $m = 1$ the unique clause is connected to variable nodes with $d_i = 1$. Thus (11.3) is trial in this case. Now, we assume (11.3) is true for a tree graph G with m check nodes and prove that it also holds for the new Gibbs measure with $m + 1$ factor nodes, obtained when one adds one factor node c connected to a variable node i in such a way that the new graph is a tree (we do not discuss the somewhat trivial case where the new factor is disconnected). The original tree G and the new tree are depicted on figure 11.1. The new measure reads

$$\mu_{\text{new}}(x_{\partial c \setminus i}, \underline{x}) = \frac{1}{Z_{\text{new}}} f_c(x_{\partial c}) \prod_a f_a(x_{\partial a}) \tag{11.4}$$

Consider the conditional probability $\mathbb{P}(x_{\partial c \setminus i} \mid \underline{x})$ of an assignement $x_{\partial c \setminus i}$ given $x_1, \ldots, x_n$. We observe that

$$\mathbb{P}(x_{\partial c \setminus i} \mid \underline{x}) = \mathbb{P}(x_{\partial c \setminus i} \mid x_i) = \frac{\nu_{\text{new},c}(x_{\partial c})}{\nu_{\text{new},i}(x_i)},$$

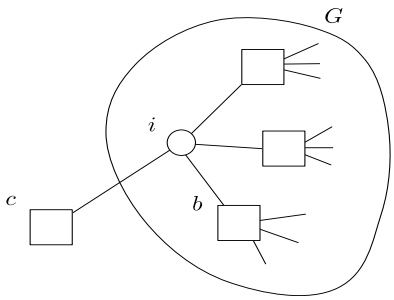


Figure 11.2 Factor graph for the marginal distribution (11.6). We select an arbitrary check $b \in \partial i \setminus c$.

where $\nu_{,c}$, $\nu_{,i}$ stand for marginals of μ_{new}. Therefore

$$\begin{aligned}\mu_{\text{new}}\left(x_{\partial c\setminus i}, \underline{x}\right) &= \Pr\left(x_{\partial c\setminus i} \mid \underline{x}\right) \nu_{\text{new}}\left(\underline{x}\right) \\ &= \nu_{\text{new},c}\left(x_{\partial c}\right)\left(\nu_{\text{new},i}\left(x_i\right)\right)^{-1}\nu_{\text{new}}\left(\underline{x}\right). \end{aligned} \tag{11.5}$$

Now, the marginal of μ_{new} over $x_{\partial c\setminus i}$ is by definition

$$\begin{aligned}\nu_{\text{new}}\left(\underline{x}\right) &= \frac{1}{Z_{\text{new}}}\sum_{x_{\partial c\setminus i}} f_c\left(x_{\partial c}\right)\prod_a f_a\left(x_{\partial a}\right) \\ &= \frac{1}{Z_{\text{new}}}\widetilde{f}_c\left(x_i\right)\prod_a f_a\left(x_{\partial a}\right). \end{aligned} \tag{11.6}$$

where we have set $\sum_{x_{\partial c\setminus i}} f_c\left(x_{\partial c}\right) = \widetilde{f}_c\left(x_i\right)$. Distribution (11.6) has the factor graph depicted on figure 11.2 This tree still has $m+1$ check nodes. However c can be "absorbed" in any arbitrarily selected check $b \in \partial i \setminus c$,

$$\begin{aligned}\nu_{\text{new}}\left(\underline{x}\right) &= \frac{1}{Z_{\text{new}}}\widetilde{f}_c\left(x_i\right)\prod_a f_a\left(x_{\partial a}\right) \\ &= \frac{1}{Z_{\text{new}}}\widetilde{f}_c\left(x_i\right) f_b\left(x_{\partial b}\right)\prod_{a\neq b} f_a\left(x_{\partial a}\right) \\ &= \frac{1}{Z_{\text{new}}}\widetilde{f}_b\left(x_{\partial b}\right)\prod_{a\neq b} f_a\left(x_{\partial a}\right) \end{aligned}$$

where we have set $\widetilde{f}_c\left(x_i\right) f_b\left(x_{\partial b}\right) = \widetilde{f}_b\left(x_{\partial b}\right)$. We recognize this last expression as a Gibbs measure defined on a tree with m check nodes, so that we can apply the induction hypothesis to ν_{new},

$$\nu_{\text{new}}(x_1,\ldots,x_n) = \prod_a \nu_{\text{new},a}\left(x_{\partial a}\right)\prod_i\left(\nu_{\text{new},i}\left(x_i\right)\right)^{1-d_i}. \tag{11.7}$$

Here $\nu_{\text{new},a}$ and $\nu_{\text{new},i}$ are the marginals of ν_{new}. But clearly, they are also the marginals of ν_{new} in (11.4). Combining (**??**) with (11.5) yields the desired result for the Gibbs measure (11.4) with $m+1$ factor nodes. □

11.2 The free energy on trees

We first recall a general and important expression (not restricted to trees) for the free energy which we have seen in Chapter **??**. This formula is best expressed when the Gibbs measure (11.1) is represented in its traditional physics form

$$\mu\left(\underline{x}\right)=\frac{1}{Z}\exp\left(-\beta\mathcal{H}\left(\underline{x}\right)\right). \tag{11.8}$$

where the relation between the Hamiltonian and the kernel functions is

$$\beta\mathcal{H}\left(\underline{x}\right)=-\sum_{a}\ln f_{a}(x_{\partial a}) \tag{11.9}$$

Replacing (11.8) in the definition of the free energy (11.2) one easily finds for the un-normalized free energy $F\equiv nf$,

$$F=\langle\mathcal{H}\rangle-\beta^{-1}S\left[\mu\right] \tag{11.10}$$

where

$$\begin{aligned}\langle\mathcal{H}\rangle&=\sum_{\underline{x}}\mathcal{H}\left(\underline{x}\right)\mu\left(\underline{x}\right)\\ S\left[\mu\right]&=-\sum_{\underline{x}}\mu\left(\underline{x}\right)\ln\mu\left(\underline{x}\right).\end{aligned}$$

Here $\langle\mathcal{H}\rangle$ is the Gibbs average of the Hamiltonian. Physically it represents the total average internal energy that the system possesses, and is commonly called the internal energy. $S[\mu]$ is called the Gibbs entropy. This is nothing else than Shannon's entropy written down for the Gibbs measure.

We now apply relation (11.10) to the Gibbs measure on a tree graph. This leads to

THEOREM 11.2 *On a tree graphical model the (un-normalized) free energy $F=nf$ can be expressed in terms of its marginals as*

$$F=\sum_{a}\sum_{x_{\partial a}}\nu_{a}\left(x_{\partial a}\right)\ln\frac{\nu_{a}\left(x_{\partial a}\right)}{f_{a}\left(x_{\partial a}\right)}+\sum_{i}\left(1-d_{i}\right)\sum_{x_{i}}\nu_{i}\left(s_{i}\right)\ln\nu_{i}\left(x_{i}\right) \tag{11.11}$$

Proof Using (11.9) the internal energy contribution yields

$$\begin{aligned}\langle\mathcal{H}\rangle_{\mu}&=-\sum_{a}\sum_{\underline{x}}\mu\left(\underline{x}\right)\ln f_{a}(x_{\partial a})\\ &=-\sum_{a}\sum_{x_{\partial a}}\nu\left(x_{\partial a}\right)\ln f_{a}\left(x_{\partial a}\right).\end{aligned}$$

This expession for the internal energy is completely general and does not depend on having a tree graph. To compute the contribution of the entropy we use (11.3)

in lemma 11.1. This gives

$$\begin{aligned} S[\mu] &= -\sum_a \sum_{\underline{x}} \mu(\underline{x}) \left(\ln \nu_a(x_{\partial a})\right) \\ &\quad + \sum_i (1-d_i) \sum_{\underline{x}} \mu(\underline{x}) \ln\left(\nu_i(x_i)\right) \\ &= -\sum_a \sum_{x_{\partial a}} \nu_a(x_{\partial a}) \ln \nu_a(x_{\partial a}) + \sum_i (1-d_i) \sum_{x_i} \nu_i(x_i) \ln \nu_i(x_i) \end{aligned}$$

This expression for the entropy is exact only for tree graphs. Combining the energetic and entropic contributions yields (11.11) □

In chapter 5 we learned how to compute the marginals in terms of exact message passing equations on the tree. We have two types of messages: those flowing from variable to check nodes $\mu_{i\to a}(x_i)$ and those flowing from check to variables node $\mu_{a\to i}(x_i)$. The exact marginals are given by,

$$\begin{cases} \nu_i(x_i) = \frac{\prod_{a\in\partial i} \hat{\mu}_{a\to i}(x_i)}{\sum_{x_i} \prod_{a\in\partial i} \hat{\mu}_{a\to i}(x_i)} \\ \nu_a(x_{\partial a}) = \frac{f_a(x_{\partial a}) \prod_{i\in\partial a} \mu_{i\to a}(x_i)}{\sum_{x_{\partial a}} f_a(x_{\partial a}) \prod_{i\in\partial a} \mu_{i\to a}(x_i)}. \end{cases} \tag{11.12}$$

and the messages by the sum-product equations by

$$\begin{cases} \mu_{i\to a}(x_i) & = \prod_{b\in\partial i\setminus a} \hat{\mu}_{b\to i}(x_i) \\ \hat{\mu}_{a\to i}(x_i) & = \sum_{\sim x_i} f_a(x_{\partial a}) \prod_{j\in\partial a\setminus i} \mu_{j\to a}(x_j) \end{cases} \tag{11.13}$$

Moreover the messages are uniquely defined by their "initial" values at the leaf nodes. From a leaf factor the outgoing message equals $f_a(x_{\partial a})$, and from a leaf variable it equals 1.

Using these expressions in (11.11), a straightforward calculation (exercise) leads to the alternative expression for the free energy

THEOREM 11.3 *On a tree graphical model the (un-normalized) free energy $F = nf$ can be expressed in terms of the BP messages as a sum of three contributions associated to variable nodes, check nodes and edges*

$$F = \sum_i F_i + \sum_a F_a - \sum_{(i,a)} F_{ia},$$

where the three contributions are

$$\begin{aligned} F_i &= \ln\left\{ \sum_{x_i} \prod_{b\in\partial i} \hat{\mu}_{b\to i}(x_i) \right\} \\ F_a &= \ln\left\{ \sum_{x_{\partial a}} f_a(x_{\partial a}) \prod_{i\in\partial a} \mu_{i\to a}(x_i) \right\} \\ F_{ia} &= \ln\left\{ \sum_{x_i} \mu_{i\to a}(x_i)\, \hat{\mu}_{a\to i}(x_i) \right\} \end{aligned}$$

It is worth to note that in this formula messages do not have to be normalized. Indeed they were not normalized in the first place in the sum-product equations. This can also be seen by checking F is invariant under the renormalizations $\hat{\mu}_{a\to i} \to \hat{z}_{a\to i}\,\hat{\mu}_{a\to i}$ and $\mu_{i\to b} \to \hat{z}_{i\to a}\,\mu_{i\to a}$ for any arbitrary numbers $\hat{z}_{a\to i}$ and $z_{i\to a}$.

11.3 Bethe free energy for general graphical models

We now turn our attention to general graphical models of the type (11.1) with a factor graph that is not necessarily a tree. We assign to each edge two ditributions $\mu_{i\to a}(s_i)$ and $\hat{\mu}_{a\to i}(s_i)$. The set of all distributions forms two vectors denoted by $\underline{\mu}$ and $\underline{\hat{\mu}}$ indexed by the directed edges $\{i \to a\}$ and $\{a \to i\}$. The notation is the same than for the BP messages for reasons that will become clear, however the reader should bear in mind that conceptually these are general distributions, not necessarily equal to the BP messages (for one thing the BP equations do not necessarily have a unique solution).

The *Bethe free energy* is, by definition, the functional (or function)

$$F_{\text{Bethe}}\left[\underline{\mu},\underline{\hat{\mu}}\right] = \sum_i F_i\left[\{\mu_{i\to b}, b\in\partial i\}\right] + \sum_a F_a\left[\{\mu_{i\to a}, i\in\partial a\}\right] - \sum_{ai} F_{ai}\left[\{\mu_{i\to a},\hat{\mu}_{a\to i}\}\right]. \tag{11.14}$$

with the three contributions associated to variable and check nodes, and edges.

$$F_i = \ln\left\{\sum_{x_j}\prod_{b\in\partial i}\hat{\mu}_{b\to i}\left(x_i\right)\right\} \tag{11.15}$$

$$F_a = \ln\left\{\sum_{x_{\partial a}} f_a\left(x_{\partial a}\right)\prod_{i\in\partial a}\mu_{i\to a}\left(x_i\right)\right\} \tag{11.16}$$

$$F_{ai} = \ln\left\{\sum_{x_i}\mu_{j\to a}\left(s_i\right)\hat{\mu}_{a\to j}\left(x_i\right)\right\}. \tag{11.17}$$

What is the idea behind this definition? The Bethe free energy exactly gives the true free energy for factor graphs that are trees. For a loopy factor graph it may seem a reasonable idea to propose the Bethe free energy as an educated guess that hopefully approximates well the true free energy.

However a crucial issue immediately arises. Since the BP equations do not necessarily have a unique solution (for loopy graphs) how does one choose the messages ? The rule of thumb is to take the messages that minimize the Bethe free energy functional. This rule is motivated by the variational principle (see Chapter 2) where the variational ansatz for the free energy must be minimized over the set of trial parameters in order to find the best possible approximation to the true free energy. However the Bethe free energy *does not* satisfy the variational principle for the simple reason that for general graphs the right hand side in

(11.3) is not a normalized probability distribution. Therefore, there is no apriori fundamental justification for choosing the messages as the ones that minimize the Bethe free energy functional. Nevertheless in practice this is often a successfull idea.

The discussion above suggests that a first important step is to look at stationnary points of the Bethe functional. One then discovers the following important result.

THEOREM 11.4 *The stationary points of the Bethe free energy satisfy the sum-product message passing equations and conversely the solutions of the sum-product equations are stationary points of the Bethe free energy.*

Proof For a finite system with a discrete alphabet the Bethe free energy functional is really a function of many variables, namely $\mu_{i\to a}(x_i)$, $\hat{\mu}_{a\to i}(x_i)$ for $x_i \in \mathcal{X}$. Thus the stationnarity conditions are simply

$$\frac{\partial F_{\text{Bethe}}}{\partial \mu_{i\to a}(x_i)} = 0, \qquad \frac{\partial F_{\text{Bethe}}}{\partial \widehat{\mu}_{a\to i}(x_i)} = 0$$

For the first derivative there is a contribution from F_a and F_{ia},

$$\frac{\partial F_{\text{Bethe}}}{\partial \mu_{i\to a}(x_i)} = \frac{\hat{\nu}_{a\to i}(x_i)}{\sum_{x_i} \mu_{i\to a}(x_i)\,\hat{\mu}_{a\to i}(x_i)} - \frac{\sum_{\sim x_i} f_a(x_{\partial a}) \prod_{j\in\partial a\setminus i} \mu_{j\to a}(x_j)}{\sum_{x_{\partial a}} f_a(x_{\partial a}) \prod_{j\in\partial a} \mu_{j\to a}(x_j)},$$

and for the second one the contribution comes from F_i and F_{ia},

$$\frac{\partial F_{\text{Bethe}}}{\partial \widehat{\mu}_{a\to i}(x_i)} = \frac{\nu_{i\to a}(x_i)}{\sum_{x_i} \mu_{i\to a}(x_i)\,\hat{\mu}_{a\to i}(x_i)} - \frac{\prod_{b\in\partial i\setminus a} \hat{\mu}_{b\to i}(x_i)}{\sum_{x_i} \prod_{b\in\partial i} \hat{\mu}_{b\to i}(x_i)}.$$

If we set the two derivatives to zero we find

$$\widehat{\mu}_{a\to i}(x_i) \propto \sum_{\sim x_i} f_a(x_{\partial a}) \prod_{j\in\partial a\setminus i} \mu_{j\to a}(x_j)$$
$$\mu_{i\to a}(x_i) \propto \prod_{b\in\partial i\setminus a} \widehat{\mu}_{b\to i}(x_i).$$

which are equivalent to the sum-product equations. Conversely it is easy to revert these calculations and show that the sum-product equations imply the stationnarity condition. □

11.4 Ising model on a random k-regular graph

11.5 Application to coding

We saw in Chapter 5 that the posterior measure used for MAP decoding is

$$\frac{1}{Z(\underline{h})} \prod_a \frac{1}{2}(1 + \prod_{i\in\partial a} s_i) \prod_{i=1}^n e^{h_i s_i}.$$

where $s_i \in \mathcal{X} = \{-1, +1\}$ (recall the change of variables $s_i = (-1)^{x_i}$, $x_i \in \{0, 1\}$). There are two types of kernel functions

$$f_i(s_i) = e^{h_i s_i}, \quad \text{and} \quad f_a(\{s_i, i \in \partial a\}) = \frac{1}{2}(1 + \prod_{i \in \partial a} s_i), \tag{11.18}$$

associated to leaf checks (representing channel observations) and usual parity checks. An example with the corresponding factor graph is shown in figure 5.6.

The messages flowing on edges connecting variable nodes and parity checks are

$$\mu_{i \to a}(s_i) \propto e^{h_{i \to a} s_i}, \qquad \widehat{\mu}_{a \to i}(s_i) \propto e^{\widehat{h}_{a \to i} s_i} \propto 1 + s_i \tanh \widehat{h}_{a \to i}.$$

The messages flowing on edges connecting leaf checks and variable nodes are

$$e^{h_i s_i}, \qquad \prod_{a \in \partial i} e^{\widehat{h}_{a \to i} s_i} \propto \prod_{a \in \partial i} (1 + s_i \tanh \widehat{h}_{a \to i}).$$

As pointed out above the normalization factors of the messages cancel out in the Bethe free energy, and this is why our parametrization only involves proportionality relations.

Replacing these messages in expressions (11.15)-(11.17) it is possible to perform exactly all sums over the spins, and express the Bethe free energy as a function of $(\underline{h}, \underline{\hat{h}}) = \{h_{i \to a}, \hat{h}_{a \to i}\}$. We give the main steps of this calculation.

From (11.15) the contribution of variable nodes is

$$\begin{aligned} F_i &= \ln\Big\{ \sum_{s_i = \pm 1} e^{h_i s_i} \prod_{a \in \partial i} (1 + s_i \tanh \hat{h}_{a \to i}) \Big\} \\ &= \ln\Big\{ e^{h_i} \prod_{a \in \partial i} (1 + \tanh \hat{h}_{a \to i}) + e^{-h_i} \prod_{a \in \partial i} (1 - \tanh \hat{h}_{a \to i}) \Big\}. \end{aligned} \tag{11.19}$$

From (11.16), for parity checks we have

$$F_a = \ln\Big\{ \sum_{s_{\partial a}} \frac{1}{2}(1 + \prod_{i \in \partial a} s_i) \prod_{i \in \partial a} (1 + s_i \tanh h_{i \to a}) \Big\}.$$

Observe that

$$\begin{aligned} \sum_{s_{\partial a}} \prod_{i \in \partial a} (1 + s_i \tanh h_{i \to a}) &= \prod_{i \in \partial a} \sum_{s_i = \pm 1} (1 + s_i \tanh h_{i \to a}) \\ &= 2^{|\partial a|} \end{aligned}$$

and

$$\begin{aligned} \sum_{s_{\partial a}} \prod_{i \in \partial a} s_i \prod_{i \in \partial a} (1 + s_i \tanh h_{i \to a}) &= \prod_{i \in \partial a} \sum_{s_i = \pm 1} (s_i + \tanh h_{i \to a}) \\ &= 2^{|\partial a|} \prod_{i \in \partial a} \tanh h_{i \to a}. \end{aligned}$$

Thus the contribution of parity checks is

$$F_a = \ln\left\{\frac{1}{2}(1+\prod_{i\in\partial a}\tanh h_{i\to a})\right\} + |\partial a|\ln 2. \tag{11.20}$$

The careful reader will also note that there is a contribution from leaf checks that happens to be given by (11.19), and also happens to cancel with the contribution of edges connecting variable and leaf check nodes.

There remains the contribution of edges connecting variable and parity check nodes

$$\begin{aligned} F_{ai} &= \ln\left\{\sum_{s_i=\pm 1}(1+s_i\tanh h_{i\to a})(1+s_i\tanh\hat{h}_{a\to i})\right\} \\ &= \ln\left\{1+\tanh h_{i\to a}\tanh\hat{h}_{a\to i}\right\} + \ln 2. \end{aligned} \tag{11.21}$$

Finally, the Bethe free energy is given by the sum of the three types of contributions (11.19), (11.20) and (11.21)

$$\begin{aligned} F_{\text{Bethe}}(\underline{h},\underline{\hat{h}}) = &\sum_i \ln\left\{e^{hi}\prod_{a\in\partial i}(1+\tanh\hat{h}_{a\to i}) + e^{-h_i}\prod_{a\in\partial i}(1-\tanh\hat{h}_{a\to i})\right\} \\ &+\sum_a \ln\left\{\frac{1}{2}(1+\prod_{j\in\partial a}\tanh\ h_{j\to a})\right\} \\ &+\sum_{ai}\ln\left\{1+\tanh h_{i\to a}\tanh\hat{h}_{a\to i})\right\} \end{aligned} \tag{11.22}$$

From this expression it is straightforward to check explicitly that the stationarity condition

$$\frac{\partial F_{\text{Bethe}}}{\partial h_{i\to a}} = \frac{\partial F_{\text{Bethe}}}{\partial \hat{h}_{a\to i}} = 0$$

is equivalent to the BP fixed equations

$$\begin{cases} h_{i\to a} = h_i + \sum_{b\in\partial i\setminus a}\hat{h}_{b\to i}, \\ \hat{h}_{a\to i} = \tanh^{-1}\left\{\prod_{j\in\partial a\setminus i}\tanh\ h_{j\to a}\right\}. \end{cases}$$

This proves again Theorem 11.4 in the particular case of coding.

We will see that the average over the channel outputs and the graph ensemble of the Bethe free energy allows to derive the so-called replica-symmetric (RS) formula for the average free energy[1]. It is known that for a large class of LDPC codes and BMS channels the RS free energy is equal to the exact free energy. In particular it allows to correctly predict the MAP noise threshold. In the next

[1] The adjective "replica-symmetric" is due to historical reasons. indeed these formulas were first derived thanks to the so-called replica method which we do not cover in this course. The approach of the replica method is algebraic in nature but mathematically more mysterious.

chapters we will derive the RS formula with the specific application of the BEC in mind, and partly prove that the RS formula is exact.

11.6 Interlude: Thouless-Anderson-Palmer free energy

To do start with Bethe and simplify down to TAP.

11.7 Application to compressive sensing

To do: start with Bethe free energy and simplify down to TAP like free energy.

11.8 Application to K-SAT

Recall from Chapter 3 the partition function of K-SAT at finite temperature (here again $s_i = (-1)^{x_i}$, $x_i \in \{0,1\}$)

$$Z = \sum_{s_1,\dots,s_n \in \{-1,+1\}^n} \prod_{a=1}^{m} \Big(1 - (1 - e^{-\beta}) \prod_{i \in \partial a} (\frac{1 + s_i J_{ia}}{2}) \Big). \tag{11.23}$$

The Bethe free energy here serves as a first guess for $-(\beta n)^{-1} \ln Z$. Recall that for $\beta = +\infty$, Z counts the number of solutions. Under the proviso that there exist at least one solution $\ln Z$ is well defined for $\beta = +\infty$, and one can also use the Bethe formula to write down a guess for the entropy of the uniform measure over solutions (the Boltzman entropy!).

To compute the Bethe free energy we replace the kernel function

$$f_a(\{x_i, i \in \partial a\}) = 1 - (1 - e^{-\beta}) \prod_{i \in a} (\frac{1 + s_i J_{ia}}{2}).$$

in (11.15)-(11.17) and use the parametrization (9.14) introduced in Chapter 9.3. The resulting expressions are easily found to be

$$\begin{aligned} F_{\text{Bethe}}(\underline{h}, \underline{\hat{h}}) = & \sum_i F_i(\{h_{j \to a}, j \in \partial a\}) + \sum_a F_a(\{\hat{h}_{b \to i}, i \in \partial b\}) \\ & - \sum_{ia} F_{ia}(h_{i \to a}, \hat{h}_{a \to i}) \end{aligned} \tag{11.24}$$

with

$$F_i = \ln\left\{ \prod_{a\in\partial i} (1 + J_{ia} \tanh \hat{h}_{a\to i}) + \prod_{a\in\partial i} (1 - J_{ia} \tanh \hat{h}_{a\to i}) \right\} \tag{11.25}$$

$$F_a = \ln\left\{ 1 - (1 - e^{-\beta}) \prod_{i\in\partial a} \frac{1 - \tanh h_{i\to a}}{2} \right\} \tag{11.26}$$

$$F_{ai} = \ln\left\{ 1 + \tan h_{i\to a} \tanh \hat{h}_{a\to i} \right\} \tag{11.27}$$

Again, the reader can easily check that the stationarity condition for $F_{\text{Bethe}}(\underline{h}, \underline{\hat{h}})$ is equivalent to BP equations presented in Chapter 9.3 (in Chapter 9.3 the equations (9.15)-(9.19) are written down for $\beta = +\infty$).

In the next chapter we discuss an important application of these formulas. When $-\beta F_{\text{Bethe}}[\underline{h}, \underline{\hat{h}}]/n$ is averaged over the graph ensemble one get a specific prediction for the entropy of the K-SAT ensemble. This prediction is not consistent with rigorous upper bounds on the SAT-UNSAT threshold. This means that the Bethe formulas and the corresponding BP equations are not good enough to inform us on the SAT-UNSAT transition. But this is not the end of the story. We will see that it is necessary to further develop the approach taken in this chapter and wander into the cavity method.

11.9 The Bethe free energy from the decoupling principle

We introduced the Bethe free energy for arbitrary factor graph models by first taking the expression of the exact free energy on trees. This is not a very satisfactory approach when the graph is complete such as is the case in the Sherrington-Kirkpatrick or compressive sensing models.

to do

12 Replica Symmetric Free Energy Functionals

The main idea behind density or state evolution analysis of message passing algorithms is to track their average behaviour. This allows to analyze their performance and derive their algorithmic (or dynamic) phase transition thresholds. But we also saw (for coding) in Chapter 10 that one can guess the (static) phase transition threshold through a Maxwell equal area construction. We defined EXIT curves computable from density evolution, on which the Maxwell construction gives the MAP threshold. However we did not provide any clear general principle for deciding what are the correct variables for which the equal area construction works. For the CW model the guess is quite trivial for symmetry reasons. In coding, for the BEC it is less trivial and even less so for general binary symmetric channels.

We will see in this chapter that by carrying the variational approach of the previous chapter one step further we will be able to provide some clues to reformulate the Maxwell construction in a less ambiguous and useful way. In particular we will be able to provide certain guiding lines determining the variables and appropriate curves on which Maxwell's construction works, thereby determining the static phase transition threshold. We will revisit the Maxwell construction for coding within the new variational approach, and as a benefit we will also formulate it for compresssive sensing. For the satisfiability problem we show here that a "naive" variational approach does not work. The correct variational approach is based on an extension of our basic message passing procedures, namely the cavity method, whose discussion is postponed to Chapters 16 and **??**.

We have seen that the sum-product (or belief propagation) equations are the stationarity conditions for the Bethe free energy. We will see in the present chapter that, analogously, the density and state evolution equations are the stationarity conditions of an "averaged" form of the Bethe free energy, called the *replica symmetric free energy functional*. The terminology "replica symmetric" is used because such functionals first appeared in the framework of the so-called replica calculations in spin glass theory. We do not need any of these magic calculations at the present point, and a brief discussion of the meaning of the terminology is best defered to Chapter 16. Obviously, since the replica symmetric functionals are a variational formulation of the density and state evolution equations they contain the information about the dynamic thresholds. But more is true. In coding and compressive sensing these functionals give the exact average free

energy (or entropy) in the infinite system size limit. Therefore they allow allow to predict the algorithmic *as well as static phase transition thresholds*. Until recently this claim was rigorously proved only in somewhat special cases such as the BEC channel or was supported by bounds. Recent proof techniques such as the *interpolation method* (Chapter 13) and spatial coupling (Chapter 14) have allowed to provide relatively simple and intuitive proofs in the cases of coding and compressive sensing.

For K-SAT as pointed out above the variational approach is more complicated. Here we will show that the predictions of the replica symmetric free energy functional are in fact wrong. Instead of being a curse this makes the subject even more fascinating. The correct thresholds, Maxwell construction and free energies are given by pushing the notions of Bethe and replica functionals "one level up" (see Chapters 16, **??**. That these predictions are correct for K-SAT and other similar constraint satisfaction problems is still to a large extent an open problem.

We refrain from giving a completely general definition of the replica symmetric free energy functional because this immediately leads to cumbersome notations. Rather we treat our three paradigms separately in the next paragraphs. Each one has its own features and going through each of them allows to cover most essential cases.

12.1 Ising model on a random k-regular graph

12.2 Coding

We first discuss the general definition of the replica symmetric free energy functional for the regular Gallager (d_v, d_c) ensemble over a general binary memoryless *symmetric* channel, and then specialize to the case of the BEC where the functional simply becomes a function of a real variable. Recall the notation $\mathrm{c}(\cdot)$ for the distribution of half-loglikehod ratios $h(y) = \frac{1}{2}\ln p(y|1)/p(y|-1)$. Formally, $\mathrm{c}(h)dh = p(y \mid 1)dy$ where $p(y \mid 1)$ is the probability that the channel outputs symbol y when the input bit is $s = (-1)^x = 1$. We recall that a symmetric channel satisfies $\mathrm{c}(-h) = e^{-h}\mathrm{c}(h)$.

Replica symmetric functionals for BMS channels

The main idea is to "pretend" that in expression (11.22) of teh Bethe free energy the messages $h_{i\to a}$, are i.i.d random variables distributed according to a trial distribution $\mathrm{x}(\cdot)$, and that $\hat{h}_{a\to i}$ are dependent random variables defined through *one of the two* BP equations (6.10)

$$\hat{h}_{a\to i} = \operatorname{atanh}\Big\{ \prod_{j\in\partial a\setminus i} \tanh\ h_{j\to a}\Big\}$$

Then one averages (11.22) which yields a functional of $\mathrm{x}(\cdot)$.

Here is the precise definition. Take a fixed "trial" probability distribution $\mathrm{x}(\cdot)$ over $\mathbb{R}$. Pick d_c i.i.d copies of the random variable $H \sim \mathrm{x}(\cdot)$, and call them H_k, $k = 1, \dots d_c$. Define the random variable

$$\hat{H} = \mathrm{atanh}\Big\{ \prod_{k=1}^{d_c} \tanh\ H_j \Big\} \tag{12.1}$$

Pick d_v i.i.d copies $\hat{H}_\ell$, $\ell = 1, \dots d_v$. Set $\underline{H} = (H_1, \dots, H_{d_c})$ and

$$\begin{aligned} f(h, \underline{H}, \underline{\hat{H}}) = & \ln\Big\{ e^h \prod_{\ell=1}^{d_v} (1 + \tanh \hat{H}_\ell) + e^{-h} \prod_{\ell=1}^{d_v} (1 - \tanh \hat{H}_\ell) \Big\} \\ & + \frac{d_v}{d_c} \ln \frac{1}{2} \Big\{ 1 + \prod_{k=1}^{d_c} \tanh H_k \Big\} - d_v \ln \Big\{ 1 + \tanh H_1 \tanh \hat{H}_1 \Big\} \end{aligned} \tag{12.2}$$

The replica symmetric (RS) free energy functional is defined as:

$$f_{\mathrm{RS}}[\mathrm{x}(.)] = \mathbb{E}[f(h, \underline{H}, \underline{\hat{H}}] \tag{12.3}$$

where the expectation is with respect to $h \sim \mathrm{c}\dot{()}$ and $\underline{H} \sim \mathrm{x}(\cdot)$. For an irregular LDPC ensemble the degrees (d_v, d_c) are random and one just has an extra average over their distribution.

We also define the replica symmetric entropy functional as

$$h_{\mathrm{RS}}[\mathrm{x}(.)] = -f_{\mathrm{RS}}[\mathrm{x}(.)] + \mathbb{E}[h] \tag{12.4}$$

The motivation for introducing this second functional will become clear in the next paragraph (see equ. (12.6)).

Replica symmetric formula and MAP threshold

Recall that the (true) average free energy is given by the thermodynamic limit $-\lim_{n\to+\infty} \mathbb{E}[\ln Z]/n$ where Z is the partition function f(3.11) for coding (recall that $\beta = 1$ in this context). The *replica symmetric formula* states that

$$-\lim_{n\to+\infty} \frac{1}{n} \mathbb{E}[\ln Z] = \inf_{\mathrm{x}\in\mathcal{S}} f_{\mathrm{RS}}[\mathrm{x}(.)] \tag{12.5}$$

where the infimum is taken over the space $\mathcal{S}$ of symmetric channel distributions, i.e. those satisfying $\mathrm{x}(-h) = e^{-h}\mathrm{x}(h)$. We stress that this formula is valid only for *symmetric* channels, a fact that reflects itself in the space where the infimum is taken.

Such formulas relating a free energy to a replica functional have been long standing conjectures since the mid 70's in the theory spin glass models on sparse and complete graph models. Much progress has been made in the last fifteen years towards their proofs. We will go through most of the proof of (12.5) in Chapters

13 and 14 when we introduced powerful techniques known as the "interpolation method" and "spatial coupling".

For us at present the most important consequence of the replica symmetric formula is the determination of the MAP threshold. In Chapter 10 we defined the MAP threshold as the smallest ϵ such that $\liminf_{n\to\infty} \mathbb{E}[H(\underline{X} \mid \underline{Y})]/n > 0$ (see definition 10.4). Recall also the general relationship (**??**)

$$\frac{1}{n}\mathbb{E}[H(\underline{X} \mid \underline{Y}(\epsilon))] = -\frac{1}{n}\mathbb{E}[\ln Z] + \mathbb{E}[h]. \tag{12.6}$$

Equation (12.5) has two consequences. One can replace lim inf by lim in the definition of the MAP threshold, but more importantly we get explicit formulas for the conditional entropy

$$\lim_{n\to+\infty} \frac{1}{n}\mathbb{E}[H(\underline{X} \mid \underline{Y})] = \sup_{\mathrm{x}\in\mathcal{S}} h_{\mathrm{RS}}[\mathrm{x}(.)] \tag{12.7}$$

and the MAP threshold

$$\epsilon_{\mathrm{MAP}} = \inf\{\epsilon \in [0,1] : \sup_{\mathrm{x}\in\mathcal{S}} h_{\mathrm{RS}}[\mathrm{x}(.)] > 0\} \tag{12.8}$$

In order to concretely calculate the conditional entropy and the MAP threshold one still has to solve the variational problem consisting in minimizing (or maximizing) the replica symmetric free energy (or entropy). It is possible to write down the stationary point condition which yields nothing else than the density evolution fixed point equation (see Equ. (6.26)-(6.27))

$$\mathrm{x} = c \otimes (\mathrm{x}^{\boxplus(d_c-1)})^{\otimes(d_v-1)}. \tag{12.9}$$

We will give in the next section a proof of this stationarity condition using the entropy functional formalism already used in Section 6.8. At any rate, this result should not come as big surprise since the stationary points of the Bethe free energy are given by the belief propagation equations and the density evolution equations are an averaged form of the belief propagation equations. Once stationary points, i.e. fixed points of (12.9), have been found one selects the one that yields the largest $h_{\mathrm{RS}}[\mathrm{x}(\cdot)]$ (or smallest $f_{\mathrm{RS}}[\mathrm{x}(\cdot)]$) and determines ϵ_{MAP}. Since in practice fixed points are found by iterative methods, it is fortunate that we only need to find *stable* fixed points. Indeed the maximum of $h_{\mathrm{RS}}[\mathrm{x}(\cdot)]$ (or minimum of $f_{\mathrm{RS}}[\mathrm{x}(\cdot)]$) is necessarily a stable fixed point.

But that is not all. We already know that (12.9) allows to determine the belief propagation threshold. Essentially, the BP threshold is the smallest noise for which a non-trivial fixed point is reached under iterations initialized with $\mathrm{x}_0(\cdot) = \mathrm{c}(\cdot)$ (recall that $\mathrm{x} = \Delta_0$ is always a trivial fixed point of (12.9)). Therefore this information is also contained in the replica symmetric free energy functional. The belief propagation threshold is the smallest noise such that the replica symetric free energy functional has a non trivial stationary point.

To summarize, the replica symmetric free energy functional contains all the information we want. In particular it allows to deduce the density evolution

equation. To determine the belief propagation threshold it suffices to solve the density evolution equation. To evaluate the MAP threshold we have to solve the density evolution equation *and* to evaluate the corresponding largest replica symmetric entropy or smallest replica symmetric free energy. In the next paragraph we specialize this discussion to the case of the BEC.

Explicit Case of the BEC

A bit transmitted through the BEC is either perfectly transmitted with probability ϵ or erased with probability $1-\epsilon$. This implies that $\mathrm{c}(\cdot) = \epsilon\Delta_0(\cdot)+(1-\epsilon)\Delta_\infty(\cdot)$. Thus the solutions of the stationarity condition are of the form

$$\mathrm{x}(\cdot) = x\Delta_0(\cdot) + (1-x)\ \Delta_\infty(\cdot) \tag{12.10}$$

where $x \in [0,1]$ satisfies the density evolution equation of the BEC $x = \epsilon(1-(1-x)^{d_c-1})^{d_v-1}$. The real variable x that parametrizes $\mathrm{x}(\cdot)$ has the interpretation of the probability of an erasure message emanating from variable nodes. These remarks allow to restrict the supremum in (12.7) to distributions of the form (12.10).

We now compute the expectation of (12.2). First note that $\hat{H}$ in (12.1) is distributed as

$$\hat{\mathrm{x}}(\cdot) = \hat{x}\Delta_0(\cdot) + (1-\hat{x})\ \Delta_\infty(\cdot), \qquad \hat{x} = 1-(1-x)^{d_c-1}. \tag{12.11}$$

One easily finds the "check node contribution" (expectation of second log in (12.2))

$$\mathbb{E}[\ln\frac{1}{2}(1+\prod_{k=1}^{d_c}\tan H_k)] = (1-x)^r\ln 2 - \ln 2 \tag{12.12}$$

and "edge contribution" (expectation of third log in (12.2))

$$\mathbb{E}[\ln(1+\tan H_1\tan\hat{H}_1)] = (1-x)(1-\hat{x})\ln 2$$

Let us now discuss the "variable node contribution" corresponding to the first term in (12.2). For the BEC, one should include the term $\mathbb{E}[h]$ in (12.4) directly

in this contribution in order to avoid working with infinite quantities. One finds

$$\begin{aligned}
&\mathbb{E}[\ln(\prod_{\ell=1}^{d_v}(1+\tanh\hat{H}_k)+e^{-2h}\prod_{\ell=1}^{d_v}(1-\tanh\hat{H}_k))] \\
&= (1-\epsilon)\sum_{e=0}^{d_v}\binom{d_v}{e}\hat{x}^e(1-\hat{x})^{d_v-e}\ln 2^{d_v-e}+\epsilon\sum_{e=0}^{d_v-1}\binom{d_v}{e}\hat{x}^e(1-\hat{x})^{d_v-e}\ln 2^{d_v-e} \\
&\quad +\epsilon\binom{d_v}{d_v}\hat{x}^{d_v}(1-\hat{x})^{d_v-d_v}\ln 2 \\
&= \sum_{e=0}^{d_v}\binom{d_v}{e}\hat{x}^e(1-\hat{x})^{d_v-e}(d_v-e)\ln 2+\epsilon\hat{x}^{d_v}\ln 2 \\
&= (1-\hat{x})\sum_{e=0}^{d_v}\hat{x}^e\frac{d}{dy}y^{d_v-e}\mid_{y=1-\hat{x}}\ln 2+\epsilon\hat{x}^{d_v}\ln 2 \\
&= (1-\hat{x})\frac{d}{dy}(\hat{x}+y)^{d_v}\mid_{y=1-\hat{x}}\ln 2+\epsilon\hat{x}^{d_v}\ln 2 \\
&= d_v(1-\hat{x})\ln 2+\epsilon\hat{x}^{d_v}\ln 2
\end{aligned}$$

Putting all three contributions together and using (**??**) one finds the replica symmetric entropy for the BEC as a function of the erasure probability x,

$$\frac{h_{\rm RS}(x;\epsilon)}{\ln 2}=(\frac{d_v}{d_c}-l)(1-x)^{d_c}+d_v(1-x)^{d_c-1}+\epsilon(1-(1-x)^{d_c-1})^{d_v}-\frac{d_v}{d_c} \tag{12.13}$$

According to (12.5) and (12.8) the conditional entropy is given by

$$\lim_{n\to+\infty}\frac{1}{n}\mathbb{E}[H(\underline{X}\mid\underline{Y}(\epsilon))]=\max_{0\le x\le 1}h_{\rm RS}(x;\epsilon) \tag{12.14}$$

and the MAP threshold can be calculated from

$$\epsilon_{\rm MAP}=\inf\{\epsilon:\max_{0\le x\le 1}h_{\rm RS}(x;\epsilon)>0\}. \tag{12.15}$$

It is immediate to check that the stationnary points are given by the usual density evolution fixed point equation $x=\epsilon(1-(1-x)^{d_c-1})^{d_v-1}$.

As pointed out before, the replica symmetric formula contains all the information about the belief propagation and MAP thresholds, so it is very useful to have an explicit idea of the shape of this function and its evolution as a function of the channel noise. Figure **??** shows $-h_{\rm RS}$ as a function of x, for various values of ϵ (we prefer to plot minus the entropy which here is the "free energy" up to an irrelevant additive constant). To avoid any confusion let us stress that there is no reason why $h_{\rm RS}(x)$ should be non-negative. It is only $\max_{0\le x\le 1}h_{\rm RS}(x)$ that really is an entropy and therefore *has to be non-negative*. For all ϵ there is a trivial minimum at $x=0$, which is also the trivial stable fixed point of density evolution. For $\epsilon<\epsilon_{\rm BP}$ this minimum is unique, and hence global. At $\epsilon=\epsilon_{\rm BP}$ the function develops a flat inflexion point, and a second local minimum as well as

— plot of potential goes here —

Figure 12.1 This plot is generic only for regular ensembles with $l \geq 3$. Irregular ensembles can have a richer behavior and the corresponding discussion is more complicated. The case $l = 2$ is somewhat special because $\epsilon_{\text{BP}} = \epsilon_{\text{MAP}}$.

a local maximum branch of. The local minimum is the stable non-trivial fixed point of density evolution, $x_{\text{st}}(\epsilon)$, and the local maximum is the unstable fixed point $x_{\text{un}}(\epsilon)$. As one increases ϵ further the local minimum at $x_{\text{st}}(\epsilon)$ decreases until it touches the horizontal axis for ϵ_{MAP}. At this threshold value there are two global minima at $x = 0$ and $x = x_{\text{st}}(\epsilon_{\text{MAP}})$, and

$$h_{\text{RS}}(0; \epsilon_{\text{MAP}}) = h_{\text{RS}}(x_{\text{st}}(\epsilon_{\text{MAP}}); \epsilon_{\text{MAP}}) \tag{12.16}$$

Finally, for $\epsilon > \epsilon_{\text{MAP}}$ it is $x_{\text{st}}(\epsilon)$ that becomes the unique global minimum and we have decoding errors since the entropy is strictly positive. To summarize, one should retain from this discussion that the replica symmetric function $h_{\text{RS}}(x, \epsilon)$ contains all the information we want. The belief propagation threshold is found by searching values of ϵ where the function develops flat inflexion points, and the MAP threshold is found by looking at values of ϵ where the two minima are at the same height.

It is interesting to cast the replica symetric formula (12.14) in an equivalent form where the notion of EXIT functions, defined somewhat arbitrarily in Chapter 10, appear in a natural and automatic way. Consider $\epsilon > \epsilon_{\text{MAP}}$ and the stable non-trivial fixed point $x_{\text{st}}(\epsilon)$. This is also the maximum of $h_{\text{RS}}(x; \epsilon)$ so

$$\begin{aligned}
\frac{d}{d\epsilon} \max_{0 \leq x \leq 1} h_{\text{RS}}(x_{\text{st}}; \epsilon) &= \frac{\partial}{\partial \epsilon} h_{\text{RS}}(x_{\text{st}}; \epsilon) + \frac{\partial}{\partial x} h_{\text{RS}}(x_{\text{st}}; \epsilon) \frac{dx_{\text{st}}}{d\epsilon} \\
&= \frac{\partial}{\partial \epsilon} h_{\text{RS}}(x_{\text{st}}; \epsilon) \\
&= (1 - (1 - x_{\text{st}}(\epsilon))^{d_c - 1})^{d_v}
\end{aligned}$$

The second equality is valid because $x_{\text{st}}(\epsilon)$ is a stationnary point of $h_{\text{RS}}(x; \epsilon)$ and $\frac{dx_{\text{st}}}{d\epsilon}$ is finite for $\epsilon > \epsilon_{\text{MAP}}$ (this last point can be checked rather explicitly for the BEC but for other channels this is much more difficult). The third equality

immediately follows from the explicit formula (12.13). We conclude

$$\frac{d}{d\epsilon}\lim_{n\to+\infty}\frac{1}{n}\mathbb{E}[H(\underline{X}\mid\underline{Y}(\epsilon))]=\begin{cases}0,\ \epsilon<\epsilon_{\mathrm{MAP}}\\(1-(1-x_{\mathrm{st}}(\epsilon))^{d_c-1})^{d_v},\ \epsilon>\epsilon_{\mathrm{MAP}}\end{cases}\tag{12.17}$$

For $\epsilon>\epsilon_{\mathrm{MAP}}$ the derivative of the conditional entropy coincides with the EXIT function $h_{\mathrm{BP}}(\epsilon)$ of Section 10.3 (for $\epsilon<\epsilon_{\mathrm{BP}}$ the two functions vanish so they also coincide; but we stress that for $\epsilon_{\mathrm{BP}}<\epsilon<\epsilon_{\mathrm{MAP}}$ these two functions *do not* coincide). Moreover the inequality in Lemma **??** which relates the MAP and BP EXIT functions can be replaced by an equality.

The reader should go back to the exact solution of the Curie-Weiss model in Chapter 4 and notice the intimate structural analogies with the present situation. The Curie-Weiss free energy is given by a variational problem $\min_{-1\le m\le 1} f(m)$ whose solutions determine both the phase transition ("MAP") threshold $h=0$ and the spinodal ("BP") points $\pm h_{\mathrm{sp}}$. Finally the derivative of the free energy yields the magnetization (a discontinuous curve at $h=0$ for $\beta J>1$). This is the analog of (12.17).

Back to the Maxwell Construction

The replica symmetric functional (12.3)-(12.4) allows to formulate the Maxwell construction in a conceptually clear and unambiguous way. In particular once this functional is known, it is more or less automatic to choose the right variables and "Van der Waals" curves for which the equal area condition works and yields the (static) phase transition threshold. We discuss the explicit case of the BEC in this paragraph. The generalization to other binary memoryless channels is more technical and requires the use of an appropriate formalism here called the "entropy functional formalism" which the interested reader can find in the next section.

For the sake of the argument, we view the replica symmetric function $h_{\mathrm{RS}}(x;\epsilon)$ as a function from the plane $(\epsilon;x)\in\mathbb{R}^2$ to $\mathbb{R}$. Consider the curve $\mathcal{F}$ given in parametric form by $(\epsilon(x);x)$ where $\epsilon(x)=x/(1-(1-x)^{d_c-1})^{d_v-1})$. Note that on $\mathcal{F}$ the density evolution equation is satisfied so the replica symmetric entropy is stationnary, $\partial h_{\mathrm{RS}}(x;\epsilon(x))/\partial x=0$. Note also that the union of the horizontal axis and $\mathcal{F}$ is the graph all density evolution fixed points. This graph shown on figure 12.2 has three branches: the horizontal axis corresponding to the trivial fixed point $x=0$ for all ϵ; a branch $(\epsilon,x_{\mathrm{st}}(\epsilon))$, $\epsilon>\epsilon_{\mathrm{BP}}$ corresponding to the non-trivial stable fixed point; and a branch $(\epsilon,x_{\mathrm{un}}(\epsilon))$, $\epsilon>\epsilon_{\mathrm{BP}}$ corresponding to the unstable fixed point.

Now consider the path $\mathcal{M}$ starting from $(\epsilon_{\mathrm{MAP}},0)$ and going to $(+\infty,0)$ on the horizontal axis, and then backwards along the curve $\mathcal{F}$ up to some x. Look at

— figure —

Figure 12.2 *Left*: the set of fixed points of density evolution. The horizontal axis corresponds to the trivial fixed point, the upper branch to the stable fixed point and the lower branch to the unstable fixed point. *Right*: EXIT curve Equ. (12.18)

the total change in RS entropy along this path. We have

$$\begin{aligned} h_{\mathrm{RS}}(x;\epsilon(x)) - h_{\mathrm{RS}}(0;\epsilon_{\mathrm{MAP}}) &= \int_{\mathcal{M}} dh_{\mathrm{RS}} = 0 + \int_0^x dx \frac{d}{dx} h_{\mathrm{RS}}(x;\epsilon(x)) \\ &= \int_0^x dx (\frac{\partial}{\partial x} h_{\mathrm{RS}}(x;\epsilon(x)) + \epsilon'(x) \frac{\partial}{\partial \epsilon} h_{\mathrm{RS}}(x;\epsilon(x))) \\ &= \int_0^x dx \epsilon'(x) \frac{\partial}{\partial \epsilon} h_{\mathrm{RS}}(x;\epsilon(x)) \\ &= \int_0^x dx \epsilon'(x) (1-(1-x)^{r-1})^l \end{aligned}$$

The last integral is recognized as the trial entropy $P(x)$, the area under the EXIT curve $(\epsilon(x), (1-(1-x)^{r-1})^l)$ introduced in Definition 10.1.

Let us highlight the main points of this discussion. The replica symmetric function gives a more fundamental definition of the BP EXIT curve in parametric form (see figure 12.2),

$$(\epsilon(x), \frac{\partial}{\partial \epsilon} h_{\mathrm{RS}}(x;\epsilon)). \tag{12.18}$$

The trial entropy is also expressed in terms of the replica symmetric function as

$$h_{\mathrm{RS}}(x;\epsilon(x)) - h_{\mathrm{RS}}(0;\epsilon_{\mathrm{MAP}}) = \int_0^x dx \epsilon'(x) \frac{\partial}{\partial \epsilon} h_{\mathrm{RS}}(x;\epsilon(x)). \tag{12.19}$$

Finally, setting $x = x_{\mathrm{st}}(\epsilon_{\mathrm{MAP}})$ in (12.19) shows that the equal area condition and (12.16) are equivalent (namely that both sides vanish).

12.3 Entropy functional formalism

In the analysis of density evolution for general binary memoryless channels we introduced the *entropy functional*

$$H[\mathrm{x}] = \int dh\, \mathrm{x}(h) \ln(1 + e^{-2h}). \tag{12.20}$$

Recall the interpretation: this is the (single letter) Shannon entropy $H(Y \mid X)$ of a binary symmetric channel whose log-likelihood distribution is $\mathrm{x}(h)$. It turns out that the replica symmetric functional can be expressed entirely in terms of the entropy functional. This gives a very convenient formalism to deal with general channels.

We first note two identities. Let $\mathrm{x}_1, \cdots, \mathrm{x}_k$ be symmetric distributions of k random variables $H_1, \cdots, H_k$. The distribution of $\sum_{i=1}^k H_i$ is given by the usual convolution operation (6.24) at variable nodes, and thus

$$H(\otimes_{i=1}^k \mathrm{x}_i) = \int \prod_{i=1}^k dh_i \mathrm{x}_i(h_i)\ \ln(1 + e^{-2\sum_{i=1}^k h_i}). \tag{12.21}$$

On the other hand the distribution of $\prod_{i=1}^k \tanh H_i$ is given by the check node convolution operation (6.25),

$$H(\boxplus_{i=1}^k \mathrm{x}_i) = -\int \prod_{i=1}^k dh_i \mathrm{x}_i(h_i)\ \ln(1 + \prod_{i=1}^k \tanh h_i). \tag{12.22}$$

Replica symmetric functionnal in terms of the entropy functional

First consider the second term in (12.2). All log-likelihood variables in this term have trial distribution x. Thus identity (12.22) implies

$$\frac{d_v}{d_c}\mathbb{E}\big[\ln \frac{1}{2}\big\{1 + \prod_{k=1}^{d_c} \tanh H_k\big\}\big] = -\frac{d_v}{d_c} H(\mathrm{x}^{\boxplus d_c}). \tag{12.23}$$

Similarly for the third term in (12.2) using also (12.1) we have

$$\begin{aligned} d_v\mathbb{E}\big[\ln\big\{1 + \tanh H_1 \tanh \hat{H}_1\big\}\big] &= d_v\mathbb{E}\big[\ln\big\{1 + \tanh H_1 \prod_{k=1}^{d_c-1} \tanh H_k\big\}\big] \\ &= d_v H(\mathrm{x} \boxplus \mathrm{x}^{\boxplus(d_c-1)}) \end{aligned} \tag{12.24}$$

Figure 12.3 Potential functional for the LDPC($d_v = 3, d_c = 6$) ensemble over a binary symmetric channel (BSC), with entropy h. The values of h for these curves are, from the top to bottom, $0.40, 0.416, 0.44, 0.469, 0.48$. The other input $\mathrm{x}(\cdot)$ to the potential functional is the LLR distribution for the binary AWGN channel (BAWGNC) with entropy $\tilde{\mathrm{h}}$. The choice of BAWGNC distribution for x is arbitrary.

To express the first term in (12.2) we first transform it into

$$\begin{aligned}
&\ln\Big\{e^{h}\prod_{\ell=1}^{d_v}(1+\tanh\hat{H}_\ell)+e^{-h}\prod_{\ell=1}^{d_v}(1-\tanh\hat{H}_\ell)\Big\}\\
&\qquad = h+\ln\Big\{\prod_{\ell=1}^{d_v}(1+\tanh\hat{H}_\ell)\Big\}+\ln\Big\{1+e^{-2h}\prod_{\ell=1}^{d_v}\frac{1-\tanh\hat{H}_\ell}{1+\tanh\hat{H}_\ell}\Big\}\\
&\qquad = h+\sum_{\ell=1}^{d_v}\ln(1+\tanh\hat{H}_\ell)+\ln\Big\{1+e^{-2(h+\sum_{\ell=1}^{d_v}\hat{H}_\ell}\Big\}
\end{aligned}$$

Then, using (12.1) together with both identities (12.22) and (12.22) we get

$$\begin{aligned}
&\mathbb{E}\big[\ln\big\{e^{h}\prod_{\ell=1}^{d_v}(1+\tanh\hat{H}_\ell)+e^{-h}\prod_{\ell=1}^{d_v}(1-\tanh\hat{H}_\ell)\big\}\big]\\
&\qquad = \mathbb{E}[h]-d_v H(\mathrm{x}^{\boxplus(d_c-1)})+H(c\otimes(\mathrm{x}^{\boxplus(d_c-1)})^{\otimes d_v}) \qquad (12.25)
\end{aligned}$$

Finally, collecting (12.23), (12.24), (12.25) and replacing in the definition of the replica symmetric entropy (12.4) we obtain the final expression,

$$\begin{aligned}
h_{\mathrm{RS}}[\mathrm{x}(\cdot)] = {} & d_v H(\mathrm{x}^{\boxplus(d_c-1)})-H(c\otimes(\mathrm{x}^{\boxplus(d_c-1)})^{\otimes d_v})\\
& +\frac{d_v}{d_c}H(\mathrm{x}^{\boxplus d_c})-d_v H(\mathrm{x}\boxplus\mathrm{x}^{\boxplus(d_c-1)}) \qquad (12.26)
\end{aligned}$$

The first two terms come the contributions of the variable nodes, while the third and fourth one come from teh contributions of check nodes and edges. Of course the fourth term is equal to $-d_v H(\mathrm{x}^{\boxplus(d_c)})$. However written as in (12.26) the formula generalizes immediately to general irregular LDPC ensembles (see exercises). Figure 12.3 shows the typical cross section of this functional for $\mathrm{x}(\cdot)$ given by a family of BAWGNC parametrized by their entropy. The channel is a BSC and the LDPC ensemble a regular Gallager ($d_v = 3, d_c = 6$) code.

We leave it as an exercise for the reader to check that the expression (12.13) for the BEC is recovered by replacing $c = \epsilon)\Delta_0 + \epsilon\Delta_\infty$ and $\mathrm{x} = x\Delta_0 + (1-x)\Delta_\infty$.

Maxwell's construction for general BMS channels

to do

12.4 Compressive Sensing

Write RS free energy (can be derived by integrating out state evolution). Illustrate thresholds it predicts. Discuss that RS is exact. Do it for Lasso or for known prior case ?

12.5 Replica symmetric free energy and entropy for K-SAT

In Chapter 11 we gave the expression of the Bethe free energy for K-SAT at finite temperature. As noted there, from this expression, as long as there exist at least one solution, taking the limit $\beta \to +\infty$ one also gets a Bethe formula for the entropy of the uniform measure over solutions. There are natural replica symmetric functionals associated to the Bethe formulas which we discuss here. But contrary to coding and compressive sensing it will become quite clear from the entropy formula the replica symmetric expressions cannot be exact for K-sat. This is not a curiosity of the model. In fact most constraint satisfaction models suffer from this problem, and a better theory is needed. This theory goes under the name *cavity theory* and is developped in Chapters 16 and **??**. In the universe of factor graph models it is coding and compressive sensing that are special, and the K-sat model is believed to display quite a generic behavior.

The construction of the replica symmetric functional like in the coding case. One takes as a starting point the Bethe expression (11.24) and treats all messages $h_{i\to a}$ as independent random variables distributed according to a trial distribution, while the messages $\hat{h}_{a\to i}$ have an induced distribution found from the message passing equation (9.19) (generalized to non-zero temperature),

$$\hat{h}_{a\to i} = -\frac{1}{2}\ln\Big\{1 - (1-e^{-\beta})\prod_{j\in\partial a\setminus i}\frac{1}{2}(1-\tanh h_{j\to a})\Big\} \tag{12.27}$$

In the coding case we limited ourselves to regular Gallager (d_v, d_c) ensembles. One difference here is that while the check nodes have regular degree K, the variable node degrees are (asymtotically) Poisson distributed with average degree αK.

Let us define the replica symmetric functional. Fix a trial distribution $\mathrm{x}(\cdot)$ on

$\mathbb{R}$. Pick K iid copies of the random variable $H \sim \mathrm{x}(\cdot)$. Call them $H_1, \dots, H_K$. Define the random variable

$$\hat{H} = -\frac{1}{2}\ln\Big\{1 - (1-e^{-\beta})\prod_{k=1}^{K-1}\frac{1}{2}(1-\tanh H_k)\Big\}. \tag{12.28}$$

Pick two Poisson distributed integers p and q with average $\alpha K/2$. Pick p copies $\hat{H}_1^+, \dots, \hat{H}_p^+$ of $\hat{H}$ and q copies $\hat{H}_1^-, \dots, \hat{H}_p^-$ of $\hat{H}$. Let

$$\begin{aligned} f(\underline{H}, \underline{\hat{H}}, p, q) = &\ln\Big\{\prod_{\ell=1}^{p}(1-\tanh\hat{H}_\ell^+)\prod_{\ell=1}^{q}(1+\tanh\hat{H}_\ell^-) \\ &\quad + \prod_{\ell=1}^{p}(1+\tanh\hat{H}_\ell^+)\prod_{\ell=1}^{q}(1-\tanh\hat{H}_\ell^-)\Big\} \\ &+ \ln\Big\{1-(1-e^{-\beta})\prod_{k=1}^{K}\frac{1}{2}(1-\tanh H_k)\Big\} \\ &- \ln\Big\{1+\tanh H_1 \tanh\hat{H}_1\Big\} \end{aligned}$$

The RS free energy functional is defined as

$$f_{\mathrm{RS}}(\mathrm{x}(\cdot)) = \mathbb{E}[f(\underline{H}, \underline{\hat{H}}, p, q)] \tag{12.29}$$

where the expectation is over all random variables $p, q, \underline{H}, \underline{\hat{H}}$ (the probability distribution of $\underline{\hat{H}}$ is itself dependent on x).

The replica symmetric prescription for computing the free energy is to take

$$f_{\mathrm{RS}}(\beta, \alpha) \equiv \sup_{\mathrm{x}(\cdot)} f_{\mathrm{RS}}(\mathrm{x}(\cdot)). \tag{12.30}$$

The replica symmetric entropy can be obtained by $s_{\mathrm{RS}}(\beta,\alpha) = \partial f_{\mathrm{RS}}/\partial\beta^{-1}$ (recall (2.18)). Let us point out that to compute the zero temperature replica symmetric entropy, i.e. a prediction for the log number of solutions, instead of differentiating (12.30) with respect to β^{-1} it is simpler to simply set $\beta = +\infty$ in all equations above.

The stationnary points of (12.30) yield a fixed point integral equation for x(.). This can be split in two integral equations linking $\mathrm{x}(\cdot)$ and $\hat{\mathrm{x}}(\cdot)$, where $\hat{\mathrm{x}}(\cdot)$ is the induced distribution of $\hat{H}$ (see (12.28))

$$\begin{cases} \mathrm{x}(H) = \sum_{p,q=0}^{+\infty}\frac{(\alpha K/2)^{p+q}}{p!q!}e^{-\alpha K}\int\big\{\prod_{\ell=1}^{p}d\hat{H}_\ell^+\hat{\mathrm{x}}(\hat{H}_\ell^+)\big\}\big\{\prod_{\ell=1}^{q}d\hat{H}_\ell^-\hat{\mathrm{x}}(\hat{H}_\ell^-)\big\} \\ \qquad\qquad \times\delta(H - (\sum_{\ell=1}^{p}\hat{H}_\ell^+ - \sum_{\ell=1}^{q}\hat{H}_\ell^-)), \\ \hat{\mathrm{x}}(\hat{H}) = \int\big\{\prod_{k=1}^{K-1}d\hat{H}_k\mathrm{x}(H_k)\big\}\delta(\hat{H} + \frac{1}{2}\ln\big\{1-(1-e^{-\beta}) \\ \qquad\qquad \times\prod_{k=1}^{K-1}\frac{1}{2}(1-\tanh H_k)\big\}). \end{cases}$$

These equations can be solved numerically, e.g. by the population dynamics method (see homework **??**). This allows to find the maximizer of the replica symmetric functional and hence obtain a prediction for the free energy and entropy.

The computation of the entropy shows that the replica symmetric prediction cannot hold in all regions of the (α, β) plane. Indeed at sufficiently low temperatures and large constraint densities the replica symmetric entropy becomes negative, whereas the true entropy must always remain non-negative. For example at zero temperature the enetropy is just the log number of solutions (per variable) and cannot be negative. Figure 12.4 shows $s_{\rm RS}(\beta = +\infty, \alpha)$ for $K = 3$. This function decreases as the clause density increases, and vanishes at $\alpha_{\rm RS} \approx 4.677$. Thus the present replica symmetric analysis certainly breaks down for zero temperature and $\alpha_{\rm RS}$. Could it be that it is correct (at zero temperature) for $\alpha < \alpha_{\rm RS}$? In other words could it be that the replica symmetric zero temperature entropy correctly predicts the satisfiability threshold? *The answer is no.* In problem **??** the reader is guided through the proof of an upper bound $\alpha_s \leq 4.666$ for $K = 3$ which is slightly lower than $\alpha_{\rm RS}$ (there are also sharper upper bounds). According to the cavity method of Chapter 16 the replica symmetric formula is exact in a region $\alpha < \alpha_c(\beta)$ with $\alpha_c(\beta)$ a threshold called the *condensation threshold*. At $\alpha_c(\beta)$ there is a phase transition (the free energy and entropy are not analytic) and for $\alpha > \alpha_c(\beta)$ one has to resort to more complicated formulas for the free energy and entropy (the "1RSB" formulas). At zero temperature one has $\alpha(\beta = +\infty) < \alpha_s$ for all K. So in the range $0 < \alpha <$ the zero-temperature replica symmetric entropy correctly counts the log number of solutions, but does not do so for $\alpha_c(\beta = +\infty) < \alpha < \alpha_s$. For $K = 3$ the cavity method predicts $\alpha_c(\beta = +\infty) \approx 3.86$ and $\alpha_s \approx 4.266$.

12.6 Notes

A few words about the concept of order paramter. Like for many physical concepts there is no rigid definition, and finding the correct order parameter is an art validated by experiment. Depending on the problem at hand this can seem more or less obvious like in fluids (the volume per particle) or in magnetism (the magnetization), but can be much more subtle like in superconductivity (the "wave function" of Cooper pairs). The Higgs field is the order parameter associated to the electroweak phase transition that occurred at an early epoch of the universe. The recently discovered Higgs bosons are elementary excitations of this field, much like spin flips are elementary excitations associated to magnetization. As we will see K-SAT is one of these problems for which the guess of the order parameter requires a stretch of imagination: probability distributions of random probability distributions.

Problems

12.1 Replica symmetric entropy for the BEC. Check that (12.13) directly follows from the expression (12.26) for general channels.

12.2 Replica symmetric entropy for general channels. Consider an LDPC ensemble with general variable node degree distribution $L(x) = \sum_k L_k x^k$ and check node degree distribution $R(x) = \sum_k R_k x^k$. The degree distributions from the edge perspective are $\lambda(x) = \sum_k \lambda_k x^k$ and $\rho(x) = \sum_k \rho_k x^k$. Generalize the replica symmetric expression (12.26) to irregular codes.

12.3 Replica symmetric expression for an LDGM code. Consider an LDGM code ensemble and give the replica symmetric expression for the entropy on a general channel and in particular on the BEC.

12.4 RS analysis for K-SAT. Derive the density evolution equations for K-SAT. Use population dynamics (as seen in homeworks of Chapter **??**) to compute the RS prediction for $\alpha_{\text{sat-unsat}}$.

12.5 Crude upper bound on the SAT-UNSAT threshold. Upper bounds for the SAT-UNSAT threshold, we call it α_s, are usually derived by counting arguments. This exercise develops the simplest such argument.

An assignment is a tuple $\underline{x} = (x_1, \ldots, x_n)$ where $x_i = 0, 1$ of n variables. The total number of possible clauses with k variables is equal to $2^k \binom{n}{k}$. A random formula F is constructed by picking, with replacement, uniformly at random, m clauses. Thus there are $(2^k \binom{n}{k})^m$ possible formulas.

We set $m = \alpha n$ and think of n and m as tending to ∞ with α fixed. This is the regime displaying a SAT-UNSAT threshold.

It is useful to keep in mind that $\mathbb{P}[A] = \mathbb{E}[1(A)]$ where $1(A)$ is the indicator function of event A. In what follows probabilities and expectations are with respect to the random formulas F.

Let $S(F)$ be the set of all assignments satisfying F and let $|S(F)|$ be its cardinality. Since F is a random formula, $|S(F)|$ is an integer valued random variable.

a) Show the Markov inequality $\mathbb{P}[F \text{ satisfiable}] \leq \mathbb{E}[|S(F)|]$.

b) Fix an assignement $\underline{x}$. Show that $\mathbb{P}[\underline{x} \,\text{satisfies}\, F] = (1 - 2^{-k})^m$. Then deduce that

$$\mathbb{E}[|S(F)|] = 2^n (1 - 2^{-k})^m.$$

c) Deduce the upper bound

$$\alpha_s < \frac{\ln 2}{|\ln(1 - 2^{-k})|}.$$

For $k = 3$ this yields $\alpha_s < 5.191$.

12.6 Bound by counting a restricted set of assignments. This is a follow-up of the previous exercise. You will study a more subtle counting argument which leads to an important improvement by Kirousis, Kranakis, Krizanc and

Stamatiou, *Approximating the Unsatisfiability Threshold of Random Formulas*, in Random Struct and Algorithms (1998). This type of method can be further refined and has led to better bounds.

We define the set $S_{\rm m}(F)$ of *maximal* satisfying assignments as follows. An assignment $\underline{x} \in S_{\rm m}(F)$ iff:

- $\underline{x}$ satisfies F,
- for all i such that $x_i = 0$ (in $\underline{x}$), the *single flip* $x_i \to 1$ yields an assignment - call it $\underline{x}^i$ - that *violates* F.

a) Show that if F is satisfiable then $S_m(F)$ is not empty. *Hint*: proceed by contradiction.

b) Show as in the first exercise the Markov inequality $\mathbb{P}[F \text{ satisfiable}] \leq \mathbb{E}[|S_m(F)|]$

c) Show that

$$\mathbb{E}[|S_m(F)|] = (1-2^{-k})^m \sum_{\underline{x}} \mathbb{P}[\cap_{i:x_i=0} (\underline{x}^i \text{ violates } F) \mid \underline{x} \text{ satisfies } F].$$

d) Fix $\underline{x}$. The events $E_i \equiv (\underline{x}^i \text{ violates } F)$ are negatively correlated, i.e

$$\mathbb{P}[\cap_{i:x_i=0} E_i \mid \underline{x} \text{ satisfies } F] \leq \prod_{i:x_i=0} \mathbb{P}[E_i \mid \underline{x} \text{ satisfies } F]$$

For the full proof which uses a correlation inequality (of FKG type) we refer to the reference given above. Here is a rough intuition for the inequality. First note that if $x_i = 0$ and $\underline{x}^i$ violates F, there must be some set S_i of clauses (in F) that are satisfied *only* by this variable $x_i = 0$ (this set might contain only one clause). This restricts the possible formulas contributing to the event E_i. Second note that sets S_i, S_j corresponding to different such variables $x_i = 0$, $x_j = 0$ must be *disjoint*. This "repulsion" between the sets S_i and S_j puts even more restrictions on the possible formulas, compared to a hypothetical situation where the events (and thus the sets S_i and S_j) would have been independent.

e) Now show that

$$\mathbb{P}[E_i \mid \underline{x} \text{ satisfies } F] = 1 - \left(1 - \frac{\binom{n-1}{k-1}}{(2^k-1)\binom{n}{k}}\right)^m.$$

Hint: note that in the event E_i there must be at least one clause containing $x_i = 0$ and containing other variables that do not satisfy it.

f) Deduce from the above results that $\lim_{n\to 0} \mathbb{P}[F \text{ satisfiable}] = 0$ as long as α satisfies

$$(1-2^{-k})^\alpha (2 - e^{-\frac{\alpha k}{2^k-1}}) < 1.$$

The improvement compared with the first exercise resides in the factor $e^{-\frac{\alpha k}{2^k-1}}$. A numerical evaluation for $k = 3$ yields the bound $\alpha_s < 4.667$.

— figure —

Figure 12.4 The replica symmetric prediction for the entropy (12.30) at $K = 3$. This curve predicts a SAT-UNSAT threshold at $\alpha \approx 4.677$ which is not consistent with known rigorous upper bounds.

13 Interpolation Methods

14 Spatial Coupling and Nucleation Phenomenon

So far we have seen that a variety of problems can be phrased in a natural way in terms of marginalizing a highly-factorized function. Message-passing algorithms are then the logical choice to accomplish this marginalization and we have seen how such algorithms perform in the thermodynamic limit.

Perhaps more surprisingly, we saw that the same quantities which were important for the analysis of the suboptimal message-passing algorithm reappeared when we looked at the seemingly more fundamental question of determining static thresholds, like the MAP threshold or the SAT/UNSAT threshold. The Maxwell construction is a graphical representation of this phenomenon.

We will now tie these two threads together. We will discuss a generic construction, called spatial coupling, which can be applied to a wide range of graphical models. The idea is to take many copies of a graphical model, to place them next to each other on a line and then to start connecting these models by "exchanging edges" in such a way that the local structure of the graphical model remains unchanged but that globally we create a larger graphical model which forms a one-dimensional chain. If in addition we impose suitable conditions at the boundaries of the model, this larger graphical model behaves very well under message-passing. Roughly speaking, the performance of the large spatially-coupled model under message-passing (in terms of the resulting threshold) is as good as if we had done optimal processing on the original graphical model.

For the most part we will only discuss the phenomenon but we will not give proofs. We will see how this phenomenon has again a nice physical interpretation. In fact – it is what is called the *nucleation* phenomenon in physics. Nucleation explains amongst other things how crystals grow, starting with a *seed* or *nucleus*.

We will discuss two important consequences of the nucleation phenomenon.

First, whenever we are in control of the graphical structure and the size of the graph is not very crucial, it is natural to construct the graph according to the above recipe. This results in graphs which are well suited for message-passing processing and give very good performance. E.g., for the coding problem this construction makes it possible to design codes which, under BP decoding, are not only provably capacity-achieving for a particular channel, but are in fact universally so, i.e., they are capacity-achieving for the whole class of BMS channels. A similar construction is possible for the compressive sensing problem.

There is a second, equally important application of the idea, namely to use

spatial coupling as a proof technique. Consider e.g. the case of the K-SAT problem. Also in this case we can use spatial coupling. This means we can construct spatially-coupled K-SAT formulas, and it is easier to find satisfiable solutions for such formulas than for the uncoupled ones. But what is the use of this? In coding, we were in charge of picking the code, and so we can pick coupled ones. The same thing applies for compressive sensing. We do not have the same degree of freedom for the constraint satisfaction problem where the formula is given to us. The idea is the following. If we are able to analyze the performance of a message-passing algorithm on coupled formulas then we can use the so-called *interpolation* method to show that this algorithmic threshold is also a lower bound on the SAT/UNSAT threshold of the uncoupled ensemble. So in this case we use spatial coupling only as a thought experiment. Indeed, the same method can be used in the context of coding to prove that the MAP threshold of the uncoupled formula is at least as large as the area threshold. Together with the upper bound on the MAP threshold which we derived in Chapter 10 this shows that the MAP threshold of the uncoupled ensemble is equal to the area threshold.

In the remainder of the chapter we go over our three running examples. In each case we describe the construction, the performance of the coupled system, as well as the consequences for our problem at hand.

14.1 Coding

There are many possible ways of constructing coupled graphical models from uncoupled ones. The "saturation phenomenon" is fairly robust with respect to the exact way of how we construct coupled models. So the difference lies mostly in how convenient the construction is either from a practical perspective or for the purpose of proofs. We present below two generic ways to achieve the spatial coupling. We start with the "protograph" construction. It has a very good performance and the additional structure is well suited for implementations. Our second construction is a "random" model. This model is well suited for proofs. Indeed, in the sequel we exclusively use the random model when it comes to showing plots and to formulating theorems.

Protograph Construction

To start, consider a protograph of a standard $(3, 6)$-regular ensemble (see (**?**, **?**) for the definition of protographs). It is shown in Figure 14.1. There are two variable nodes and there is one check node. Let M denote the number of variable nodes at each position. For our example, $M = 100$ means that we have 50 copies of the protograph so that we have 100 variable nodes at each position. For all future discussions we will consider the regime where M tends to infinity.

Next, consider a collection of $(2L+1)$ such protographs as shown in Figure 14.2. These protographs are non-interacting and so each component behaves just like

Figure 14.1 Protograph of a standard $(3, 6)$-regular ensemble.

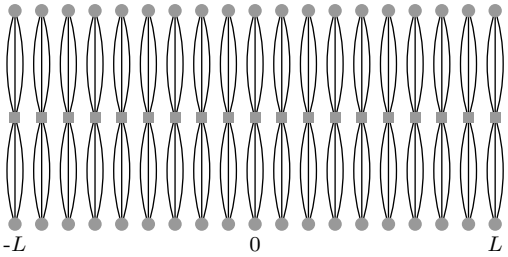


Figure 14.2 A chain of $(2L + 1)$ protographs of the standard $(3, 6)$-regular ensembles for $L = 9$. These protographs do not interact.

a standard $(3, 6)$-regular component. In particular, the belief-propagation (BP) threshold of each protograph is just the standard threshold, call it $\epsilon^{\mathrm{BP}}(d_v = 3, d_c = 6)$. Slightly more generally: start with an $(d_v, d_c = kd_v)$-regular ensemble where d_v is odd so that $\lfloor l/2 \rfloor = (d_v - 1)/2 \in \mathbb{N}$.

We will now "coupled" these copies. To achieve this coupling, connect each protograph to $\lfloor l/2 \rfloor$ protographs "to the left" and to $\lfloor l/2 \rfloor$ protographs "to the right." This is shown in Figure 14.3 for the two cases $(d_v = 3, d_c = 6)$ and $(d_v = 7, d_c = 14)$.

Note that $\lfloor l/2 \rfloor$ extra check nodes are added on each side to connect the "overhanging" edges at the boundary. This reduces the rate of this ensemble from $1 - \frac{d_v}{d_c} = \frac{k-1}{k}$ to

$$\begin{aligned} R(d_v, d_c = kd_v, L) &= \frac{(2L + 1) - (2(L + \lfloor l/2 \rfloor) + 1)/k}{2L + 1} \\ &= \frac{k - 1}{k} - \frac{2\lfloor l/2 \rfloor}{k(2L + 1)}, \end{aligned}$$

Note that this rate loss decreases with the length of the chain. Therefore, in practice we want to pick the length not too small. Of course, this increases the blocklength and so there is a natural trade-off between the block length and the rateloss due to the boundary.

In the above construction we had to assume that d_v was odd and also the "width" of the connection was linked directly to the degree d_v. In this case the construction leads to the very symmetric ensemble. It is not very hard to

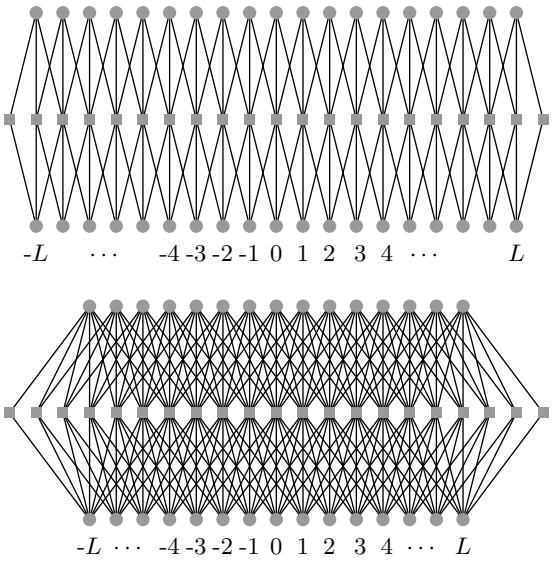


Figure 14.3 Two coupled chains of protographs with $L = 9$ and $(d_v = 3, d_c = 6)$ (top) and $L = 7$ and $(d_v = 7, d_c = 14)$ (bottom), respectively.

extend this construction to cases where d_v is even and so that "width" of the connection is no longer directly linked to d_v. But instead of following this path, let us directly go to another extreme and introduce an ensemble which includes much more randomness.

Random Construction

For the purpose of analysis, the following random ensemble is much betters suited. Let us assume that $d_c \geq d_v$, so that the ensemble has a non-trivial design rate.

We assume that the variable nodes are at positions $[-L, L]$, $L \in \mathbb{N}$. At each position there are M variable nodes, $M \in \mathbb{N}$. Conceptually we think of the check nodes to be located at all integer positions from $[-\infty, \infty]$. Only some of these positions actually interact with the variable nodes. At each position there are $\frac{d_v}{d_c} M$ check nodes. It remains to describe how the connections are chosen.

Rather than assuming that a variable at position i has exactly one connection to a check node at position $[i - \lfloor l/2 \rfloor, \ldots, i + \lfloor l/2 \rfloor]$, we assume that each of the d_v connections of a variable node at position i is uniformly and independently chosen from the range $[i, \ldots, i + w - 1]$, where w is a "smoothing" parameter. In the same way, we assume that each of the d_c connections of a check node at position i is independently chosen from the range $[i - w + 1, \ldots, i]$. We no longer require that d_v is odd.

More precisely, the ensemble is defined as follows. Consider a variable node at position i. The variable node has d_v outgoing edges. A *type* t is a w-tuple of non-

negative integers, $t = (t_0, t_1, \ldots, t_{w-1})$, so that $\sum_{j=0}^{w-1} t_j = d_v$. The operational meaning of t is that the variable node has t_j edges which connect to a check node at position $i + j$. There are $\binom{d_v+w-1}{w-1}$ types. Assume that for each variable we order its edges in an arbitrary but fixed order. A *constellation* c is an d_v-tuple, $c = (c_1, \ldots, c_{d_v})$ with elements in $[0, w-1]$. Its operational significance is that if a variable node at position i has constellation c then its k-th edge is connected to a check node at position $i + c_k$. Let $\tau(c)$ denote the type of a constellation. Since we want the position of each edge to be chosen independently we impose a uniform distribution on the set of all constellations. This imposes the following distribution on the set of all types. We assign the probability

$$p(t) = \frac{|\{c : \tau(c) = t\}|}{w^{d_v}}.$$

Pick M so that $Mp(t)$ is a natural number for all types t. For each position i pick $Mp(t)$ variables which have their edges assigned according to type t. Further, use a random permutation for each variable, uniformly chosen from the set of all permutations on d_v letters, to map a type to a constellation.

Under this assignment, and ignoring boundary effects, for each check position i, the number of edges that come from variables at position $i - j$, $j \in [0, w-1]$, is $M\frac{d_v}{w}$. In other words, it is exactly a fraction $\frac{1}{w}$ of the total number Md_v of sockets at position i. At the check nodes, distribute these edges according to a permutation chosen uniformly at random from the set of all permutations on Md_v letters, to the $M\frac{d_v}{d_c}$ check nodes at this position. It is then not very difficult to see that, under this distribution, for each check node each edge is roughly independently chosen to be connected to one of its nearest w "left" neighbors. Here, "roughly independent" means that the corresponding probability deviates at most by a term of order $1/M$ from the desired distribution. As discussed beforehand, we will always consider the limit in which M first tends to infinity and then the number of iterations tends to infinity. Therefore, for any fixed number of rounds of DE the probability model is exactly the independent model described above.

LEMMA 14.1 (Design Rate) *The design rate of the ensemble (d_v, d_c, L, w), with $w \leq 2L$, is given by*

$$R(d_v, d_c, L, w) = (1 - \frac{d_v}{d_c}) - \frac{d_v}{d_c}\frac{w + 1 - 2\sum_{i=0}^{w}\left(\frac{i}{w}\right)^{d_c}}{2L+1}.$$

Proof Let V be the number of variable nodes and C be the number of check nodes that are connected to at least one of these variable nodes. Recall that we *define* the design rate as $1 - C/V$.

There are $V = M(2L+1)$ variables in the graph. The check nodes that have potential connections to variable nodes in the range $[-L, L]$ are indexed from $-L$ to $L+w-1$. Consider the $M\frac{d_v}{d_c}$ check nodes at position $-L$. Each of the d_c edges of each such check node is chosen independently from the range $[-L-w+1, -L]$. The probability that such a check node has at least one connection in the range

$[-L, L]$ is equal to $1 - \left(\frac{w-1}{w}\right)^{d_c}$. Therefore, the expected number of check nodes at position $-L$ that are connected to the code is equal to $M\frac{d_v}{d_c}(1 - \left(\frac{w-1}{w}\right)^{d_c})$. In a similar manner, the expected number of check nodes at position $-L+i$, $i = 0, \dots, w-1$, that are connected to the code is equal to $M\frac{d_v}{d_c}(1 - \left(\frac{w-i-1}{w}\right)^{d_c})$. All check nodes at positions $-L+w, \dots, L-1$ are connected. Further, by symmetry, check nodes in the range $L, \dots, L+w-1$ have an identical contribution as check nodes in the range $-L, \dots, -L+w-1$. Summing up all these contributions, we see that the number of check nodes which are connected is equal to

$$C = M\frac{d_v}{d_c}[2L - w + 2\sum_{i=0}^{w}(1 - \left(\frac{i}{w}\right)^{d_c})].$$

□

Discussion: In the above lemma we have *defined* the design rate as the normalized difference of the number of variable nodes and the number of check nodes that are involved in the ensemble. This leads to a relatively simple expression which is suitable for our purposes. But in this ensemble there is a non-zero probability that there are two or more degree-one check nodes attached to the same variable node. In this case, some of these degree-one check nodes are redundant and do not impose constraints. This effect only happens for variable nodes close to the boundary. Since we consider the case where L tends to infinity, this slight difference between the "design rate" and the "true rate" does not play a role. We therefore opt for this simple definition. The design rate is a lower bound on the true rate.

Density Evolution

The protograph construction has a slightly better performance if we look at codes of finite length and also, due to the extra structure, it might be easier to implement. On the other hand, the random ensemble is easier to deal with when it comes to proofs. Since asymptotically they behave essentially the same, we concentrate in the sequel on the random case.

The (d_v, d_c, L, w) ensemble is just an LDPC ensemble with some additional structure. It's asymptotic performance can hence again be assessed via density evolution. Therefore, as a first step let us write down the density evolution equations. The only difference compared to the DE equations of the uncoupled ensemble is that now we have a potentially different erasure probability for *every position*. The state is therefore no longer a scalar quantity but a vector of the length of the chain.

DEFINITION 14.2 (Density Evolution of (d_v, d_c, L, w) Ensemble) Let x_i, $i \in \mathbb{Z}$, denote the average erasure probability which is emitted by variable nodes at position i. For $i \notin [-L, L]$ we set $x_i = 0$. For $i \in [-L, L]$ the FP condition

implied by DE is

$$x_i = \epsilon\Big(1 - \frac{1}{w}\sum_{j=0}^{w-1}\big(1 - \frac{1}{w}\sum_{k=0}^{w-1} x_{i+j-k}\big)^{d_c-1}\Big)^{d_v-1}. \tag{14.1}$$

If we define

$$y_i = \Big(1 - \frac{1}{w}\sum_{k=0}^{w-1} x_{i-k}\Big)^{d_c-1}, \tag{14.2}$$

then (14.1) can be rewritten as

$$x_i = \epsilon\Big(1 - \frac{1}{w}\sum_{j=0}^{w-1} y_{i+j}\Big)^{d_v-1}.$$

EXIT Curves

As for uncoupled ensembles we can draw EXIT curves for the coupled case. Recall that in the uncoupled case, the EXIT curve is a plot of the channel parameter ϵ as a function of the EXIT value $(1-(1-x)^{r-1})^l$, see e.g., Figure 10.4. In the uncoupled case we had a simple analytical formula for this curve. For the coupled case, no such formula exists, but one can compute the curves numerically.

Figure 14.4 shows the EXIT curves for the $(d_v = 3, d_c = 6, L)$ for $L = 1, 2, 4, 8, 16, 32, 64$, and 128. Note that these EXIT curves show a dramatically

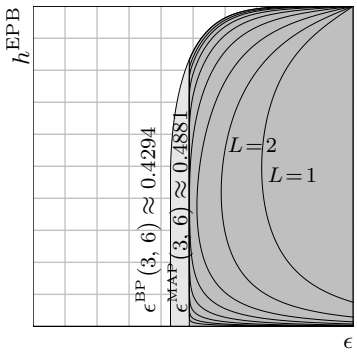


Figure 14.4 EBP EXIT curves of the ensemble $(d_v = 3, d_c = 6, L)$ for $L = 1, 2, 4, 8, 16, 32, 64$, and 128. The BP/MAP thresholds are $\epsilon^{\mathrm{BP/MAP}}(3,6,1) = 0.714309/0.820987$, $\epsilon^{\mathrm{BP/MAP}}(3,6,2) = 0.587842/0.668951$, $\epsilon^{\mathrm{BP/MAP}}(3,6,4) = 0.512034/0.574158$, $\epsilon^{\mathrm{BP/MAP}}(3,6,8) = 0.488757/0.527014$, $\epsilon^{\mathrm{BP/MAP}}(3,6,16) = 0.488151/0.505833$, $\epsilon^{\mathrm{BP/MAP}}(3,6,32) = 0.488151/0.496366$, $\epsilon^{\mathrm{BP/MAP}}(3,6,64) = 0.488151/0.492001$, $\epsilon^{\mathrm{BP/MAP}}(3,6,128) = 0.488151/0.489924$. The light/dark gray areas mark the interior of the BP/MAP EXIT function of the underlying $(3,6)$-regular ensemble, respectively.

different behavior compared to the EBP EXIT curve of the underlying ensemble. These curves appear to be "to the right" of the threshold $\epsilon^{\mathrm{MAP}}(3,6) \approx 0.48815$.

For small values of L one might be led to believe that this is true since the design rate of such an ensemble is considerably smaller than $1 - d_v/d_c$. But even for large values of L, where the rate of the ensemble is close to $1 - d_v/d_c$, this dramatic increase in the threshold is still true. Empirically we see that, for L increasing, the EBP EXIT curve approaches the MAP EXIT curve of the underlying $(d_v = 3, d_c = 6)$-regular ensemble. In particular, for $\epsilon \approx \epsilon^{\text{MAP}}(d_v, d_c)$ the EBP EXIT curve drops essentially vertically until it hits zero.

Decoding Wave

"The" key to understanding why spatially coupled ensembles perform so well is to study their FPs under density evolution. Recall that for uncoupled ensembles the FPs are scalars. For the coupled case the state of the system is no longer a scalar but a vector, where the length of the vector is equal to the length of the chain. Due to this fact, there are some very interesting FPs which appear.

Assume we are operating much above the threshold. Let us assume that we decode until we are stuck and let us plot the final erasure probability at each section along the chain. Then it is reasonably to expect that this erasure probability is equal to the erasure probability which we would observe for an uncoupled ensemble. The only exception are positions very close to the boundary where the behavior is a little bit better due to the extra information we have there. The top picture in Figure 14.6 shows this situation together with the position of the FP on the EXIT curve. Since the FP is symmetric with respect to the middle of the chain, only one half is shown. Imagine that we now slowly lower the erasure probability of the channel. Due to the improved conditions at the boundary, the "effective" erasure probability at the boundary will at some point be below the BP threshold of the uncoupled ensemble and the BP decoder will be able to decode the bits at the boundary. But once these bits are decoded this will lower the "effective" erasure probability for bits a little bit further into the chain. This effect propagates like a wave and the whole chain will get decoded. The middle and the bottom picture in Figure 14.6 show the wave in various stages.

The perhaps the most surprising aspect is that the BP threshold for the coupled chain is exactly the area threshold of the uncoupled one.

¡¡¡¡¡¡¡ .mine Figure 14.6 shows the FP for various parameters of the channel together with the position of the FP on the EXIT curve. Since the FP is symmetric with respect to the middle of the chain, only one half is shown.

Main Statement

THEOREM 14.3 (BP Threshold of the (d_v, d_c, L, w) Ensemble) *Consider transmission over the BEC(ϵ) using random elements from the ensemble (d_v, d_c, L, w). Let $\epsilon^{BP}(d_v, d_c, L, w)$ denote the BP threshold and let $R(d_v, d_c, L, w)$ denote the design rate of this ensemble.*

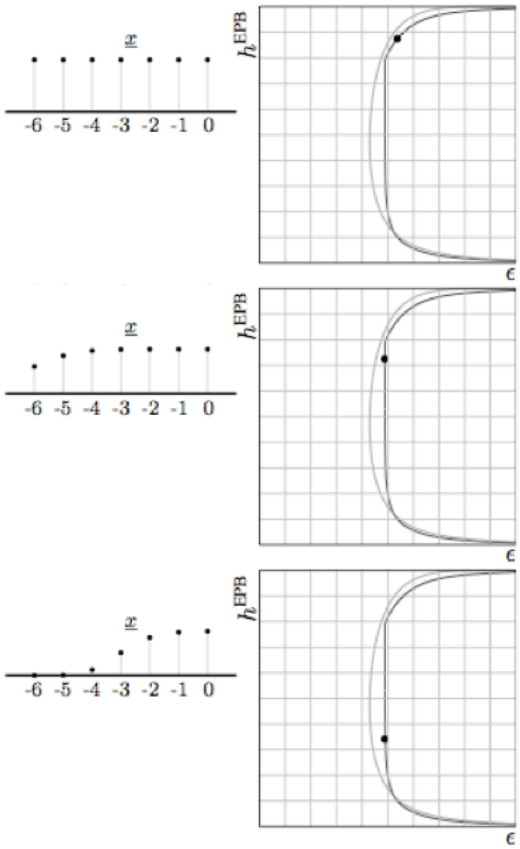


Figure 14.5 FPs for various parameters of the channel together with the position of the FP on the EXIT curve.

Figure 14.6 FPs for various parameters of the channel together with the position of the FP on the EXIT curve.

Then, in the limit as M tends to infinity, and for w sufficiently large

$$\epsilon^{BP}(d_v,d_c,L,w) \leq \epsilon^{MAP}(d_v,d_c,L,w) \leq \epsilon^{MAP}(d_v,d_c) + \frac{w-1}{2L(1-(1-x^{MAP}(d_v,d_c))^{d_c-1})^{d_v}}, \tag{14.3}$$

$$\epsilon^{BP}(d_v,d_c,L,w) \geq \left(\epsilon^{MAP}(d_v,d_c) - w^{-\frac{1}{8}} \frac{8d_vd_c + \frac{4d_cd_v^2}{(1-4w^{-\frac{1}{8}})^{d_c}}}{(1-2^{-\frac{1}{d_c}})^2}\right) \times \left(1-4w^{-1/8}\right)^{d_cd_v}. \tag{14.4}$$

In the limit as M, L and w (in that order) tend to infinity,

$$\lim_{w\to\infty}\lim_{L\to\infty} R(d_v, d_c, L, w) = 1 - \frac{d_v}{d_c}, \tag{14.5}$$

$$\begin{aligned}\lim_{w\to\infty}\lim_{L\to\infty} \epsilon^{BP}(d_v, d_c, L, w) &= \lim_{w\to\infty}\lim_{L\to\infty} \epsilon^{MAP}(d_v, d_c, L, w)\\ &= \epsilon^{MAP}(d_v, d_c).\end{aligned} \tag{14.6}$$

Roughly speaking, the above theorems states that the BP threshold of the coupled chain is equal to its MAP threshold and also to the MAP threshold of the uncoupled chain. The statements in the theorem are considerably weaker than what can be observed empirically. In particular, the convergence with respect to the coupling width is conjectured to be exponential in w.

A very similar statement can be shown to hold for transmission over general channels. In particular, one can show that these ensembles are good universally for the whole class of BMS channels.

14.2 Compressive Sensing

The idea of spatial coupling can also be used in compressive sensing to attain optimal performance by message passing. In a nutshell, the idea is to construct appropriate sensing matrices that correspond to a "spatially coupled" factor graph and then to apply an AMP type algorithm. The performance of the algorithm is then analyzed through a state evolution recursion tailored to the spatially coupled graph. This turns out to be a one-dimensional recursion which displays similar phenomena than those described for the BEC.

In Chapter 8 our starting point was the Lasso estimator which is a reasonable starting point to develop a universal algorithm that doe not assume a prior knowledge of the signal distribution in the class $\mathcal{F}_\epsilon$. Recall that the state evolution equation in Chapter **??** has at most one fixed point. Therefore, intuitively, one does not expect that any improvement in performance can be obtained by spatial coupling. This has indeed been corroborated by numerical simulations. We will therefore turn our attention to a setting where the prior distribution of the signal is known.

AMP when the prior is known

We assume that the signal distribution is from the class $\mathcal{F}_\epsilon$ and that it is known. In other words $p_0(x) = (1-\epsilon)\delta_0(x) + \epsilon\phi_0(x)$ for a known $\phi_0(x)$ (for example a Gaussian distribution). As explained in Chapter 3, in this setting the optimal estimator is the MMSE estimator (3.35). In Chapter 5 we went through the belief propagation equations in Example 16. This approach can be systematically developed in order to recursively compute the BP-estimate. Furthermore, following the same route as in Chapter 8, these message-passing equations can be

simplified in order to arrive at an AMP algorithm that is very similar to (8.37). By skimming through the previous chapters one can almost guess the form of the new algorithm.

In (8.37) the update of the AMP-estimate uses the soft thresholding function $\eta(y,\lambda)$ found by solving the scalar Lasso problem. The reader should not be too surprised that now the AMP updates involve a thresholding function given by the MMSE estimator of the scalar case. Consider a scalar measurement $y = x + \nu z$ of "signal" x affected by Gaussian noise with variance ν^2 (so $Z \sim N(0,1)$) the thresholding function is

$$\eta_0(y,\nu) = \mathbb{E}[X|X+\nu Z = y] = \frac{\int dx\, x\, p_0(x) e^{-\frac{(y-x)^2}{2\nu^2}}}{\int dx\, p_0(x) e^{-\frac{(y-x)^2}{2\nu^2}}}.$$

We stress that $\eta_0(y,\nu)$ is not universal and depends on the prior. Here ν plays the role of a threshold level analogous to λ in the Lasso case. It will be adjusted at each AMP iteration. The mean square error for this optimal estimator (of the scalar problem) is the MMSE function[1]

$$\begin{aligned}\mathrm{mmse}(\nu^{-2}) &= \mathbb{E}\big[(X - \mathbb{E}[X|X+\nu Z])^2\big] \\ &= \int dx\, p_0(x) \int dz\, \frac{e^{-\frac{z^2}{2}}}{\sqrt{2\pi}} (x - \eta_0(x+\nu z,\nu))^2.\end{aligned}$$

The AMP updates are the same than in Chapter 8 except η is replaced by η_0,

$$\hat{x}_i^{(t+1)} = \eta_0\big(x_i^{(t)} + \sum_{a=1}^{m} A_{ai} r_a^{(t)}, \nu^{(t)}\big), \tag{14.7}$$

$$r_a^{(t)} = y_a - \sum_{j=1}^{n} A_{aj} \hat{x}_j^{(t-1)} + b^{(t)} r_a^{t-1}. \tag{14.8}$$

If you go back to the derivation of the Onsager term in Chapter 8 you will see that it can be traced back to a derivative of the soft thresholding function. You can guess that now

$$b^{(t)} = \frac{1}{\delta n} \sum_{i=1}^{n} \eta_0'\big(x_i^{(t-1)} + \sum_{a=1}^{m} A_{ai} r_a^{(t-1)}, \nu^{(t)}\big). \tag{14.9}$$

Similarly recall that in Chapter **??** we expressed the threshold level $\nu^{(t)}$ thanks to the MSE through (8.49). Here one arrives at the same conclusion, namely

$$(\nu^{(t)})^2 = \sigma^2 + \frac{1}{\delta}(\tau^{(t)})^2, \tag{14.10}$$

where $\tau^{(t)2}$ is the average (normalized) MSE of the AMP algorithm $(\tau^{(t)})^2 = \lim_{n\to+\infty} \frac{1}{n}\mathbb{E}\|\underline{\hat{x}}^{(t)} - \underline{x}_0\|_2$. We can track its evolution thanks to the recursion (same as (**??**) with correct η_0-function)

$$(\tau^{(t+1)})^2 = \mathrm{mmse}((\nu^{(t)})^{-2}). \tag{14.11}$$

[1] By convention the argument of the MMSE function is a signal-to-noise-ratio, here ν^{-2}.

In hindsight one can develop an interpretation for this equation: at time $t+1$ the total quadratic error $(\tau^{(t+1)})^2$ for the AMP estimate is given by the MMSE of a scalar signal with effective noise variance $\sigma^2 + \frac{1}{\delta}(\tau^{(t)})^2$ at time t.

Let us summarize. Equations (14.11) and (14.10) give the evolution of the MSE and the threshold level. These quantities can be precomputed. Equations (14.8) and (14.9) define the AMP algorithm, and allow to compute the estimates for the signal.

Construction of the measurement matrix

Let us first explain the general idea. In the standard case considered so far, the measurement matrices have iid entries $A_{ai} \sim \mathcal{N}(0, \frac{1}{\sqrt{m}})$ so that "their factor graph" is a complete bipartite graph with m checks and n variables. The ratio $\delta = m/n$ is the sampling rate. Inspired by the construction of spatially coupled codes one may try to use matrices associated to a spatial chain of L complete bipartite graphs coupled across a window of size w. This turns out to be a successful idea! The sampling rate is still equal to δ in the bulk of the chain. At the boundary one has to add extra check nodes or equivalently one has to oversample. Indeed, in order to create a seed that gets the nucleation process started one needs a good estimate of the first few components of the signal. The increase in sampling rate is negligible in the thermodynamic limit.

In practice, because the AMP algorithm updates purely local quantities (the BP messages flowing along edges have been eliminated), one can forget about the factor graph and specify directly the sensing matrix. You can convince yourself that the sensing matrix described here has a factor graph that is a chain of coupled complete bipartite graphs. There are many possible constructions and ways to optimize the finite length performance. But these issues will not concern us here, and we discus a similar construction which is similar to the one presented in the coding case.

The signal has n components in total and we make m measurements. The measurement matrix has n columns and m rows. Think of n given and m to be determined later. Partition the columns in L groups[2] $c \in \{1, \dots, L\}$ with N columns each, so $N = n/L$. Consider $L + w - 1$ groups of rows $r \in \{-(w-2), \dots, 0, 1, \dots, L\}$, each with $M = \delta N$ rows. The total number of measurements is $m = (w-1)M + ML = \delta n(1 + (w-1)/L)$. The contribution of the oversampling rate to the total rate $m/n = \delta(1 + (w-1)/L)$ vanishes for large L.

Now consider an $(L + w - 1) \times L$ matrix of variances J_{rc}. A simple choice is

$$J_{r,c} = \begin{cases} \frac{1}{2w-1} & \text{if } \ c \in \{r - w + 1, \dots, r + w - 1\} \\ 0 & \text{otherwise} \end{cases}$$

Here we use a simple square-like and symmetric shape function for $J_{r,c}$. One can generalize this to $J_{r,c} = \rho \mathcal{J}(\rho|r - c|)$ with $\rho = (2w-1)^{-1}$ and a shape function

[2] One can visualize the groups as positions along the chain.

$\mathcal{J}(z)$ that is positive, supported on $[-1,+1]$ and $\int_{-1}^{+1} dz\, \mathcal{J}(z) = 1$. Let us also note that taking larger variances for the seeding part of the matrix may lead to better performance. In the sequel all equations are valid for general choices of $J_{r,c}$.

To specify the matrix elements of A_{ai}, we introduce the notation $R(a)$ and $C(i)$ for the groups (r and c) to which row a and column i belong. A simple choice is to take iid entries

$$A_{ai} \sim (0, \frac{1}{M} J_{R(a),C(i)})$$

We notice that by construction we have the normalization $\sum_i A_{ai}^2 \approx 1$, as in the standard (uncoupled) case. This matrix has a band structure with a band of height and width $wM \times wN$. However the correct regime in which the spatially coupled model is used is $N >> L$ so effectively the matrix is "full".

Spatially coupled AMP

The starting point - the BP equations - are exactly the same except they are applied to a bigger factor graph. The derivation of the coupled AMP algorithm then proceeds in the usual way by retaining only important terms *in the regime* $N \to +\infty$ and L fixed.

It turns out that the resulting equations have a few extra complications. Namely, due to coupling, the sensing matrix elements get "renormalized" and the threshold level as well as the Onsager term get "averaged". The AMP equations now read

$$\hat{x}_i^{(t+1)} = \eta_0\big(x_i^{(t)} + \sum_{a=1}^{m} Q^{(t)}_{R(a),C(i)} A_{ai} r_a^{(t)}, \nu^{(t)}_{C(i)}\big) \tag{14.12}$$

$$r_a^{(t)} = y_a - \sum_{j=1}^{n} A_{aj}\hat{x}_j^{(t-1)} + b^{(t)}_{R(a)} r_a^{t-1} \tag{14.13}$$

where

$$b^{(t)}_{R(a)} = \frac{1}{\delta}\sum_{c=1}^{L} J_{R(a),c} Q^{t-1}_{R(a),c}\left\{\frac{1}{N}\sum_{i\,\mathrm{s.t}\,C(i)=c} \eta_0'\big(x_i^{(t)} + \sum_{b=1}^{m} Q^{(t)}_{R(b),C(i)} A_{bi} r_b^{(t)}, \nu^{(t)}_{C(i)}\big)\right\}$$

The threshold levels $\nu^{(t)}_{C(i)}$ and the weights $Q_{R(a),C(i)}$ depend only on the *local* MSE $(\tau_c^{(t)})^2 = \lim_{N\to+\infty} \frac{1}{N}\sum_{i\,\mathrm{s.t}\,C(i)=c} \mathbb{E}\|\hat{x}_i^{(t)} - x_{0,i}\|_2^2$. These quantities can all be pre-computed from state evolution. The threshold level is given by (a generalization of (14.10))

$$(\nu_c^{(t)})^{-2} = \sum_r J_{r,c}\big(\sigma^2 + \frac{1}{\delta}\sum_c J_{r,c}(\tau_c^{(t)})^2\big)^{-1}, \tag{14.14}$$

This equation says that the threshold for estimates of the signal components in group c is given by an average of the signal to noise ratios for measurements in

the groups $r \in \{c-w+1, \dots, c+w-1\}$, and the later are themselves given by an average of the local MSE in the groups $c \in \{r-w+1, \dots, r+w-1\}$. The sensing matrix gets renormalized by weights

$$Q_{r,c} = \frac{\left(\sigma^2 + \frac{1}{\delta}\sum_c J_{r,c}(\tau_c^{(t)})^2\right)^{-1}}{\sum_r J_{r,c}\left(\sigma^2 + \frac{1}{\delta}\sum_c J_{r,c}(\tau_c^{(t)})^2\right)^{-1}}.$$

Finally, the local MSE evolves as

$$(\tau_c^{(t+1)})^2 = \mathrm{mmse}((\nu_c^{(t)})^{-2}), \qquad c = 1, \dots, L \tag{14.15}$$

Equations (14.14)-(14.15) are the one dimensional state evolution recursion and can be used to derived the performance of AMP on the spatially coupled model. The reader should ponder on this recursion and realize that its structure is perfectly analogous to the DE recursion in coding for the BEC.

Analysis of Performance and Phase Diagram

The discussion in this paragraph is valid for a fairly wide class of functions $\phi_0(x)$, but a good exercise for the reader is to verify the claims for a Gaussian $\phi_0(x)$. This can be done analytically for the uncoupled case and numerically in the coupled case. Notice that in this case $\eta_0(y,s)$ can be explicitly be computed.

Consider the recursion (14.11) and look at the corresponding fixed point equation. Let

$$\tilde{\delta}(p_0) \equiv \sup_{\nu}\{\nu^{-2}\mathrm{mmse}(\nu^{-2})\} > \epsilon$$

Here the equality is definition. The inequality is a fact, which follows by remarking $\lim_{\nu\to 0}\nu^{-2}\mathrm{mmse}(\nu^{-2}) = \epsilon$. For a sampling rate $\delta > \tilde{\delta}(p_0)$ there exists only one fixed point solution $(\tau_{\mathrm{good}})^2 = O(\sigma^2)$. This corresponds to correct reconstruction in the small noise limit $\sigma \to 0$. Now, decrease the sampling rate in the range $\epsilon < \delta < \tilde{\delta}(p_0)$. One finds two or more stable fixed points (as well as unstable ones) for all $\sigma^2 > 0$. Besides the "good" fixed point satisfies $(\tau_{\mathrm{good}})^2 = O(\sigma^2)$ there is a "bad" one, i.e. $(\tau_{\mathrm{bad}})^2 = \Theta(1)$ as $\sigma \to 0$. Under the (natural) initial condition $(\tau^0)^2 = +\infty$ one always tends to $(\tau_{\mathrm{bad}})^2$. This means that the noise sensitivity $\lim_{\sigma\to 0} MSE/\sigma^2$ diverges, and exact reconstruction is not possible even for very small noise. In this context $\tilde{\delta}(p_0)$ is the algorithmic threshold of AMP. The analogous quantity in our coding model is ϵ_{BP} and it the CW model it is the spinodal point.

This threshold is lower than the Lasso (or l_1) threshold derived in Chapter **??**. This is not too surprising since the later concerns the worst case distribution for $p_0 \in \mathcal{F}_\epsilon$. It is instructive to compute the phase diagram and plot the optimal, Lasso and AMP phase transition lines in the (ϵ, δ) plane.

Let us now turn our attention to the coupled model. The performance is analyzed through the one dimensional recursion (14.14)-(14.15) which gives the evolution of the MSE profile $\tau_c^{(t)}$, as a function of time t and position along the

chain $c = 1, \dots, L$. For $\delta > \tilde{\delta}(p_0)$ the local MSE tends to $(\tau_{c,\text{good}})^2 = O(\sigma^2)$ uniformly along the chain. The advantage brought by spatial coupling appears for a sampling rate in the range $\epsilon < \delta < \tilde{\delta}(p_0)$. For $L \to +\infty$ and fixed $w \geq 2$ there is a $\tilde{\delta}(p_0, w) < \tilde{\delta}(p_0)$ such that for $\delta > \tilde{\delta}(p_0, w)$ the local MSE per position is bounded by $O(\sigma^2)$, and in particular the noise sensitivity remains finite. Because of the oversampling of the first few signal components, the MSE falls down to a level $O(\sigma^2)$ for these components, and then an estimation wave propagates along the chain. Eventually the local MSE converges to the good fixed point for all positions $\tau_{\text{good},c} = O(\sigma^2)$. Furthermore one observes that $\tilde{\delta}(p_0, w) \to \epsilon$ as $w \to +\infty$. In other words in the regime $N >> L >> w >> 1$ the dynamical AMP threshold saturates towards the optimal phase transition threshold. Figure **??** illustrates the phase diagram and the phase transition lines in teh (ϵ, δ) plane for various values of L and w.

14.3 K-SAT

For the random K-SAT problem we discussed several algorithms. The best one is BP-guided decimation. We described this algorithm and its empirical performance in Chapter 9.3. If we apply spatial coupling to this algorithm we see no boost in performance. This does not mean that spatial coupling does not help for this problem. It just means that BP-guided decimation is not the right setting for the nucleation phenomenon. The "right" setting is in fact a more sophisticated algorithm called *survey propagation*.

Rather than pursuing this avenue, let us go to a simpler algorithm, namely the UCP algorithm which we discussed in Chapter 9. We will see that spatially coupled formulas have a significantly higher threshold under UCP than uncoupled ones. Combined with the interpolation method this gives good lower bounds on the SAT/UNSAT threshold of uncoupled systems.

Construction

As for the case of coding, there are various ways of constructing coupled K-SAT formulas. E.g., Figure 14.7 shows the equivalent of a protograph ensemble for the case $K = 3$ where each clause at position i has exactly one connection to a variable at position i, $i + 1$, and $i + 2$.

For the purpose of analysis it is again more convenient to consider a random ensemble. As before, let w be a window size. Then, for each clause at position i and for each of its K connections we independently and uniformly pick a variable at a position in the range $[i, i + w - 1]$ and connect it to this variable with a uniformly chosen sign. This is the ensemble which we consider in the sequel.

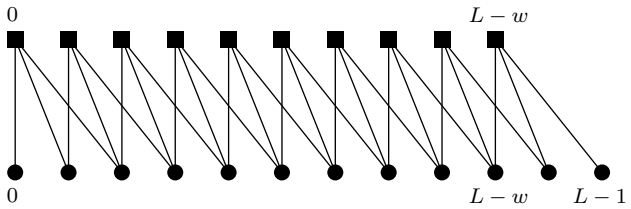


Figure 14.7 A "protograph"-like coupled K-SAT ensembles or $K = 3$.

Performance under the UCP Algorithm

Let us now focus on the UC algorithm for the coupled formulas. As for the un-coupled case, the UC algorithm consists of two main steps: free and forced. The operation of the algorithm at a forced step is clear: remove all the unit-clauses until no further unit-clause exists. However, at a free step, depending on how we might want to use the chain structure of the formula, we can have different *schedules* for choosing a free variable. For a coupled formula, the schedule within which we are choosing a variable in a free step is important

Consider for instance the following naive schedule – at a free step, pick a variable uniformly at random from all the remaining variables and fix it by flipping a coin. Computer experiments indicate that this naive schedule has no threshold gain compared to the un-coupled ensemble. This is not surprising since this schedule does not exploit the spatial (chain) structure of the formula. Hence, in order for the UC algorithm to have a threshold improvement over the coupled ensemble, we need to come up with schedules that exploit the additional spatial structure of the formula. We proceed by illustrating one such successful schedule.

In the very beginning of the algorithm, all the check nodes have degree K and there are no unit clauses. Hence, we are free to fix the variables in the first few steps of the algorithm. Let us fix the variables from the left-most position (i.e., the boundary). If we do this then we are creating in effect a seed at the boundary of the chain. Continuing this action at the free steps, we will eventually create unit clauses and at these forced steps a natural choice is just to clear all the unit clauses. However, when we are confronted with a free step, we will again try to help this seed to grow inside the chain, i.e., we always fix variables from the left-most possible position. Consequently, the schedule that we apply is as follows.

- At a *free step*, pick a variable randomly from the left-most position at which variables exists and fix it permanently by flipping a fair coin.
- At a *forced step*, remove unit clauses as long as they exist.

Computer experiments show that this schedule indeed exhibits a threshold improvement over the un-coupled ensemble. E.g., for the coupled 3-SAT problem, experiments suggest that the threshold of the UC algorithm is around 3.67. This

is a significant improvement compared to the threshold of UC for the un-coupled ensemble which is $\frac{8}{3}$.

To prove that indeed this schedule leads to this threshold we use again the Wormald method. This means, we write down a set of differential equations which describe the expected progress of the algorithm. Not surprisingly, the number of differential equations we need scales linearly in the chain length.

Phases, Types, and Rounds

For the coupled ensemble, the analysis of the evolution of UC is much more involved than the un-coupled ensemble. This is because of the fact that the schedule we have used prefers the left-most variable position in a free step. Hence, the number of variables in different positions will evolve differently. As an example, one can easily see that during the algorithm, the first position that all its variables are set is the left-most position (i.e., position 0). After the evacuation of position 0, position 1 becomes the left-most position of the graph and hence, the second position that becomes empty of variables is position 1. Continuing in this manner, the last position that is evacuated is position $L + w - 2$. With these considerations, we consider $L+w-1$ *phases* for this algorithm (see Figure 14.8). At phase $p \in \{0, 1, \cdots, L + w - 2\}$, all the variables at positions prior to p have been set permanently and as a result, at a free step we will pick a variable from position p.

This statistical asymmetry in the number of variables at each position also affects the the behavior of the number of check nodes in each position. As a result, we consider *types* for the check nodes. For instance, consider a degree two check node. It is easy to see that the probability that this degree two check node is hit (removed or shortened) is greatly dependent on the position of variables that it is connected to. This means that, dependent on the variable positions to which they are connected, we have different types of degree two check nodes. Clearly, the same statement holds for clauses of degree three, four, etc.

Let us now formally define the ingredients needed for the analysis. The notation we use here is slightly hard to swallow immediately. Thus, for the sake of maximum clarity, we try to uncover the details as smoothly as possible. We consider *rounds* for this algorithm. Each round consists of one free step followed by the forced steps that follow it. More precisely, at the beginning of each round we perform a free step and then we clear out all the unit-clauses as long as they exist (forced steps). We let time t be the number of rounds passed so far. This time variable will be called *round time*. The relation between t and the *natural time* (the total number of permanent fixes) is not linear. We also let $L_i(t)$ be the *number of literals* left in variable position $i \in \{0, 1, \cdots, L + w - 2\}$.

We now define the check types. Consider a coupled K-SAT formula to begin with. For such a formula there are L sets of check nodes placed at positions $\{0, 1, \cdots, L\}$. Let us consider a specific position $i \in \{0, 1, \cdots, L\}$ and look at the check nodes at position i. Each of these check nodes can potentially be connected

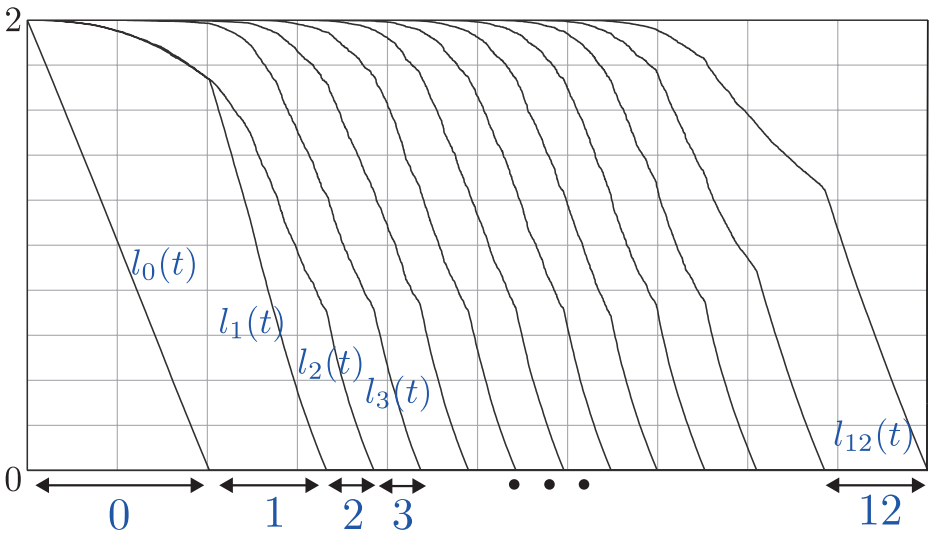


Figure 14.8 A schematic representation of how the literals at each of the positions vary in time. The horizontal axis corresponds to time t which is the number of free steps. Here we have $L = 11$ and $w = 3$. This plot corresponds to an implementation of the UC algorithm on a random coupled instance. The blue numbers below the plot are the phases of the algorithm. In the beginning of the algorithm, we are in phase 0. This phase lasts until all the literals in the first position are peeled off and as a result $\ell_0(t)$ reaches 0. We then go immediately to phase 1 and this phase lasts till $\ell_1(t)$ reaches 0 and so on. We have in total $L + w - 1 = 13$ phases.

to any set of K variables resting in variable positions $\{i, i+1, \cdots, i+w-1\}$. Some thought shows that there are various types of check nodes depending on the variable positions that they are connected to. For example, there is a type of check nodes for which all of the K edges go only into a single variable position $j \in \{i, i+1, \cdots, i+w-1\}$ or there is a type for with some of its edges go to position i and the rest go to position $i+1$ and so on. Also, as we proceed through the UC algorithm, some of these checks are shortened to create new types of checks with degrees less than K. We now explain a natural way to encode these various types.

By $C(t, i, \underline{\tau})$ we mean the number of check nodes at check position $i \in \{0, 1, \cdots, L\}$ that have type $\underline{\tau}$ at round tme t. The type $\underline{\tau} = (\tau_0, \cdots, \tau_{w-1})$ is a w-tuple and indicates that relative to position i, how many edges the check has in (variable) positions $i, i+1, \cdots, i+w-1$. The best way to explain $\underline{\tau}$ is through an example. Let us assume $w = 4$ and consider the set of check nodes at check position 20 that are only connected to variable positions $20, 22, 23$ in the following way. For each of these check nodes there are exactly two edges going to position 20, and 1 edge going to position 22 and 1 edge going to position 23 (thus each of these checks have degree 4). Figure 14.9 illustrates a generic check node of this set.

We denote the number of these checks at time t by $C(t, 20, (1, 0, 2, 1))$. In

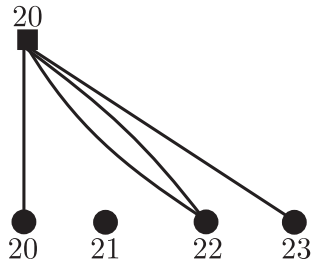


Figure 14.9 A schematic representation of checks which contribute to $C(t, 20, (1, 0, 2, 1))$. All the check nodes that contribute to $C(t, i, \underline{\tau})$, were initially (at time 0) degree K check nodes resting at check position i. However, the algorithm has evolved in a way that these check nodes have been deformed (possibly shortened or remained unchanged) to have a specific type $\underline{\tau}$.

other words, the type is computed as follows: the check position number that the check rests in is 20. This check is connected to a variable at position 20, and 2 variables at position 22, and a variable at position 23. So, relative to the check position 20, we see the edge-tuple $(1, 0, 2, 1)$. Let us now repeat and generalize: By $C(t, i, \underline{\tau})$ we mean the number of check nodes, at time t, which rest in position i, and $\underline{\tau}$ is a w-tuple that indicates relative to variable position i, the number of edges that go to positions $i, i+1, \cdots, i+w-1$, respectively. One can easily see that by summing up elements of the w-tuple $\underline{\tau} = (\tau_0, \cdots, \tau_{w-1})$, we find the degree of the corresponding check type. We denote the degree of a type $\underline{\tau}$ by $\deg(\underline{\tau})$. It is also easy to see that there are $\binom{d+w-2}{d-1}$ different types of degree d for $d \in \{2, 3, \cdots, K\}$. We are now ready to write the differential equations. Our approach is as follows. Assume the phase of the algorithm is p and we are in a round t. At a free step, we fix a variable at position p (free step). This will create a number of forced steps in each of the positions $p, p+1, \cdots, L+w-1$. We first compute the average of these forced fixes in each variable position as a function of the number of degree two check nodes. Using these averages, we then update the average number of check and variable nodes at each position. We proceed by explaining a key property for the analysis.

The Differential Equations

Now, having the vector $\underline{\beta}$ we can find how the number of variables and checks evolve. For all $i \geq 0$,

$$\Delta L_i(t) = L_i(t+1) - L_i(t) = -2\beta_i(t). \tag{14.16}$$

To see how the check types evolve, we note that for a given check type there are two kinds of flows to be considered. A negative flow going out and a positive flow coming in from the checks of higher degrees. In this regard, for a type $\underline{\tau} = (\tau_0, \cdots, \tau_{w-1})$ with $\deg(\underline{\tau}) < K$ let $\partial\underline{\tau}$ be the set of types of degree $\deg(\underline{\tau})+1$

such that by removing one edge from them we reach to the type $\underline{\tau}$. The set $\partial\underline{\tau}$ consists of w types which we denote by $\underline{\tau}^d$, $d \in \{0, 1, \cdots, w-1\}$, such that

$$\underline{\tau}^d = \underline{\tau} + (0, \cdots, \overset{d}{1}, \cdots, 0), \tag{14.17}$$

where $+$ denotes vector addition in the field of reals. Thus, if $\deg(\underline{\tau}) < K$, we obtain

$$\Delta C(t, i, \underline{\tau}) = -2 \sum_{d=0}^{w-1} \beta_{i+d}(t) \frac{\tau_d c(t, i, \underline{\tau})}{L_{i+d}(t)} + \sum_{d=0}^{w-1} (1 + \tau_d) \beta_{i+d}(t) \frac{C(t, i, \underline{\tau}^d)}{L_{i+d}(t)}. \tag{14.18}$$

The right-hand side of (14.18) has two parts. The first part corresponds to the flow that is going out of $C(t, i, \underline{\tau})$ and has negative sign. The right part is the incoming flow from the check nodes of higher degrees. In the case where $\deg(\underline{\tau}) = K$, we only have an outgoing flow since no check node with higher degrees exist. Hence, for the case $\deg(\underline{\tau}) = K$ we can write

$$\Delta C(t, i, \underline{\tau}) = -2 \sum_{d=0}^{w-1} \beta_{i+d}(t) \frac{\tau_d c(t, i, \underline{\tau})}{L_{i+d}(t)}. \tag{14.19}$$

We now write the initial conditions for the variables and check types. Firstly, note that $L_i(0) = 2N$. In the beginning of the algorithm, all checks are of degree K, thus for types $\underline{\tau}$ such that $\deg(\underline{\tau}) < K$, we have $C(0, i, \underline{\tau}) = 0$. For $\deg(\underline{\tau}) = K$ we have

$$C(0, i, \underline{\tau}) = \alpha N \frac{\binom{K}{\tau_0, \tau_1, \cdots, \tau_{w-1}}}{w^K}. \tag{14.20}$$

In order to write the differential equations, we re-scale the (round) time by N, i.e.

$$t \leftarrow \frac{t}{N}, \tag{14.21}$$

and also normalize all our other numbers by N, i.e.,

$$c(t, \cdot, \cdot) = \frac{C(Nt, \cdot, \cdot)}{N} \text{ and } \ell_i(t) = \frac{L_i(Nt)}{N}. \tag{14.22}$$

We then obtain for $i \in \{0, 1, \cdots, L + w - 2\}$,

$$\frac{d\ell_i(t)}{dt} = -2\beta_i(t). \tag{14.23}$$

For $i \in \{0, 1, \cdots, L-1\}$ and $\deg(\underline{\tau}) < K$ we have

$$\frac{dc(t, i, \underline{\tau})}{dt} = -2 \sum_{d=0}^{w-1} \beta_{i+d}(t) \frac{\tau_d c(t, i, \underline{\tau})}{\ell_{i+d}(t)} + \sum_{d=0}^{w-1} (1 + \tau_d) \beta_{i+d}(t) \frac{c(t, i, \underline{\tau}^d)}{\ell_{i+d}(t)}, \tag{14.24}$$

and otherwise if $\deg(\underline{\tau}) = K$ we have

$$\frac{dc(t, i, \underline{\tau})}{dt} = -2 \sum_{d=0}^{w-1} \beta_{i+d}(t) \frac{\tau_d c(t, i, \underline{\tau})}{\ell_{i+d}(t)}. \tag{14.25}$$

K	3	4	5
$\alpha_{\mathrm{UC}}(K)$	2.66	4.50	7.58
$\alpha_{\mathrm{UC},L=50,w=3}(K)$	3.67	7.81	15.76

Table 14.1 *First line*: The thresholds for UCP on the uncoupled ensemble. *Second line*: UCP threshold for a coupled chain with $w = 3$, $L = 50$.

The vector $\bar{\beta}$ is also found as follows. For p being the current phase, we have

$$\underline{\beta}(t) = (\beta_0(t), \cdots, \beta_{L+w-2}(t))^T = (I - A)^{-1} e_p, \tag{14.26}$$

where $A = [A_{i,j}]_{(L+w-1)(L+w-1)}$ has the form

$$A_{i,j} = \frac{1}{\ell_j(t)} \begin{cases} \sum_{k=i-w+1}^{i} 2c(t, k, \pi_{i-k,i-k}) & i = j, \\ \sum_{k=j-w+1}^{i} c(t, k, \pi_{i-k,j-k}) & 0 <\mid i - j \mid < w, \\ 0 & \text{otherwise} \end{cases} \tag{14.27}$$

Finally, the initial conditions are given by:

$$\begin{aligned} &\ell_i(0) = 2, \text{ for } 0 \le i \le L + w - 2 \\ &c(0, i, \underline{\tau}) = \begin{cases} \alpha \frac{\binom{K}{\tau_0, \tau_1, \cdots, \tau_{w-1}}}{w^K} & \text{if } \deg(\underline{\tau}) = K \text{ and } 0 \le i \le L - 1, \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{14.28}$$

Numerical Implementation

We have implemented the above set of differential equations in C. We define the threshold $\alpha_{\mathrm{UC},L,w}(K)$ as the highest density for which the spectral norm (largest eigenvalue) of the matrix A is strictly less than one throughout the whole algorithm. A practical point to notice here is that, for the sake of implementation, we assume a phase p finishes when its corresponding variable $\ell_p(t)$ goes below a (very) small threshold $\epsilon > 0$. In our implementations, we have typically taken $\epsilon = 10^{-5}$. However, it can be made arbitrarily small as long as the computational resources allow.

Table 14.1 shows the value of $\alpha_{\mathrm{UC},L,w}(K)$ with $L = 50$ and $w = 3$ for different choices of K. As we observe from Table 14.1, for the UC algorithm with the specific schedule mentioned above, there is a significant threshold improvement over the un-coupled ensemble.

For $L = 50, w = 3, K = 3$ and several values of α, we have plotted in Figure 14.10 the evolution of largest eigenvalue of A as a function of round time t.

In order to characterize analytically the ultimate threshold for the UC algorithm when L and w grow large, we proceed by further analyzing the set of differential equations.

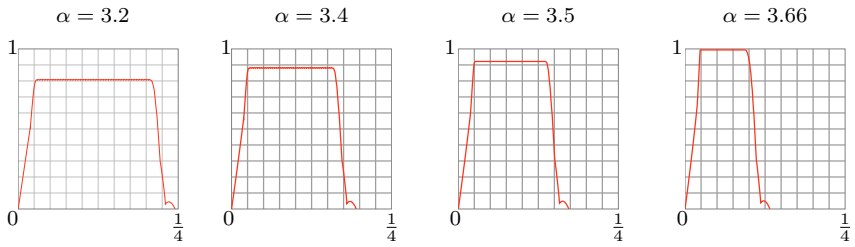


Figure 14.10 The largest eigenvalue of the matrix A, plotted versus the round time t (the number of rounds divided by the total number of variables NL). The plots correspond to an actual implementation of the UC algorithm for the 3-SAT coupled ensemble with $L = 50$ and $w = 3$. As we observe, for $\alpha < 3.67$, there is a gap between the largest eigenvalue of A and the value 1 throughout the UC algorithm. By increasing α this gap shrinks to 0. For $\alpha = 3.66$ (the right-most plot) this gap is around 0.006.

15 Spatial Coupling as a proof technique

Spatial coupling has another use besides engineering constructions for performant coding or compressed sensing designs. It can be used as a proof technique for deriving rigorous results on the phase diagrams of our problems. It is the goal of this chapter to expose the main principles of this idea.

16 Cavity Method: Basic Concepts

Message passing algorithms have been very successful in providing efficient algorithms in the realm of coding and compressive sensing. In the Bethe and associated replica symmetric variational formulas for the free energy enable us to calculate phase transition thresholds for these models. Finally, the Maxwell construction and spatial coupling technique tie together the algorithmic and variational approaches, and in particular the dynamical and phase transition thresholds. On the other hand these methods are not as successful for generic constraint satisfaction problems such as K-SAT. For example, plain belief propagation does not allow to find solutions of a K-SAT formula and had to be supplemented by a decimation process. Belief propagation guided decimation finds solutions up to some constraint density, but it is not clear if this limitation corresponds to some sort of fundamental dynamic threshold similar to the belief propagation threshold. Also, for the moment, we are not able to find the SAT-UNSAT threshold by a Maxwell construction or spatial coupling technique. At the same time we also saw that the replica symmetric entropy functional does not count correctly the number of solutions.

We already pointed out what could potentially go wrong in generic models with the plain Bethe and the subsequent replica symmetric approach. We argued in Sect 11.9, that if a Gibbs measure satisfies a "decoupling principle" then its (true) marginals satisfy the sum-product equations. This motivates the introduction of the Bethe free energy without any reference to a tree, since its stationnary points satisfy the sum-product equations. Furthermore this also suggests that the replica symmetric free energy should be exact. For sparse graphical models typical instances have no short loops so the decoupling principle translates to the absence of long range point-to-point correlations. For such models it is the absence of long range point-to-point correlations that is at the heart of the success of the replica symmetric approach. These arguments suggest that the failure of the replica symmetric formula for the free energy, e.g. in K-SAT when the constraint density increases, is related to the appearance of long range point-to-point correlations.

The subject of long range correlations is a very subtle one which we will be able to clarify only once we have developped the cavity method. For the moment we just mention that besides point-to-point correlations there is another type of long range correlation - called point-to-set correlation - which may exist even if

the replica symmetric free energy remains exact. Point-to-set correlations are not visible in the static properties such as the free energy, which remains analytic even in their presence, but they are responsible for a slowdown of the dynamics of algorithms.

In statistical mechanics of low dimensional Ising type models on regular grids there is a well developped mathematical theory of long range correlations and of their relation to the coexistence of extremal states (or measures). The Gibbs measure then has to be described by a convex superposition of these extremal states. Although it is very much of a challenge to develop some similar rigorous theory in the context of spin glass or constraint satisfaction models, analogous heuristic concepts form a conceptual framework at the basis of the cavity method. The cavity method boldly pushes the idea of convex decomposition of the Gibbs distribution to its limit in the sense that we will postualte a convex superposition with an arbitrary number (possibly exponentially large in n) of extremal states. Once this is accepted and extremal states identified, the theory, although technically challenging, more or less flows. Indeed it turns out this convex superposition defines a new factor graph model which can again be analyzed by the sum-product equations and Bethe free energy functional. That we can again revert to these techniques "one level up" is one of the fascinating aspects of the subject.

The approach taken in this chapter is not algorithmic. However, besides allowing us to understand the phase diagram of K-SAT, the cavity method suggests new efficient message passing algorithms discussed in Chapter **??**.

16.1 Coexistence of states

For concreteness we illustrate the notions of *extremal* or *pure* states and their coexistence in the framework of the best understood case, the Ising model introduced in Sect. 2.1. This is a very brief and informal overview of the subject which will suffice for our purposes.

Let us quickly summarize a few features that will be needed from the phase diagram of the model for dimensions greater or equal to two where phase transitions are present. The reader can refer back to Sect. 4.9 for more specific details. Consider the model on a cubic subset of a regular grid $\Lambda \subset \mathbb{Z}^d$, $d \geq 2$ with Hamiltonian

$$\mathcal{H}_\Lambda(\underline{s}) = -J \sum_{(i,j)\in E} s_i s_j - h \sum_{i\in V} s_i \tag{16.1}$$

where $J > 0$, $h \in \mathbb{R}$, E the set of edges in Λ, V the set of vertices in Λ. For convenience we center the cube Λ at the origin $o = (0, \cdots, 0) \in \mathbb{Z}^d$, and consider

the local magnetization

$$\langle s_i \rangle_\Lambda(\beta, h) = \frac{1}{Z_\Lambda} \sum_{\underline{s}} s_i e^{-\beta \mathcal{H}_\Lambda(\underline{s})} \tag{16.2}$$

at any *fixed* vertex $i \in \Lambda \subset \mathcal{Z}^d$. By fixed vertex we mean that its coordinates are fixed independent of the Λ, for example this vertex could be the origin itself $i = o$. We consider the thermodynamic limit of the local magnetization along any sequence of cubes centered at the origin $\Lambda \uparrow \mathbb{Z}^d$, and call this limit $m(\beta, h)$. It can be shown that this limit is independent of the chosen vertex as long as its coordinates are *fixed*. For any (β, h) not on the coexistence line, i.e. such that $\beta < \beta_c$ or $h \neq 0$, the limit $m(\beta, h)$ is an an analytic function. But on the coexistence line, i.e. $(\beta > \beta_c, h = 0)$, the thermodynamic limit is discontinuous $\lim_{h \to 0_\pm} m(\beta, h) = m_\pm(\beta)$ (with $m_+(\beta) \neq m_-(\beta)$). In particular, for $\beta < \beta_c$ the limit $h \to 0_\pm$ is unique and by the spin flip symmetry of the Hamiltonian $m(\beta, 0) = 0$.

This can be generalized to any Gibbs average $\langle s_X \rangle_\Lambda(\beta, h)$ where $s_X = \prod_{i \in X} s_i$ for any *fixed and bounded* subset $X \subset \Lambda$ (more generally one can consider any function of $\underline{s}$ with local support). Away from the coexistence line, $\beta < \beta_c$ or $h \neq 0$, $\lim_{\Lambda \to \mathbb{Z}^d} \langle s_X \rangle_\Lambda(\beta, h)$ is analytic and in particular

$$\lim_{h \to 0\pm} \lim_{\Lambda \to \mathbb{Z}^d} \langle s_X \rangle_\Lambda(\beta, h) \equiv \langle s_X \rangle \tag{16.3}$$

is unique. On the coexistence line the limit is in general not unique (for the Ising model due to spin flip symmetry both limits will be identical if X contains an even number of vertices)

$$\lim_{h \to 0\pm} \lim_{\Lambda \to \mathbb{Z}^d} \langle s_X \rangle_\Lambda(\beta, h) \equiv \langle s_X \rangle_\pm \tag{16.4}$$

The set of all limits (16.3) and (16.4) when X runs over all bounded subsets of $\mathbb{Z}^d$ esssentially defines infinite volume Gibbs states or measures for the zero field Ising Hamiltonian (in other words (16.1) with $h = 0$). Away from the coexistence line we obtain a "high temperature state" $\langle - \rangle$ with zero magnetization, and on the coexistence line we have constructed two "low temperature states" $\langle - \rangle_+$ and $\langle - \rangle_-$ with magnetizations $m_\pm(\beta)$. From the two states on the coexistence line one can form an infinite family of *mixed* Gibbs states by convex superpositions,

$$\langle - \rangle_w = w \langle - \rangle_+ + (1 - w) \langle - \rangle_-, \qquad 0 \leq w \leq 1 \tag{16.5}$$

One can prove that the low temperature states $\langle - \rangle_\pm$, as well as the high temperature state $\langle - \rangle$, are *extremal* or *pure* in the sense that they cannot be written as non-trivial convex superpositions (a convex superposition is non-trivial if the weights in the linear combination are not equal to zero or one). One can also prove that an extremal or pure state satisfies the *clustering* property. This property states that there is no long range correlation in the sense that for any two bounded distjoint sets X and Y,

$$\langle s_x s_Y \rangle_{\text{extr}} - \langle s_X \rangle_{\text{extr}} \langle s_Y \rangle_{\text{extr}} \to 0, \qquad \text{dist}(X, Y) \to +\infty \tag{16.6}$$

where $\langle - \rangle_{\text{extr}}$ stands for any extremal state (away from the critical point the decay is exponential in the distance between X and Y and becomes algebraic at the tip of the coexistence line). On the other hand a mixed state *cannot* satisfy the clustering property and displays point-to-point long range correlations. This is easy to see. Indeed using (16.5) and (16.6) one finds

$$\langle s_x s_Y \rangle_w - \langle s_X \rangle_w \langle s_Y \rangle_w \to w(1-w)(\langle s_X \rangle_+ - \langle s_X \rangle_-)(\langle s_Y \rangle_+ - \langle s_Y \rangle_-) \quad (16.7)$$

as $\text{dist}(X, Y) \to +\infty$. The prototypical case is $X = i$ and $Y = j$ with $|i - j| \to +\infty$, where one finds $\langle s_i s_j \rangle_w - \langle s_i \rangle_w \langle s_j \rangle_w \to 4w(1-w)m_+(\beta)^2$. In a non trivial mixed state we necessarily have "point-to-point" long range corelations.

It is noteworthy that when one can show or argue that long range point to point correlations are present then necessarily there must exist more than one extremal state. One can then ask among all possible non-trivial convex superposition is there one which is more natural than others? This is not a very well defined question and depends on the problem at hand. For example for the Ising model if we want to describe an infinite volume state which is invariant under translations and a spin flip transformation then it is natural to choose the convex combination with weights $w = 1 - w = 1/2$.

Even when one knows that more than one extremal state exists, one should not have any prejudice on their number. First of all there is in general more than one way to construct an extremal state. For the zero field Ising model for example we described above two low temperature states by using an external symmetry breaking infinitesimal magnetic field which then tends to $0_\pm$. But one can construct the same states by letting $h = 0$ and breaking the symmetry thanks to boundary conditions by fixing the spin assignement on the boundary of Λ to all pluses or all minuses. It turns out that this yields the same states as above $\langle - \rangle_\pm$. But clearly one can imagine other boundary conditions or infinitesimal inhomogeneous magnetic fields. For the two-dimensional Ising model it is proven that the picture described above is complete. Away from the coexistence line the infinite volume Gibbs state is unique while on the coexistence line (excluding the critical point) there are only two extremal states. Any other infinite volume Gibbs state is a convex combination of these two extremal states. But the situation is richer in higher dimensions $d \geq 3$ where other extremal states corresponding to interfaces between a positive and negative magnetization regions can be constructed.

16.2 Convex superposition ansatz for models on sparse graphs

We now turn to generic spin glass models on sparse graphs keeping in mind our main application which will be the K-SAT problem. Our discussion applies to a

Gibbs distribution of the form (11.1)

$$\mu\left(\underline{x}\right)=\frac{1}{Z}\prod_{a}f_{a}\left(x_{\partial a}\right) \tag{16.8}$$

where the underlying factor graph is sparse and locally tree like. The measure is random in the sense that it belongs to an ensemble (e.g. the usual K-SAT ensemble) but for teh moment we consider a fixed instance. We pointed out in the introduction to this chapter that a failurem of the replica symmetric formula suggests the presence of long range point-to-point corelations are present in typical instances. This is the case for K-SAT above a certain constraint density as shown in Sect. 12.5. Our experience with low dimensional statistical mechanics models on regular grids, such as the Ising model, then suggests that many "extremal" or "pure" "states" must coexist. Unlike the Ising model we do have a good mathematical theory of such notions for generic spin glass or constraint satisfaction models. Nevertheless we will assume some sort of analogous notions exist and in particular we will assume that in "extremal states" (those that cannot be written as a non trivial convex superposition) the point-to-point correlations decay "fast enough". Let $\langle-\rangle_{\alpha}$, $\alpha=1,\ldots,\mathcal{N}$ be the set of all "extremal states". Because it is not clear how to select one such state we attempt to capture the infinite system Gibbs measure when none of them is selected in particular. One must then represent (16.8) as a convex superposition

$$\mu(\cdot)=\sum_{\alpha=1}^{\mathcal{N}}w_{\alpha}\langle-\rangle_{\alpha}. \tag{16.9}$$

We now have to discuss two issues. How can we concretely represent or approximate $\langle-\rangle_{\alpha}$? What are the natural weights w_{α} that we should take?

We pointed out that on a sparse graph when point-to-point correlations decay the sum-product equations are satisfied by the (true) marginals of (16.8). Therefore each "extremal state" $\langle-\rangle_{\alpha}$ has marginals that satisfy the sum-product equations. We will therefore identify these states with messages $\{\mu_{i\to a}^{\alpha},\mu_{a\to i}^{\alpha}\}$, $\alpha=1,\ldots,\mathcal{N}$ that are solutions of

$$\begin{cases}\mu_{i\to a}^{\alpha}(x_i)=\frac{\prod_{b\in\partial i\setminus a}\hat{\mu}_{b\to i}^{\alpha}(x_i)}{\sum_{x_i}\prod_{b\in\partial i\setminus a}\hat{\mu}_{b\to i}^{\alpha}(x_i)}\\ \\ \hat{\mu}_{a\to i}^{\alpha}(x_i)=\frac{\sum_{\sim x_i}f_a(x_{\partial a})\prod_{j\in\partial a\setminus i}\mu_{j\to a}^{\alpha}(x_i)}{\sum_{x_{\partial a}}f_a(x_{\partial a})\prod_{j\in\partial a\setminus i}\mu_{j\to a}^{\alpha}(x_i)}\end{cases} \tag{16.10}$$

Here we enforce the normalization in the sum-product equations in order not to overcount messages that are equivalent up to a normalization factor. Moreover we expect that such states should be stable with respect to small perturbations so we only consider solutions that correspond to *local minima* of the Bethe free energy functional (11.14). The Bethe free energy of the state $\langle-\rangle_{\alpha}$ is the value of the functional at such a minimum of state $\langle-\rangle_{\alpha}$

$$F_{\alpha}\equiv F_{\text{Bethe}}\left[\underline{\mu}^{\alpha},\underline{\hat{\mu}}^{\alpha}\right]. \tag{16.11}$$

States which have marginals satisfying the sum-product equations and which are minima of the bethe free energy functional play the role of extremal states and will be henceforth called *Bethe extremal states*.

Let us now turn to the choice of the weights w_α. Consider first the K-SAT problem at zero temperature. Then $\mu(\underline{x})$ is nothing else than the uniform measure over solutions assuming we are in a SAT phase where solutions exist. Clearly in that case we should take $w_\alpha = \Omega_\alpha / \sum_{\alpha=1}^{\mathcal{N}} \Omega_\alpha$, Ω_α the number of solutions in the zero temperature limit of the state $\langle - \rangle_\alpha$. The natural finite temperature generalization of this weight is

$$w_\alpha = \frac{e^{-\beta F_\alpha}}{\sum_{\alpha=1}^{\mathcal{N}} e^{-\beta F_\alpha}} \tag{16.12}$$

This can be checked heuristically from the thermodynamic relation $F_\alpha = U_\alpha - \beta^{-1} S_\alpha$ where U_α is the internal energy of the state and S_α its entropy. In a SAT phase the ground state energy vanishes so $\lim_{\beta\to+\infty} \beta F_\alpha = -\lim_{\beta\to+\infty} S_\alpha = -\ln \Omega_\alpha$ and (16.12) reduces to $w_\alpha = \Omega_\alpha / \sum_{\alpha=1}^{\mathcal{N}} \Omega_\alpha$.

To sumarize we arrive at the following ansatz for the convex representation of (16.8)

$$\mu(\cdot) = \frac{\sum_{\alpha=1}^{\mathcal{N}} e^{-\beta F_\alpha} \langle - \rangle_\alpha}{\sum_{\alpha=1}^{\mathcal{N}} e^{-\beta F_\alpha}} \tag{16.13}$$

The sum over α is over extremal Bethe states. In practice these are identified with messages $\{\mu^\alpha_{i\to a}, \mu^\alpha_{a\to i}\}$, $\alpha = 1, \ldots, \mathcal{N}$ satisfying sum-product equations and which are local minima of the Bethe functional.

The most informative quantity which we will compute and allows to determine the phase diagram and various thresholds is the number of extremal Bethe states that *effectively contribute* to the sums over α in the numerator and denominator of (16.13). In the next section we set up the necessary formalism to determine the number of relevant extremal Bethe states.

16.3 Counting states: level-one model and complexity

In generic constraint satisfaction problems or spin glass models it is expected that at low temperatures there exist numerous extremal states corresponding to minima of the Bethe free energy functional. It is useful to have in mind as a mental picture to think about the Bethe functional as an "energy landscape" in the space of messages $\{\mu_{a\to i}, \mu_{i\to a}\} \in \mathbb{R}^{4|E|}$, depicted on figure 16.1. The energy landscape contains many critical points (minima, maxima, saddles) that are solutions of the sum-product equations, but at low temperatures essentially only the low lying minima are those that contribute to the convex superposition. The low lying minima are the relevant Bethe extremal states that we must count.

— figure —

Figure 16.1 Cartoon of the effective energy landscape: a low temperature Bethe free energy functional in a phase with many solutions to the sum-product equations

As we will see the number of relevant extremal Bethe states can be exponentially large. In such situations their exponential growth rate can be computed using a thermodynamic formalism. We introduce a partition function

$$Z_1(x) = \sum_{\alpha} e^{-\beta x F_\alpha}. \tag{16.14}$$

of an auxiliary system whose microscopic degrees of freedoms are extremal Bethe states $\alpha \equiv (\underline{\mu}^\alpha, \hat{\underline{\mu}}^\alpha)$ and energy function (or Hamiltonian) is $F_\alpha = F_{\text{Bethe}}[\underline{\mu}^\alpha, \hat{\underline{\mu}}^\alpha]$. Here x is a real parameter whose use will soon become clear. For $x = 1$ (16.14) is precisely the denominator of (16.13), so $x = 1$ is the natural value we should use if we want to determine the number of extremal states that effectively contribute to the Gibbs measure. But as we will see this choice for x breaks down when there is a transition from an exponentially large number of extremal states to a finite number of them. We will refer to the auxiliary model introduced here as the *level-one model*.

In principle the sum in (16.14) should carry over extremal states or minima of the energy landscape. However this constraint is difficult to implement in practice and we will sum over all critical points of the energy landscape, i.e. all solutions of the sum-product equations. At low temperatures we do not expect that this makes a significant difference because in effect only minima effectively contribute. We will refer to the auxiliary model introduced here as the *level-one model*.

We now discuss how the level-one model allows in principle to count extremal Bethe states. For now, we assume that we are in a regime where the number of extremal Bethe states are exponentially numerous as a function of the system size. The number of minima of the free energy landscape with free energies in the infinitesimal interval $[f - df, f]$ is equal to $e^{n\Sigma(f)}$. The growth rate $\Sigma(f)$ is called the *complexity*, and can be interpreted as a "Boltzman entropy" for the

level-one model. As $n \to +\infty$ we approximate the sum as an integral

$$Z_1(x) \approx \int df\, e^{n(\Sigma(f)-\beta x f)}. \tag{16.15}$$

and the free energy of the level one auxiliary model $\beta x f_1(x) = -\lim_{n\to+\infty} \ln Z_1(x)$ becomes

$$\beta x f_1(x) = \min_f(\beta x f - \Sigma(f)) \tag{16.16}$$

This is the usual thermodynamic relation stating that the free energy of the level-one model is the Legendre transform of the entropy where the free energy f is traded for the "temperature" x. Assuming that $\Sigma(f)$ is concave there is a unique minimum $f_*(x)$ solving $\beta x = \partial\Sigma(f)/\partial f$ and we find $\Sigma(f_*(x)) = \beta x(f_*(x) - x f_1(x))$. There is also a more direct way to compute the later function of x, namely using the relation $\Sigma(x) = \partial f_1(x)/\partial x^{-1}$. Now, setting $x = 1$ we find that there are $e^{n\Sigma(x=1)}$ extremal Bethe states contributing to the convex superposition (16.13) with free energies $f_*(x = 1))$ obtained from the equations,

$$\Sigma(x=1) = \left.\frac{\partial f_1(x)}{\partial x^{-1}}\right|_{x=1}, \qquad f_*(x=1) = \beta^{-1}\Sigma(f_*(x=1)) + f_1(x=1). \tag{16.17}$$

These formulas break down when the extremal Bethe states are not exponentially numerous. In practice, when this is the case, one finds $\Sigma(x = 1) < 0$ which is not an acceptable solution. In a such a regime the correct prescription is to enforce the condition that the complexity should vanish by taking the value $0 < x_* < 1$ closest to $x = 1$ such that

$$\Sigma(x_*) = \left.\frac{\partial f_1(x)}{\partial x^{-1}}\right|_{x=x_*} = 0, \qquad f_*(x_*) = f_1(x_*). \tag{16.18}$$

In this situation there are a finite number of extremal Bethe states contributing to (16.13)) with free energies $f_1(x_*)$.

Typical predictions for the complexity

Before embarking in the substantially technical development of the cavity method in the next sections this is a good point to illustrate the typical predictions of this formalism for concrete models. We take here for concreteness the example of K-SAT for $K \geq 4$ which is has the most generic behaviour and discuss the complexity as function of the constraint density α.

We will see that the calculation of the complexity shows the existence of two thresholds for the constraint density: the *dynamical* threshold $\alpha_d(K)$ and the *condensation* threshold $\alpha_c(K)$ (see figure 16.2 for $K = 4$). For $\alpha < \alpha_d(K)$ we find a zero complexity $\Sigma(x = 1) = 0$. In this regime there a single extremal Bethe state and the free energy is equal to the replica symmetric prediction. At $\alpha_d(K)$ the complexity jumps to a finite positive value, remains positive in

— figure —

Figure 16.2 The complexity of 4-SAT as a function of the constraint density α.

an interval $\alpha_d(K) < \alpha < \alpha_c(K)$, vanishes at $\alpha_c(K)$ and then becomes negative $\Sigma(x = 1) < 0$. As we will see the free energy is still given by the replica symmetric formula all the way up to $\alpha_c(K)$ and thus remains analytic. The dynamical threshold $\alpha_d(K)$ does not correspond to a static phase transition but rather it affects the dynamical properties of the model (e.g. the mixing time of sampling algorithms such as MCMC) due to the proliferation of extremal states. At $\alpha_c(K)$ there is a non-analyticity of the free energy and replica symmetric formula breaks down. This is a static phase transition called *condensation transition* because the number of extremal Bethe states that contribute to the Gibbs measure changes from exponentially many to a finite number. We will also discover that for $\alpha_d(K) < \alpha < \alpha_c(K)$ the convex superposition has marginals that still satisfy the sum-product equations. At the same there are no long range point-to-point correlations in this regime (and also for $\alpha < \alpha_d(K)$ since there is only one state). Above $\alpha_c(K)$ there are long range point-to-point correlations and the marginals of the Gibbs measure do not satisfy the sum-product equations.

16.4 Level-one model as a factor graph model

In order to make technical progress we have to compute the free energy $f_1(x)$ of the level-one model. From there on, one can deduce the complexity and the various thresholds, as well as the total free energy and internal free energies of extremal states. At first solving the level one model may seem quite challenging. Indeed in the partition function (16.14) the microscopic degrees of freedom $\{\mu^{\alpha}_{i\to a}, \hat{\mu}^{\alpha}_{a\to i}\}$ as well as the weights F_α are already complicated objects. Moreover we must take into account the the sum-product constraints (16.10)) in the sum over microscopic degrees of freedom. The key point allowing such calculations possible here is that the level-one model is in effect just another factor graph model. Once this is recognized we can essentially revert to all our usual sum-product and Bethe formalism.

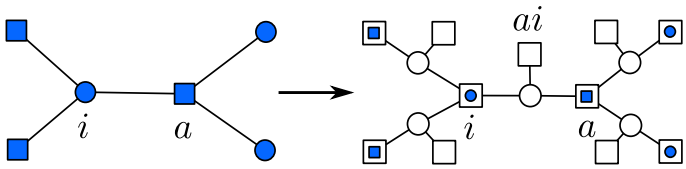


Figure 16.3 On the left, an exemple of an original graph Γ. On the right its corresponding graph Γ_1 for the level-one model.

We first rewrite (16.14) more explicitly as

$$Z_1(x) = \sum_{\underline{\mu},\underline{\hat{\mu}}} e^{-xF_{\text{Bethe}}\,(\underline{\mu},\underline{\hat{\mu}})}\, \mathbb{1}_{\text{sp}}(\underline{\mu},\underline{\hat{\mu}}) \tag{16.19}$$

where the "indicator function" $\Delta_{\text{sp}}(\underline{\mu},\underline{\hat{\mu}})$ selects solutions of the sum product equations (16.10). Note that in these equations we normalized the messages so as not to overcount equivalent solutions. We explicitly represent this indicator function as

$$\mathbb{1}_{\text{sp}}(\underline{\mu},\underline{\hat{\mu}}) = \left\{\prod_{i=1}^{n}\prod_{a\in\partial i}\Delta_{i\to a}\right\}\left\{\prod_{a=1}^{m}\prod_{i\in\partial a}\hat{\Delta}_{a\to i}\right\} \tag{16.20}$$

with

$$\begin{cases} \Delta_{i\to a} = \mathbb{1}\left(\mu_{i\to a}(\cdot) = \dfrac{\prod_{b\in\partial i\setminus a}\hat{\mu}_{b\to i}(\cdot)}{\sum_{x_i}\prod_{b\in\partial i\setminus a}\hat{\mu}_{b\to i}(x_i)}\right) \\ \hat{\Delta}_{a\to i} = \mathbb{1}\left(\hat{\mu}_{a\to i}(\cdot) = \dfrac{\sum_{\sim x_i} f_a(\cdot,x_{\partial a_1})\prod_{j\in\partial a\setminus i}\mu_{j\to a}(x_j)}{\sum_{x_{\partial a}} f_a(x_{\partial a})\prod_{j\in\partial a\setminus i}\mu_{j\to a}(x_i)}\right) \end{cases} \tag{16.21}$$

Recall that the Bethe free energy (11.14) has three contributions from variable nodes, function nodes and edges. Thus each term in the sum (16.19) is

$$\left\{\prod_{i=1}^{n} e^{-xF_i}\prod_{a\in\partial i}\Delta_{i\to a}\right\}\left\{\prod_{a=1}^{m} e^{-xF_a}\prod_{i\in\partial a}\hat{\Delta}_{a\to i}\right\}\left\{\prod_{\text{edges}ia} e^{+xF_{ia}}\right\} \tag{16.22}$$

This last expression, when normalized by $Z_1(x)$, is the explicit form of the Gibbs weight of the level-one model.

If $\Gamma = (V, C, E)$ is the original factor graph, then the level-one model has the factor graph $\Gamma_1 = (V_1, C_1, E_1)$ depicted on Fig. 16.3. A variable node $i \in V$ of the original graph becomes a function node $i \in C_1$ in the new graph, with factor

$$\psi_i = e^{-xF_i}\prod_{a\in\partial i}\Delta_{i\to a}. \tag{16.23}$$

A function node $a \in C$ in the original graph becomes a function node $a \in C_1$ in the new graph, with factor

$$\psi_a = e^{-xF_a}\prod_{i\in\partial a}\hat{\Delta}_{a\to i}. \tag{16.24}$$

An edge $(a, i) \in E$ in the original graph becomes a variable node $(a, i) \in V_1$ in

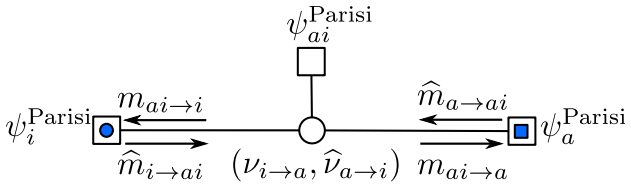


Figure 16.4 We use the letter m for messages outgoing from a "level-one variable node" and $\widehat{m}$ for messgaes outgoing from a "level-one factor node".

the new graph. There is also an extra function node attached to each variable node of the new graph, or equivalently attached to each edge of the old graph. The corresponding factor is

$$\psi_{ai} = e^{+xF_{ai}}. \tag{16.25}$$

With these definitions the Gibbs weight of the level-one model has the structure of a factor graph model (with three sort of factors),

$$\mu_1(\underline{\mu}, \underline{\widehat{\mu}}) = \frac{1}{Z_1(x)} \prod_{i\in V} \psi_i \prod_{a\in C} \psi_a \prod_{ai\in E} \psi_{ai}. \tag{16.26}$$

16.5 Message passing solution of the level-one model

Level-one sum-product equations

Our first task is to compute the marginals of (16.26). These are distributions over $\{\mu_{i\to a}, \hat{\mu}_{i\to a}\}$; so they are distributions of distributions. If the level-one factor graph was a tree (this would be the case if the original factor graph was a tree) the sum-product equations would give the exact marginals. This is not the case but can still be fruitful (e.g. for K-SAT) to use the sum-product equations, much as this was a successful idea in coding and compressive sensing.

The sum product equations for (16.26) involve four kind of messages shown on figure 16.4. Messages flowing from a "level one function node" to a "level-one variable node" satisfy

$$\begin{aligned}\widehat{m}_{a\to ai} &\propto \sum_{\sim(\mu_{i\to a}, \widehat{\mu}_{a\to i})} \psi_a \prod_{aj\in\partial a\setminus ai} m_{aj\to a} \\ &= \sum_{\sim(\mu_{i\to a}, \widehat{\mu}_{a\to i})} \hat{\Delta}_{a\to i} e^{-xF_a} \prod_{aj\in\partial a\setminus ai} m_{aj\to a}\end{aligned}$$

and

$$\begin{aligned}\widehat{m}_{i\to ai} &\propto \sum_{\sim(\mu_{i\to a}, \widehat{\mu}_{a\to i})} \psi_i \prod_{bi\in\partial i\setminus ai} \widehat{m}_{bi\to i} \\ &= \sum_{\sim(\mu_{i\to a}, \widehat{\mu}_{a\to i})} \Delta_{i\to a} e^{-xF_i} \prod_{bi\in\partial i\setminus ai} \widehat{m}_{bi\to i}\end{aligned}$$

Messages from a "level-one variable node" to a "level-one function node" satisfy

$$m_{ai\to i} = e^{xF_{ai}}\widehat{m}_{a\to ai}, \qquad m_{ai\to a} = e^{xF_{ai}}\widehat{m}_{i\to ai}.$$

Notice that $m_{ai\to i}$ is independent of $\widehat{\mu}_{a\to i}$ and $m_{ai\to a}$ is independent of $\mu_{i\to a}$. This allows to simplify the four message passing equations. To achieve this simplification define two distributions (of distributions)

$$Q_{i\to a}(\mu_{i\to a})m_{ai\to a}, \qquad \widehat{Q}_{a\to i}(\widehat{\mu}_{a\to i}) = m_{ai\to i} \tag{16.27}$$

which flow on the edges of the original factor graph $\Gamma = (V, C, E)$. It is easy to see that the four message passing equations above reduce to

$$\begin{cases} \widehat{Q}_{a\to i}(\widehat{\mu}_{a\to i}) \propto \sum_{\underline{\mu}} \hat{\Delta}_{a\to i} e^{-x(F_a - F_{ai})} \prod_{j\in\partial a\setminus i} Q_{j\to a}(\mu_{j\to a}) \\ Q_{i\to a}(\mu_{i\to a}) \propto \sum_{\underline{\hat{\mu}}} \Delta_{i\to a} e^{-x(F_i - F_{ai})} \prod_{b\in\partial i\setminus a} \widehat{Q}_{b\to i}(\widehat{\mu}_{b\to i}). \end{cases} \tag{16.28}$$

In this form the level-one sum-product equations are often called *cavity equations* and the distributions (16.27) they connect *cavity messages*. Note that cavity equations (16.28) do not make any reference to the level-one graph Γ_1 and we can revert to the original one.

The x dependent exponentials in (16.28) are sometimes called reweighting factors. Their explicit expressions (obtained from (11.14)) will be useful later on,

$$e^{-(F_i - F_{ai})} = \sum_{x_i} \prod_{b\in\partial i\setminus a} \widehat{\mu}_{b\to i}(x_i), \qquad e^{-(F_a - F_{ai})} = \sum_{x_{\partial a}} f_a(x_{\partial a}) \prod_{\partial j\in a\setminus i} \mu_{j\to a}(x_i) \tag{16.29}$$

Note that these are in fact the normalization factors in (16.10).

Level-one Bethe free energy

The Bethe free energy functional of the level-one model is a functional of the cavity messages (16.27). We could derive it as in Chapter 11 by first deriving the exact free energy $f_1(x)$ on a tree, and then take this expression as a defition for general graph instances. Alternatively we could write down a functional of the cavity messages such that its stationary points satisfy the cavity equations (16.28). But we can also guess the fomula. It is basically given by the usual definition, but with the extra feature that it must contain the degrees of freedom $(\underline{\mu}, \underline{\hat{\mu}})$ must be correctly weighted. This is enough information to guess (and check a posteriori) that the Bethe free energy functional is

$$\mathcal{F}_{\text{Bethe}}(\underline{Q}, \underline{\widehat{Q}}) = \sum_{i\in V} \mathcal{F}_i + \sum_{a\in C} \mathcal{F}_a - \sum_{ai\in E} \mathcal{F}_{ai} \tag{16.30}$$

where

$$\mathcal{F}_i(\{\widehat{Q}_{b\to i}\}_{b\in\partial i}) = -\frac{1}{x}\ln\Big\{\sum_{\underline{\hat{\mu}}} e^{-xF_i}\prod_{b\in\partial i}\widehat{Q}_{b\to i}\Big\}, \tag{16.31}$$

$$\mathcal{F}_a(\{Q_{j\to a}\}_{j\in\partial a}) = -\frac{1}{x}\ln\Big\{\sum_{\underline{\mu}} e^{-xF_a}\prod_{j\in\partial a} Q_{j\to a}\Big\}, \tag{16.32}$$

$$\mathcal{F}_{ai}(Q_{i\to a},\widehat{Q}_{a\to i}) = -\frac{1}{x}\ln\Big\{\sum_{\mu,\hat{\mu}} e^{-xF_{ai}}Q_{i\to a}\widehat{Q}_{a\to i}\Big\}. \tag{16.33}$$

We will see when we apply the formalism to K-SAT that an "averaged" form of this expression yields the so called "one-step replica symmetry broken" formula for the free energy. This allows to calculate the free energy in regimes where the replica symmetric formula fails.

Complexity functional

We saw in Sect. 16.3 that the complexity is given by $\Sigma(x) = \partial f_1(x)/\partial x^{-1}$. Within the Bethe formalism we have $\Sigma_{\rm Bethe} = \frac{\partial}{\partial x^{-1}}\mathcal{F}_{\rm Bethe}$. One finds

$$\Sigma_{\rm Bethe}(\underline{Q},\underline{\widehat{Q}}) = \sum_{i\in V}\Sigma_i + \sum_{a\in C}\Sigma_a - \sum_{ai\in E}\Sigma_{ai} \tag{16.34}$$

where

$$\Sigma_i(\{\widehat{Q}_{b\to i}\}_{b\in\partial i}) = -x\mathcal{F}_i + x\frac{\sum_{\underline{\hat{\mu}}} F_i e^{-xF_i}\prod_{b\in\partial i}\widehat{Q}_{b\to i}}{\sum_{\underline{\hat{\mu}}} e^{-xF_i}\prod_{b\in\partial i}\widehat{Q}_{b\to i}}, \tag{16.35}$$

$$\Sigma_a(\{Q_{j\to a}\}_{j\in\partial a}) = -x\mathcal{F}_a + x\frac{\sum_{\underline{\mu}} F_a e^{-xF_a}\prod_{j\in\partial a} Q_{j\to a}}{\sum_{\underline{\mu}} e^{-xF_a}\prod_{j\in\partial a} Q_{j\to a}}, \tag{16.36}$$

$$\Sigma_{ai}(Q_{i\to a},\widehat{Q}_{a\to i}) = -x\mathcal{F}_{ai} + x\frac{\sum_{\mu,\hat{\mu}} F_{ai}e^{-xF_{ai}}Q_{i\to a}\widehat{Q}_{a\to i}}{\sum_{\mu,\hat{\mu}} e^{-xF_{ai}}Q_{i\to a}\widehat{Q}_{a\to i}}. \tag{16.37}$$

Comparing (16.31)-(16.33) and (16.35)-(16.37) we see that

$$\mathcal{F}_{\rm Bethe} = \langle F_{\rm Bethe}\rangle_{\rm cav} - x^{-1}\Sigma_{\rm Bethe}. \tag{16.38}$$

A bit of thought shows that the bracket $\langle -\rangle_{\rm cav}$ is the Gibbs average of the level-one model calculated from the cavity messages. Relationship (16.38) was expected on thermodynamic grounds but it is reassuring that it can be derived explicitly in the present framework.

In the application to K-SAT we use an "averaged" form of the complexity functional to calculate the dynamical and condensation thresholds.

16.6 Simplifications for $x = 1$

We argued in Sect. 16.3 as long as the complexity is non-negative we should set $x = 1$ in the level-one model. It is fortunate that in this case a large portion of the formalism above can be simplified by eliminating entirely the need for reweighting factors. This makes the computation of the average complexity and thresholds much simpler both theoretically and numerically.

Free energy for $x = 1$

Let us first discuss the level-one Bethe free energy. Replacing (11.15)-(11.17) in (16.31)-(16.33) we find

$$\mathcal{F}_{\rm Bethe}(\underline{Q},\underline{\widehat{Q}})\big|_{x=1} = F_{\rm Bethe}(\underline{\mu}^{\rm av},\underline{\hat{\mu}}^{\rm av}) \tag{16.39}$$

where the right hand side is the usual Bethe free energy expressed in terms of *average messages*,

$$\mu^{\rm av}_{i\to a}(x_i) = \sum_{\mu_{i\to a}} \mu_{i\to a}(x_i) Q_{i\to a}(\mu_{i\to a}), \quad \hat{\mu}^{\rm av}_{a\to i}(x_i) = \sum_{\hat{\mu}_{a\to i}} \hat{\mu}_{a\to i}(x_i)\hat{Q}_{a\to i}(\hat{\mu}_{a\to i}). \tag{16.40}$$

It turn out the average messages satisfy the usual sum-product equations,

$$\begin{cases} \mu^{\rm av}_{i\to a}(x_i) = \prod_{b\in\partial i\setminus a} \hat{\mu}^{\rm av}_{b\to i}(x_i), \\ \hat{\mu}^{\rm av}_{i\to a}(x_i) = \sum_{x_{\partial a}} f_a(x_{\partial a}) \prod_{j\in\partial a\setminus i} \mu^{\rm av}_{j\to a}(x_j). \end{cases} \tag{16.41}$$

To prove (16.41) we notice that (16.39) and (16.40) imply ($\delta_R G$ is an infinitesimal variation of G with respect to R)

$$\delta_{Q_{i\to a}} \mathcal{F}_{\rm Bethe}\big|_{x=1} = (\delta_{\mu^{\rm av}_{i\to a}} F_{\rm Bethe})\mu_{i\to a}(x_i)$$

$$\delta_{\hat{Q}_{i\to a}} \mathcal{F}_{\rm Bethe}\big|_{x=1} = (\delta_{\hat{\mu}^{\rm av}_{i\to a}} F_{\rm Bethe})\hat{\mu}_{i\to a}(x_i),$$

therefore if $(\underline{Q},\underline{\hat{Q}})$ is a stationary point of $\mathcal{F}_{\rm Bethe}|_{x=1}$ then $(\mu^{\rm av},\hat{\mu}^{\rm av})$ is a stationary point of $F_{\rm Bethe}$. Thus the cavity equations for $(\underline{Q},\underline{\hat{Q}})$ imply the sum-product equations for $(\mu^{\rm av},\hat{\mu}^{\rm av})$. This conclusion can also be reached by a direct calculation starting from the cavity equations for $x = 1$.

Equations (16.39) and (16.41) are quite remarquable. They suggest that as long as the choice $x = 1$ is valid the average free energy is given by the replica symmetric formula. In K-SAT for example we will see that this is the case for $\alpha < \alpha_d$ where $\Sigma(x = 1) = 0$ and also for $\alpha_d < \alpha < \alpha_c$ where $\Sigma(x = 1) > 0$. In particular there is no static phase transition in the whole regime where $x = 1$, i.e for $\alpha < \alpha_c$, and to access the free energy we do not have to solve the full cavity equations. Of course to determine α_d and α_c we must compute the complexity but for this quantity also we will shotly see that the cavity equations can be simplified.

Conceptually $\mu^{\rm av}_{i\to a}$ and $\hat\mu^{\rm av}_{i\to a}$ are very natural messages to consider. Suppose for the sake of the argument that $Q(\mu_{i\to a})$ and $\hat Q(\hat\mu_{i\to a})$ are the true marginals of the level-one model. Then the average messages are the Gibbs averages of the dynamical variables of the level-one model, much like the magnetization is the Gibbs average of the spin variable. In other words if we sample among the set of solutions of the sum-product equations (16.10) according to the weight $e^{-F_{\rm Bethe}}/Z_1(x=1)$ these are the expected messages that we get. From these expected messages one can derive marginals of the convex superposition (16.13). It is quite remarkable that the average messages satisfy the usual sum product equations. But one must bear in mind that they do not describe extremal Bethe states but their convex superposition. To summarize, one must bear in mind that even when belief propagation correctly computes marginals of a measure $\mu(\underline{x})$ (e.g. (**??**)) this does not ncessarily mean that we are able to access the marginals of pure states. This is the case only when there is a single pure state. This has an important algorithmic consequence for example for finding solutions of K-SAT. For $\alpha<\alpha_d$ there is a single extremal Bethe state and belief propagation marginals give us useful information for finding solutions (e.g. when decimation is used). However for $\alpha_d<\alpha<\alpha_c$ the belief propagation marginals do not correctly represent the marginals of the extremal states and cannot be used to find solutions. We come back to these matters in Chapter **??**.

Complexity for $x=1$

We now turn to the Bethe complexity for $x=1$. According to (16.38) and (16.39)

$$\Sigma_{\rm Bethe}\big|_{x=1}=\langle\mathcal{F}_{\rm Bethe}\rangle_{\rm cav}\big|_{x=1}-F_{\rm Bethe}(\underline{\mu}^{\rm av},\underline{\hat\mu}^{\rm av}) \tag{16.42}$$

The second term is computed by solving usual sum-product equations instead of the complete cavity equations, but the first term in the present form still requires to compute complicated averages $\langle F_i\rangle_{\rm cav}$, $\langle F_a\rangle_{\rm cav}$, $\langle F_{ai}\rangle_{\rm cav}$ read off from (16.35)-(16.37).

Let us start with $\langle F_i\rangle_{\rm cav}$. Replacing (11.15) in this average and setting $x=1$ we find

$$\begin{aligned}\langle F_i\rangle_{\rm cav}\big|_{x=1}&=\frac{\sum_{\underline{\hat\mu}}\ln\Big\{\sum_{x_i}\prod_{b\in\partial i}\hat\mu_{b\to i}(x_i)\Big\}\sum_{x_i}\prod_{b\in\partial i}\hat\mu_{b\to i}(x_i)\hat Q_{b\to i}}{\sum_{\underline{\hat\mu}}\sum_{x_i}\prod_{b\in\partial i}\hat\mu_{b\to i}(x_i)\hat Q_{b\to i}}\\&=\frac{\sum_{\underline{\hat\mu}}\ln\Big\{\sum_{x_i}\prod_{b\in\partial i}\hat\mu_{b\to i}(x_i)\Big\}\sum_{x_i}\prod_{b\in\partial i}\hat\mu_{b\to i}(x_i)\hat Q_{b\to i}}{\sum_{x_i}\prod_{b\in\partial i}\hat\mu^{\rm av}_{b\to i}(x_i)}\\&=\sum_{\underline{\hat\mu}}\ln\Big\{\sum_{x_i}\prod_{b\in\partial i}\hat\mu_{b\to i}(x_i)\Big\}\sum_{x_i}\nu_i^{\rm av}(x_i)\prod_{b\in\partial i}\hat R_{b\to i}(\hat\mu_{b\to i}|x_i)\end{aligned} \tag{16.43}$$

where in the last line the weighting factor over messages is expressed thanks to

the average marginal

$$\nu_i^{\text{av}}(x_i) = \frac{\prod_{b\in\partial i} \hat{\mu}_{b\to i}^{\text{av}}(x_i)}{\sum_{x_i}\prod_{b\in\partial i} \hat{\mu}_{b\to i}^{\text{av}}(x_i)} \tag{16.44}$$

and the conditional distribution over messages

$$\hat{R}_{b\to i}(\hat{\mu}_{b\to i}|x_i) = \frac{\hat{\mu}_{b\to i}(x_i)\hat{Q}_{b\to i}}{\hat{\mu}_{b\to i}^{\text{av}}(x_i)} \tag{16.45}$$

Now we turn to $\langle F_a\rangle_{\text{cav}}$. Replacing (11.16) in this average and setting $x = 1$ we find

$$\begin{aligned}
\langle F_a\rangle_{\text{cav}}\big|_{x=1} &= \frac{\sum_{\underline{\mu}} \ln\Big\{\sum_{x_{\partial a}}\prod_{i\in\partial a}\mu_{i\to a}(x_i)\Big\}\sum_{x_{\partial a}} f_a(x_{\partial a})\prod_{i\in\partial a}\mu_{i\to a}(x_i)Q_{i\to a}}{\sum_{\underline{\mu}}\sum_{x_{\partial a}} f_a(x_{\partial a})\prod_{i\in\partial a}\mu_{i\to a}(x_i)\hat{Q}_{i\to a}} \\
&= \frac{\sum_{\underline{\mu}} \ln\Big\{\sum_{x_{\partial a}} f_a(x_{\partial a})\prod_{i\in\partial a}\mu_{i\to a}(x_i)\Big\}\sum_{x_{\partial a}} f_a(x_{\partial a})\prod_{i\in\partial a}\mu_{i\to a}(x_i)Q_{i\to a}}{\sum_{x_{\partial a}} f_a(x_{\partial a})\prod_{i\in\partial a}\mu_{i\to a}^{\text{av}}(x_i)} \\
&= \sum_{\underline{\mu}} \ln\Big\{\sum_{x_{\partial a}} f_a(x_{\partial a})\prod_{i\in\partial a}\mu_{i\to a}(x_i)\Big\}\sum_{x_{\partial a}}\nu_a^{\text{av}}(x_{\partial a})\prod_{i\in\partial a} R_{i\to a}(\mu_{i\to a}|x_i)
\end{aligned} \tag{16.46}$$

where the weighting factor over messages is again expressed thanks to an average marginal

$$\nu_a^{\text{av}}(x_{\partial a}) = \frac{f_a(x_{\partial a})\prod_{i\in\partial a}\mu_{i\to a}^{\text{av}}(x_i)}{\sum_{x_{\partial a}} f_a(x_{\partial a})\prod_{i\in\partial a}\mu_{i\to a}^{\text{av}}(x_i)} \tag{16.47}$$

and the conditional distribution

$$R_{i\to a}(\mu_{i\to a}|x_i) = \frac{\mu_{i\to a}(x_i)Q_{i\to a}}{\mu_{i\to a}^{\text{av}}(x_i)} \tag{16.48}$$

Similarly for the last term $\langle F_{ai}\rangle_{\text{cav}}$ using (11.17) we find for $x = 1$

$$\begin{aligned}
\langle F_{ai}\rangle_{\text{cav}}\big|_{x=1} &= \frac{\sum_{\underline{\mu},\underline{\hat{\mu}}} \ln\Big\{\sum_{x_i}\hat{\mu}_{a\to i}(x_i)\mu_{i\to a}(x_i)\Big\}\sum_{x_i}\hat{\mu}_{a\to i}(x_i)\hat{Q}_{a\to i}\mu_{i\to a}(x_i)Q_{i\to a}}{\sum_{\underline{\mu},\underline{\hat{\mu}}}\sum_{x_i}\hat{\mu}_{a\to i}(x_i)\hat{Q}_{a\to i}\mu_{i\to a}(x_i)Q_{i\to a}} \\
&= \frac{\sum_{\underline{\mu},\underline{\hat{\mu}}} \ln\Big\{\sum_{x_i}\hat{\mu}_{a\to i}(x_i)\mu_{i\to a}(x_i)\Big\}\sum_{x_i}\hat{\mu}_{a\to i}(x_i)\hat{Q}_{a\to i}\mu_{i\to a}(x_i)Q_{i\to a}}{\sum_{x_i}\hat{\mu}_{a\to i}^{\text{av}}(x_i)\mu_{i\to a}^{\text{av}}(x_i)} \\
&= \sum_{\underline{\mu},\underline{\hat{\mu}}} \ln\Big\{\sum_{x_i}\hat{\mu}_{a\to i}(x_i)\mu_{i\to a}(x_i)\Big\}\sum_{x_i}\nu_{ai}^{\text{av}}(x_i)\hat{R}_{a\to i}(\hat{\mu}_{a\to i}|x_i)R_{i\to a}(\mu_{i\to a}|x_i)
\end{aligned} \tag{16.49}$$

where

$$\nu_{ai}^{\text{av}}(x_i) = \frac{\hat{\mu}_{a\to i}^{\text{av}}(x_i)\mu_{i\to a}^{\text{av}}(x_i)}{\sum_{x_i}\hat{\mu}_{a\to i}^{\text{av}}(x_i)\mu_{i\to a}^{\text{av}}(x_i)} \tag{16.50}$$

So far we have shown that $\langle \mathcal{F}_{\rm Bethe}\rangle_{\rm cav}\big|_{x=1}$ can be entirely expressed in terms of the average messages $\hat{\mu}^{\rm av}_{a\to i}$, $\mu^{\rm av}_{i\to a}$ and the conditional distributions $\hat{R}_{a\to i}(\hat{\mu}_{a\to i}|x_i)$, $R_{i\to a}(\mu_{i\to a}|x_i)$. We have already seen that the average messages satisfy the usual sum-product equations. We will now show that the conditional distributions satisfy similar equations that are only slightly more complicated. Multiplying the cavity equations (16.28) by $\mu_{a\to i}(x_i)$ and $\hat{\mu}_{a\to i}(x_i)$, and using the expressions of the reweighting factor (16.29) we get for $x = 1$

$$\mu_{i\to a}(x_i)Q_{i\to a}(\mu_{i\to a}) \propto \sum_{\underline{\widehat{\mu}}} \Delta_{i\to a} \prod_{b\in\partial i\setminus a} \widehat{\mu}_{b\to i}(x_i)\widehat{Q}_{b\to i}(\widehat{\mu}_{b\to i})$$

$$\hat{\mu}_{a\to i}(x_i)\widehat{Q}_{a\to i}(\widehat{\mu}_{a\to i}) \propto \sum_{\sim x_i} f_a(x_{\partial a}) \sum_{\underline{\mu}} \hat{\Delta}_{a\to i} \prod_{j\in\partial a\setminus i} \mu_{j\to a}(x_j)Q_{j\to a}(\mu_{j\to a})$$

Dividing each member of these equalities by $\mu^{\rm av}_{i\to a}(x_i)$, $\mu^{\rm av}_{a\to i}(x_i)$ and using the sum-product equations (16.41) one finds a closed set of equations linking the conditional distributions,

$$\begin{cases} R_{i\to a}(\mu_{i\to a}|x_i) \propto \sum_{\underline{\widehat{\mu}}} \Delta_{i\to a} \prod_{b\in\partial i\setminus a} \hat{R}_{b\to i}(\hat{\mu}_{b\to i}|x_i) \\ \hat{R}_{a\to i}(\hat{\mu}_{a\to i}|x_i) \propto \sum_{\sim x_i} \pi_{a,i}(x_{\partial a\setminus i}|x_i) \sum_{\underline{\mu}} \hat{\Delta}_{a\to i} \prod_{j\in\partial a\setminus i} R_{j\to a}(\mu_{j\to a}|x_j) \end{cases} \tag{16.51}$$

where

$$\pi_{a,i}(x_{\partial a\setminus i}|x_i) = \frac{f_a(x_{\partial a}) \prod_{j\in\partial a\setminus i} \mu^{\rm av}_{j\to a}(x_j)}{\sum_{\sim x_i} f_a(x_{\partial a}) \prod_{j\in\partial a\setminus i} \mu^{\rm av}_{j\to a}(x_j)} \tag{16.52}$$

These equations are quite similar to standard sum-product equations and are much easier to solve than the original cavity equations.

Let us summarize the main result of these lengthy calculations. To calculate the complexity one must solve two sets of "sum-product" equations, the usual ones (16.41) and (16.51). Then one obtains the total free energy from the usual Bethe formula (see (16.39)) and the "internal energy" from (16.43), (16.46), (16.49). Finally the complexity equals to the difference (16.42) of the total free and internal free energies.

16.7 Application of the cavity equations to K-SAT

We apply the general theory to K-SAT using the notations and parametrizations of Sect. 9.4. Recall $J_{ia} = +1$ (resp. -1) labels full (resp. dashed) edges and $s_i = (-1)^{x_i} = J_{ia}$ (resp. $s_i = (-1)^{x_i} = -J_{ia}$) does not satisfy a (resp. satisfies a).

Messages $\mu_{i\to a}$ and $\hat{\mu}_{a\to i}$ are parametrized by fields $h_{i\to a}$, $\hat{h}_{a\to i}$ (see (9.14)) so in the cavity theory all sums over messages become integrals over fields. To ease the notations we will use a short hand notation for integrals over fields,

$\int dh Q(h)\cdots \to \int Q(h)\cdots$ and $\int d\hat{h}\hat{Q}(\hat{h})\cdots \to \int \hat{Q}(\hat{h})\cdots$, when no confusion is possible. Using (9.20) the constraints (16.21) become

$$\Delta_{i\to a} \to \delta_{i\to a} = \delta\big(h_{i\to a} - \sum_{b\in\partial i\setminus a} J_{ia}J_{ib}\hat{h}_{b\to i}\big)$$
$$\hat{\Delta}_{a\to i} \to \hat{\delta}_{a\to i} = \delta\big(\hat{h}_{a\to i} - \frac{1}{2}\ln\{1 - \prod_{j\in\partial a\setminus i}\frac{1-\tanh_{j\to a}}{2}\}\big).$$

Also, the weighting factors (16.29) become

$$\begin{aligned} z_{i\to a} = &\prod_{b\in\partial i\setminus a}(1 - J_{ia}J_{ib}\tanh\hat{h}_{b\to i}) \\ &+ \prod_{b\in\partial i\setminus a}(1 + J_{ia}J_{ib}\tanh\hat{h}_{b\to i}) \\ z_{a\to i} = &\, 2 - \prod_{j\in\partial a\setminus i}\frac{1}{2}(1-\tanh_{j\to a}) \end{aligned}$$

With these formulas the cavity equations (16.28) reduce to

$$\begin{cases} Q_{i\to a}(h_{i\to a}) = \int\{\prod_{b\in\partial i\setminus a}\hat{Q}_{b\to i}(\hat{h}_{b\to i})\}\,(z_{i\to a})^x\,\delta_{i\to a} \\ \hat{Q}_{a\to i}(\hat{h}_{a\to i}) = \int\{\prod_{j\in\partial a\setminus i}\hat{Q}_{j\to a}(h_{j\to a})\}\,(z_{a\to i})^x\,\hat{\delta}_{a\to i} \end{cases} \tag{16.53}$$

From (16.31)-(16.33) and (11.25)-(11.27) we get the expression of the level-one Bethe free energy for general x, $\mathcal{F}_{\rm Bethe}(x) = \sum_i \mathcal{F}_i(x) + \sum_a \mathcal{F}_a(x) - \sum_{ai}\mathcal{F}_{ai}(x)$,

$$\begin{aligned} \mathcal{F}_i(x) = &-x^{-1}\ln\Big\{\int\Big\{\prod_{a\in\partial i}\hat{Q}_{a\to i}(\hat{h}_{a\to i})\Big\}\Big(\prod_{a\in\partial i}(1 + J_{ia}\tanh\hat{h}_{a\to i}) \\ &\times\prod_{b\in\partial i}(1 - J_{ia}\tanh\hat{h}_{a\to i})\Big)^x\Big\} \end{aligned} \tag{16.54}$$

$$\mathcal{F}_a(x) = -x^{-1}\ln\Big\{\int\Big\{\prod_{j\in\partial a}Q_{j\to a}(h_{j\to a})\Big\}\Big(2 - \prod_{j\in\partial a}\frac{1-\tanh_{j\to a}}{2}\Big)^x\Big\} \tag{16.55}$$

$$\mathcal{F}_{ai}(x) = -x^{-1}\ln\Big\{\int\Big\{\hat{Q}_{a\to i}(\hat{h}_{a\to i})Q_{i\to a}(h_{i\to a})\Big\}\Big(1 + \tanh_{a\to i}\tanh_{i\to a}\Big)^x\Big\} \tag{16.56}$$

Similar formulas can be obtained for the complexity from $\Sigma_{\rm Bethe} = \partial\mathcal{F}_{\rm Bethe}/\partial x^{-1}$. In principle from the set of equations (16.53) and (16.54)-(16.56) one can deduce all predictions of the cavity theory for K-SAT. In the next section we deduce the *one step replica broken* (1RSB) formula for the average free energy; a formula that contrary to the replica symmetric one is believed to yield the correct free energy in all regimes.

In practice most important information about the phase diagram (e.g. the location of the thresholds) can be accessed from the complexity for $x = 1$. We now derive the formulas for this case. We introduce "average fields" to parametrize average messages $\hat{\mu}^{\rm av}_{a\to i}, \mu^{\rm av}_{i\to a}$, similarly to (9.14), namely $h^{\rm av} =$

$\frac{1}{2}\ln(\mu^{\rm av}(-J)/\mu^{\rm av}(J))$ (dropping edge subscripts and hats). Then because of (16.41) the average fields satisfy the usual belief propagation equations (this follows by the calculation shown in Sect. 9.4),

$$\begin{cases} h^{\rm av}_{i\to a} = \sum_{b\in\partial i\setminus a} J_{ia}J_{ib}\hat{h}^{\rm av}_{b\to i}, \\ \hat{h}^{\rm av}_{a\to i} = \frac{1}{2}\ln\{1-\prod_{j\in\partial a\setminus i}\frac{1}{2}(1-\tanh h^{\rm av}_{j\to a})\}. \end{cases} \tag{16.57}$$

The other set of sum-product equations (16.51) links the conditional distributions (recall $s_i=(-1)^{x_i}$)

$$\begin{cases} R_{i\to a}(h_{i\to a}|s_i) = \int\{\prod_{b\in\partial i\setminus a}\hat{R}_{b\to i}(\hat{h}_{b\to i}|s_i)\}\,\delta_{i\to a} \\ \hat{R}_{a\to i}(\hat{h}_{a\to i}|s_i) = \sum_{\sim s_i}\pi_{a,i}(s_{\partial a\setminus i}|s_i)\int\{\prod_{j\in\partial a\setminus i}R_{j\to a}(h_{j\to a}|s_j)\}\,\hat{\delta}_{a\to i} \end{cases} \tag{16.58}$$

where for K-SAT

$$\pi_{a,i}(s_{\partial a\setminus i}|s_i) = \text{recalculate this expression} \tag{16.59}$$

The internal free energy is derived from (16.43), (16.46), (16.49). We find the following three contributions

$$\langle F_i\rangle_{\rm cav}\big|_{x=1} = \sum_{s_i}\nu^{\rm av}_i(s_i)\int\prod_{a\in\partial i}\hat{R}_{a\to i}(\hat{h}_{a\to i}|s_i)\ln\Big\{\prod_{a\in\partial i}(1+J_{ia}\tanh\hat{h}_{a\to i}) \tag{16.60}$$

$$+\prod_{a\in\partial i}(1-J_{ia}\tanh\hat{h}_{a\to i})\Big\}$$

$$\langle F_a\rangle_{\rm cav}\big|_{x=1} = \sum_{s_{\partial a}}\nu^{\rm av}_a(s_{\partial a})\int\prod_{i\in\partial a}R_{i\to a}(h_{i\to a}|s_i)\ln\Big\{2-\prod_{j\in\partial a}\frac{1}{2}(1-\tanh_{j\to a})\Big\} \tag{16.61}$$

$$\langle F_{ai}\rangle_{\rm cav}\big|_{x=1} = \sum_{s_i}\nu^{\rm av}_{ai}(s_i)\int\hat{R}_{a\to i}(\hat{h}_{a\to i}|s_i)R_{i\to a}(h_{i\to a}|x_i) \tag{16.62}$$

$$\times\ln\Big\{1+\tanh\hat{h}_{a\to i}\tanh_{i\to a}\Big\}$$

where

$$\nu^{\rm av}_i(s_i) = \frac{1}{2}\big(1+s_i\tanh(\sum_{b\partial i}J_{ib}\hat{h}^{\rm av}_{b\to i})$$

$$\nu^{\rm av}_a(s_{\partial a}) = \text{recalculate this}$$

$$\nu^{\rm av}_{ai}(s_i) = \frac{1}{2}(1+s_i\tanh(\hat{h}^{\rm av}_{a\to i}+h^{\rm av}_{i\to a}))$$

Finally the complexity for $x=1$ is obtained by subtracting the internal free energy (given by (16.60)-(16.62)) from $F_{\rm Bethe}(\underline{h}^{\rm av},\underline{\hat{h}}^{\rm av})$ (given by (11.25)-(11.27)).

16.8 1RSB analysis for K-SAT

The cavity equations in Sect. 16.7 concern fixed instances. Here we analyze their consequences for random instances from the ensemble $\mathcal{F}(n, m, K)$. This leads to predictions called for historical reasons *one step replica symmetry broken* (1RSB) predictions or formulas (information on the seemingly strange terminology which comes from spin glass theory is found in the notes). It is important to keep in mind that there are two levels of randomness involved. We pick a factor graph at random so the messages $Q_{i\to a}(\cdot)$ and $\hat{Q}_{a\to i}(\cdot)$ are random (as in usual message passing in part II), and these random messages are themselves probability distributions over random variables $h_{i\to a}$ and $\hat{h}_{a\to i}$.

General 1RSB solution

To compute the ensemble average of the free energy one should in principle solve (16.53) for each (typical) graph and then average (16.54)-(16.56) over all graphs. Here we assume that in the large system size limit it is correct to treat the solutions of (16.53) as iid random variables $Q(\cdot)$ and $\hat{Q}(\cdot)$.

Let p and q be two Poisson($\alpha K/2$) random variables. These represent the degree distributions at the variable nodes from the edge as well as the node perspective. Then $Q(\cdot)$ and $\hat{Q}(\cdot)$ satisfy

$$Q(h) \stackrel{\text{distr}}{=} \frac{1}{\mathcal{N}} \int \prod_{\ell=1}^{p} \hat{Q}_\ell^+(\hat{h}_\ell^+) \prod_{\ell=1}^{q} \hat{Q}_\ell^-(\hat{h}_\ell^-) \Big(\prod_{\ell=1}^{p} (1-\tanh \hat{h}_\ell^+) \prod_{\ell=1}^{q} (1+\tanh \hat{h}_\ell^-) + \prod_{\ell=1}^{p} (1+\tanh \hat{h}_\ell^+) \prod_{\ell=1}^{q} (1-\tanh \hat{h}_\ell^-) \Big)^x \delta\big(h - (\sum_{\ell=1}^{p} h_\ell^+ - \sum_{\ell=1}^{q} h_\ell^-)\big) \tag{16.63}$$

$$\hat{Q}(\hat{h}) \stackrel{\text{distr}}{=} \frac{1}{\hat{\mathcal{N}}} \int \prod_{k=1}^{K-1} Q_k(h_k) \left(2 - \prod_{k=1}^{K-1} \frac{1}{2}(1-\tanh h_k)\right)^x \times \delta\big(\hat{h} - \frac{1}{2} \ln\{1 - \prod_{k=1}^{K-1} \frac{1}{2}(1-\tanh h_k)\}\big) \tag{16.64}$$

Here the symbol $\stackrel{\text{distr}}{=}$ means that the probability distributions of the right and left hand side are equal. More precisely let $\mathcal{Q}(\cdot)$, $\hat{\mathcal{Q}}(\cdot)$ denote the distribution of the r.v's $Q(\cdot)$ and $\hat{Q}(\cdot)$. Then in (16.63)-(16.64) the random variables are iid with $Q_\ell^\pm \sim \mathcal{Q}(\cdot)$ and $\hat{Q}_\ell \sim \hat{\mathcal{Q}}(\cdot)$, which imposes selfconsistent conditions on the distributions $\mathcal{Q}(\cdot)$ and $\hat{\mathcal{Q}}(\cdot)$.

Similarly, from the assumption of independence of messages, the free energy

of a the level-one model is given by the random variable

$$
\begin{aligned}
&f_1(x; Q(.), \hat{Q}(.), p, q) \\
&\quad = x^{-1} \ln\Bigg\{ \int \prod_{\ell=1}^{p} \hat{Q}_\ell^+(\hat{h}_\ell^+) \prod_{\ell=1}^{q} \hat{Q}_\ell^-(\hat{h}_\ell^-) \Bigg(\prod_{\ell=1}^{p} (1 - \tanh \hat{h}_\ell^+) \prod_{\ell=1}^{q} (1 + \tanh \hat{h}_\ell^-) \\
&\qquad + \prod_{\ell=1}^{p} (1 + \tanh \hat{h}_\ell^+) \prod_{\ell=1}^{q} (1 - \tanh \hat{h}_k^-) \Bigg)^x \Bigg\} \\
&\qquad + x^{-1} \ln\Bigg\{ \int \prod_{k=1}^{K} Q_k(h_k) \left(1 - \prod_{k=1}^{K} \frac{1 - \tanh h_k}{2} \right)^x \Bigg\} \\
&\qquad - x^{-1} \ln\Bigg\{ \int Q(h) \hat{Q}(\hat{h}) \left(1 + \tanh h \tanh \hat{h} \right)^x \Bigg\}
\end{aligned}
\tag{16.65}
$$

We are ready to formulate the precise 1RSB formula for the free energy. This is done in a manner analogous to the replica symmetric formulas in Chapter 12. Fix a trial distribution $\mathcal{Q}(\cdot)$ of a random variable $Q(\cdot)$. Take $K-1$ iid copies $Q_k(.) \sim \mathcal{Q}(\cdot)$, $k = 1, \cdots, K-1$, define the random variable $\hat{Q}(.)$ through (16.64), and call $\hat{\mathcal{Q}}(\cdot)$ the distribution induced by this same equation. Define the 1RSB free energy functional

$$
f_{\text{1RSB}}(x; \mathcal{Q}(.)) = \mathbb{E}[f_1(x; Q(.), \hat{Q}(.), p, q)] \tag{16.66}
$$

where the expectation is with respect to Q, p, q. The 1RSB formula states that the free energy of K-SAT at finite temperature is given by the variational formula

$$
-\lim_{n \to +\infty} \frac{1}{\beta n} \mathbb{E}[\ln Z] = \max_{x \in [0,1]} \sup_{\mathcal{Q}} f_{\text{1RSB}}(x; \mathcal{Q}(.)) \tag{16.67}
$$

We shall not prove it here but similarly to the replica symmetric variational problems the stationnarity condition for $\mathcal{Q}(\cdot)$ yields equation (16.63). For the maximum over x we must distinguish two cases. When it is attained at $x = 1$ the 1RSB formula (16.67) reduces to the replica symmetric solution (this was shown for instances in the preceding section and is briefly shown again below for the average free energy). This is the case for $\alpha < \alpha_c(\beta)$. As α increases past $\alpha_c(\beta)$ the maximum is attained for some $0 < x_*(\alpha) < 1$ and the stationarity condition is nothing else than the condition that the complexity vanishes.

Equ. (16.67) is conjectured to be exact. Thanks to an extension of the interpolation method developped in Chapter 13 it has ben proven that the 1RSB formula is a lower bound to the free energy.

THEOREM 16.1 *For any trial distribution $\mathcal{Q}(.)$ and any $0 < x < 1$, the thermodynamic limit of the free energy of SAT exists, and moreover is lower bounded by the 1RSB formula*

$$
-\lim_{n \to +\infty} \frac{1}{\beta n} \mathbb{E}[\ln Z] \geq \max_{x \in [0,1]} \sup_{\mathcal{Q}} f_{\text{1RSB}}(x; \mathcal{Q}(.))
$$

Reduction of the free energy to the replica symmetric solution for $x = 1$

We first demonstrate that for $x = 1$ the 1RSB formula (16.67) for the free energy reduces to the replica symmetric result found in Sect. 12.5. Let us define the random variables h^{rmav} and $\hat{h}^{rmav}$ through

$$\tanh h^{\mathrm{av}} = \int Q(h) \tanh h, \qquad \tanh \hat{h}^{\mathrm{av}} = \int \hat{Q}(\hat{h}) \tanh \hat{h} \tag{16.68}$$

The distribution of h^{av} is induced by $Q(\cdot) \sim \mathcal{Q}(\cdot)$, and equation (16.64) defines the random variable $\hat{Q}(\cdot) \sim \hat{\mathcal{Q}}(\cdot)$ which induces a distribution for $\hat{h}^{\mathrm{av}}$. In fact, for $x = 1$, from (16.64) we can deduce the explicit relation between the random variables h^{av} and $\hat{h}^{\mathrm{av}}$. Multiplying (16.64) by $\tanh \hat{h}$ and integrating both sides over $\hat{h}$ leads after a few lines of algebra (for $x = 1$) to the "density evolution" relation (see exercises)

$$\hat{h}^{\mathrm{av}} \stackrel{\mathrm{distr}}{=} \frac{1}{2} \ln\{1 - \prod_{k=1}^{K-1} \frac{1}{2}(1 - \tanh h_k^{\mathrm{av}})\} \tag{16.69}$$

This result should not appear as a surprise in view of the general formulas (16.57) (derived for instances). Now, putting $x = 1$ in (16.65) it is immediate to see that the random variable $f_1(x; Q(.), \hat{Q}(.), p, q)$ reduces to

$$\begin{aligned} f_1(x = 1; \underline{h}^{\mathrm{av}}, \underline{\hat{h}}^{\mathrm{av}}, p, q) = \ln\Big\{ & \prod_{\ell=1}^{p} (1 - \tanh \hat{h}_\ell^{+,\mathrm{av}}) \prod_{\ell=1}^{q} (1 + \tanh \hat{h}_\ell^{-,\mathrm{av}}) \\ & + \prod_{\ell=1}^{p} (1 + \tanh \hat{h}_\ell^{+,\mathrm{av}}) \prod_{\ell=1}^{q} (1 - \tanh \hat{h}_\ell^{-,\mathrm{av}}) \Big\} \\ & + \ln\Big\{ 1 - \prod_{k=1}^{K} \frac{1}{2}(1 - \tanh h_k^{\mathrm{av}}) \Big\} + \ln\Big\{ 1 + \tanh h^{\mathrm{av}} \tanh \hat{h}^{\mathrm{av}} \Big\} \end{aligned} \tag{16.70}$$

If we call $\mathrm{x}(\cdot)$ the trial distribution of h^{av} that is induced by $Q(\cdot) \sim \mathcal{Q}(\cdot)$ the 1RSB free energy function (16.66) reduces to

$$f_{\mathrm{RS}}(\mathrm{x}(\cdot)) = \mathbb{E}[f_1(x = 1; \underline{h}^{\mathrm{av}}, \underline{\hat{h}}^{\mathrm{av}}, p, q)] \tag{16.71}$$

where the expectation is over $\mathrm{x}(\cdot)$ and the Poisson$(\alpha K/2)$ integers p and q. We recognize here the replica symmetric free energy functional, namely (12.29). The replica symmetric free energy is obtained by maximizing the functional over $\mathrm{x}(\cdot)$; and not surpisingly the stationarity condition is the second "density evolution" equation

$$h^{\mathrm{av}} \stackrel{\mathrm{distr}}{=} \sum_{\ell=1}^{p} \hat{h}_\ell^{+,\mathrm{av}} - \sum_{\ell=1}^{q} \hat{h}_\ell^{-,\mathrm{av}} \tag{16.72}$$

Finally we note that Theorem 16.1 implies the replica symmetric lower bound (valid for all α)

$$-\lim_{n \to +\infty} \frac{1}{\beta n} \mathbb{E}[\ln Z] \geq \max_{x \in [0,1]} \sup_{\mathcal{Q}} f_{\mathrm{1RSB}}(x; \mathcal{Q}(.)) \geq \sup_{\mathrm{x}(\cdot)} f_{\mathrm{RS}}(\mathrm{x}(\cdot)) \tag{16.73}$$

which we proved in Chapter 13.

1RSB complexity function for $x = 1$

For specific instances we saw that the complexity at $x = 1$ is found by subtracting the internal free energy contributions (16.60)-(16.62) from the Bethe free energy (11.25)-(11.27). To compute the 1RSB complexity of random formulas we need the averaged version these equations. For random formulas the average Bethe free energy is already given by the replica symmetric free energy, i.e (16.71) maximized over the trial distribution x$(\cdot)$. So we just have to discuss the average of the internal free energy contributions (16.60)-(16.62).

We start with the averaged form of (16.58). These become relations between distributions

$$\begin{cases} R(h|s,h^{\rm av}) = \frac{1}{\mathcal{N}} \int \prod_{\ell=1}^{p} \hat{R}^{+}_{\ell}(\hat{h}_\ell|s,\hat{h}^{+,{\rm av}}_{\ell}) \prod_{\ell=1}^{q} \hat{R}^{-}_{\ell}(\hat{h}_\ell|s,\hat{h}^{-,{\rm av}}_{\ell}) \\ \qquad\qquad \times\delta\big(h-(\sum_{\ell=1}^{p} h^{+}_{\ell} - \sum_{\ell=1}^{q} h^{-}_{\ell})\big) \\ \hat{R}(\hat{h}|s,\hat{h}^{\rm av}) = \frac{1}{\hat{\mathcal{N}}} \sum_{\sim s} \pi(s_1,\cdots,s_{K-1}|s) \int \prod_{k=1}^{K-1} R_k(h_k|s,h^{\rm av}_k) \\ \qquad\qquad \times\hat{\delta}\big(\hat{h}-\frac{1}{2}\ln\{1-\prod_{k=1}^{K-1}\frac{1}{2}(1-\tanh h_k)\}\big) \end{cases} \tag{16.74}$$

where

$$\pi(s_1,\cdots,s_{K-1}|s) = {\rm recalculatethisexpression}$$

The only level of randomness in this equation is in the average fields which are solutions of the distributional equations (16.69) and (16.72). To better understand (16.74) it is instructive to rederive these equations directly from the general 1RSB fixed point equations (16.63)-(16.64) for K-SAT. First define

$$\bar{Q}(h|h^{\rm av}) = \mathbb{E}_{\mathcal{Q}(\cdot|h^{\rm av})}[Q(h)], \qquad \hat{\bar{Q}}(\hat{h}|\hat{h}^{\rm av}) = \mathbb{E}_{\hat{\mathcal{Q}}(\cdot|\hat{h}^{\rm av})}[\hat{Q}(\hat{h})]$$

where the conditional expectations mean that we "integrate" over Q and $\hat{Q}$ such that (16.68) holds. One should now bear in mind that the only randomness in $\bar{Q}(h|h^{\rm av})$ and $\hat{\bar{Q}}(\hat{h}|\hat{h}^{\rm av})$ is in $h^{\rm av}$ and $\hat{h}^{\rm av}$ (the "overbar" is here to remind us that, given the average fields, these distributions are not random, contrary to $Q, \hat{Q}$). These distributions satisfy exactly the same equations (16.63)-(16.64). This follows immediately by taking the conditional expectation of these equations and using the fact they are multilinear in Q's and $\hat{Q}$'s. Now define

$$R(h|s,h^{\rm av}) \equiv \frac{(1+s\tanh h)\bar{Q}(h|h^{\rm av})}{1+s\tanh h^{\rm av}}, \qquad \hat{R}(\hat{h}|s) \equiv \frac{(1+s\tanh \hat{h})\hat{\bar{Q}}(\hat{h}|\hat{h}^{\rm av})}{1+s\tanh \hat{h}^{\rm av}}$$

The reader will note these relations are nothing else than an averaged version of (16.48), (16.45). It follows that

$$\bar{Q}(h|h^{\rm av}) = \sum_{s=\pm 1} \frac{1}{2}(1+s\tanh h^{\rm av}) R(h|s,h^{\rm av}),$$

and similarly for $\hat{\bar{Q}}(\hat{h}|\hat{h}^{\rm av})$. Replacing these representations of $\bar{Q}$ and $\hat{\bar{Q}}$ in (16.63)-(16.64) (more precisely in the conditional expectation of these equations) we obtain (16.74).

At this point it is quite easy to see that the expected version of internal free energy contributions (16.60)-(16.62) becomes (conditional on the average fields)

$$
\begin{aligned}
F_{\rm int}(\underline{h}^{\rm av},\underline{\hat{h}}^{\rm av}) &= \sum_{s=\pm 1}\nu^{\rm av}(s)\int\prod_{\ell=1}^{p}\hat{R}^{+}_{\ell}(\hat{h}^{+}_{\ell}|s,h^{+,{\rm av}}_{\ell})\prod_{\ell=1}^{q}\hat{R}^{-}_{\ell}(\hat{h}^{-}_{\ell}|s,h^{-,{\rm av}}_{\ell})\\
&\times\ln\Big\{\prod_{\ell=1}^{p}(1-\tanh\hat{h}^{+}_{\ell})\prod_{\ell=1}^{q}(1+\tanh\hat{h}^{-}_{\ell})+\prod_{\ell=1}^{p}(1+\tanh\hat{h}^{+}_{\ell})\prod_{\ell=1}^{q}(1-\tanh\hat{h}^{-}_{\ell})\Big\}\\
&+\sum_{s_1,\cdots,s_K}\nu^{\rm av}(s_1,\cdots,s_K)\int\prod_{k=1}^{K}R_k(h_k|s,h^{\rm av})\ln\Big\{2-\prod_{k=1}^{K}\frac{1}{2}(1-\tanh h_k)\Big\}\\
&-\sum_{s}\nu^{\rm av}(s)\int\hat{R}(\hat{h}|s,\hat{h}^{\rm av})R(h|s,h^{\rm av})\ln\Big\{1+\tanh\hat{h}\tanh h\Big\}
\end{aligned}
\tag{16.75}
$$

where

$$
\begin{aligned}
\nu^{\rm av}(s) &= \frac{1}{2}\big(1+s\tanh(\sum_{\ell=1}^{p}\hat{h}^{+,{\rm av}}_{\ell}-\sum_{\ell=1}^{q}\hat{h}^{-,{\rm av}}_{\ell})\big)\\
\nu^{\rm av}(s_1,\cdots,s_K) &= \text{recalculate this}\\
\nu^{\rm av}(s) &= \frac{1}{2}(1+s\tanh(\hat{h}^{\rm av}+h^{\rm av}))
\end{aligned}
$$

Let us summarize. To find the complexity one first solves (16.69) and (16.72) in order to find the distributions $\mathrm{x}(\cdot),\hat{\mathrm{x}}(\cdot)$ of the average fields and obtain $f_{\rm RS}$, Equ. (16.71). For each typical instance of the average fields one solves (16.74) to find the distributions $R(\cdot|\pm 1,h^{\rm av}),\hat{R}(\cdot|\pm 1,\hat{h}^{\rm av})$ and obtain (16.75). Then one computes the average internal free energy (the expecetation is with respect to $\mathrm{x}(\cdot)$ and Poisson integers p, q)

$$
f_{\rm int}=\mathbb{E}[F_{\rm int}(\underline{h}^{\rm av},\underline{\hat{h}}^{\rm av})] \tag{16.76}
$$

The complexity is finally given by

$$
\Sigma_{\rm 1RSB}(\alpha)=f_{\rm RS}-f_{\rm int} \tag{16.77}
$$

16.9 Phase diagram of K-SAT at finite temperature

Figure 16.5 shows the 1RSB phase diagram in the (α,β^{-1}) plane (i.e. constraint density and temperature) plane. The 1RSB analysis predicts the existence of two thresholds $\alpha_d(\beta)$ and $\alpha_c(\beta)$, called the *dynamical* and *condensation* thresholds, at which the nature of the convex decomposition (16.13) changes drastically. A few values of the thresholds are given in Table 16.1 for $\beta\to+\infty$ and compared to the SAT-UNSAT threshold for a few values of K. Note that the case $K=3$

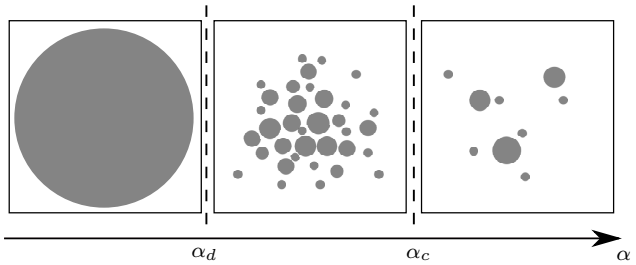


Figure 16.5 The 1RSB analysis predicts two thresholds $\alpha_d(\beta)$ and $\alpha_c(\beta)$. On the left of the dynamical threshold there is only one extremal Gibbs state depicted by a large ball. Between the dynamical and the condensation threholds the Gibbs distribution is described by a convex superposition of an exponential number of extremal Bethe states depicted by numerous balls. On the right of the condensation threshold there are only finitely many extremal Bethe states that dominate the convex superposition. The size of the balls represents their internal free energy.

stands out because $\alpha_d = \alpha_c$. It is only for $K \geq 4$ that the behavior is generic. We briefly indicate how the thresholds are found and discuss their significance.

The easiest way to access the thresholds is to compute the complexity for $x = 1$ by using a population dynamics method (see exercises). We first remark that equations (16.74) always have (for all α) a trivial solution $R(h| \pm 1, h^{\rm av}) = \delta(h - h^{\rm av})$, $\hat{R}(\hat{h}| \pm 1, \hat{h}^{\rm av}) = \delta(\hat{h} - \hat{h}^{\rm av})$. Replacing this trivial solution in the 1RSB expression of the complexity for $x = 1$ one finds a cancellation between the free and internal energy contributions leading to a zero complexity. The dynamical threshold is defined as the constraint density where a non trivial solution appears for $\alpha > \alpha_d(\beta)$. Therefore for $\alpha < \alpha_d(\beta)$ the complexity vanishes $\Sigma_{\rm 1RSB}(\alpha) = 0$, the replica symmetric free energy is correct, and one expects that there is a unique Bethe extremal state. For $\alpha > \alpha_d(\beta)$ one has to select the non-trivial solution of (16.74) which maximizes the 1RSB free energy functional. One finds a complexity $\Sigma_{\rm 1RSB}(\alpha)$ which jumps to a strictly positive value at $\alpha_d(\beta)$, and decreases monotonically until it becomes negative past a threshold $\alpha_c(\beta)$. A negative value for the complexity is not consistent and this means that for $\alpha > \alpha_c(\beta)$ it is not correct to set $x = 1$. For $\alpha_d(\beta) < \alpha < \alpha_c(\beta)$ the value $x = 1$ is correct since it maximizes the 1RSB free energy. Indeed recall since $\Sigma_{1RSB}(\alpha) > 0$ the maximum is attained at the right boundary of the interval $[0, 1]$. In this intermediate regime of densities the theory predicts an exponentially large, namely $exp(n\Sigma(\alpha))$, number of extremal Bethe states α that equally dominate convex decomposition of the Gibbs distribution (16.13). The 1RSB formula for the free energy F_α of the dominant extremal Bethe states is given by (16.76) (the "internal energy" of the level-one model). The total free energy is still given by the replica symmetric formula which is analytic. There is no static phase transition in the whole range $0 < \alpha < \alpha_c$. For $\alpha > \alpha_c(\beta)$ the complexity one cannot set $x = 1$ and has to resort to the general 1RSB solution. Equations (16.63)-(16.64) are solved by a population dynamics technique and

K	α_d	α_c	α_s
3	3.86	3.86	4.267
4	9.38	9.55	9.93

Table 16.1 Dynamical and condensation thresholds of K-SAT in the zero temperature limit. Note that for 3-SAT the dynamical and condensation thresholds are the same. We also indicate for comparison the SAT-UNSAT threshold computed in the next chapter

the average free energy and complexity of the level one model are computed for all $0 < x < 1$. The correct value $x_*(\alpha)$ is the one that maximizes the free energy, i.e.such that such that $\Sigma(x_*(\alpha)) = 0$. The complexity vanishes and the number of Bethe extremal states dominating the convex decomposition of the Gibbs measure is expected to be finite (or subexponential). The finiteness of the dominant extremal Bethe states implies that the free energies are equal to the 1RSB free energy (16.67) (equivalently $\Sigma(x_*(\alpha)) = 0$ means that the entropy of the level-one model vanishes).

For densities below the dynamical threshold there is a unique Bethe extremal state and therefore belief propagation correctly computes the marginals of the Gibbs distribution. For intermediate densities between the dynamical and condensation thresholds there are exponentially many extremal Bethe states. But as we saw in this regime $x = 1$ and the cavity equations reduce to the belief propagation equations. Therefore the belief propagation equations still yield correct marginals. This is quite remarkable. The convex superposition (16.13) is not an extremal Bethe state but its marginals still satisfy the sum-product equations. For this reason the convex superposition is sometimes said to be a *Bethe state* albeit a notationan extremal one. We already pointed out that the dynamical threshold does not correspond to a static phase transition. Rather it is believed to have some algorithmic significance. For example a Markov Chain Monte Carlo algorithm will not be able to correctly sample the Gibbs measure. The condensation threshold is a static phase transition threshold where the free energy has a non-anlyticity. Sum-product equations do not correctly compute the marginals of the Gibbs distribution.

16.10 Long range correlations

In section **??** we indicated that in Ising models there is an intimate connection between the decay of correlations and the extremality of the Gibbs measure. This is also true for constarint satisfaction models defined on random graph ensembles. However the correct correlation functions have to be used. In the present context two type of correlation functions have been discovered. Point-to-set correlations

K	α_d	$\alpha_{d,80,3}$	α_c	$\alpha_{c,80,3}$	α_s	$\alpha_{s,80,3}$
3	3.86	3.86	3.86	3.86	4.267	4.268
4	9.38	9.55	9.55	9.56	9.93	10.06

Table 16.2 Thresholds of individual and coupled K-SAT model for $L = 80$ and $w = 3$. Note that for 3-SAT the dynamical and condensation thresholds are the same. The condensation and SAT-UNSAT thresholds correspond to non analyticities of the entropy and ground state energy and remain unchanged (for $L \to +\infty$). Already for $w = 3$ the dynamical threshold saturates very close to α_c and α_s.

defined as

$$C(i,B) = \sum_{\underline{x}_{\partial B}} \nu(\underline{x}_{\partial B}(\nu(x_i|\underline{x}_{\partial B}) - \nu(x_i))^2$$

where B is the set $\{x_j|\{\text{dist}(x_i,x_j) \geq d\}$. Within the cavity method one can compute $\lim_{d\to+\infty}\lim_{n\to+\infty} C(i,B$ and finds that the limit vanishes $\alpha < \alpha_d$, while it remains strictly positive for $\alpha > \alpha_d$. Moreover for all $\alpha < \alpha_c$ and all randomly chosen bounded set of variables

$$\mathbb{E}[(\nu(x_{i_1},\dots x_{i_k}) - \nu(x_{i_1})\dots\nu(x_{i_k}))^2] = O(\frac{1}{n})$$

This is similar to the decoupling property we discussed for the CW model. At α_c this decoupling property breaks down.

16.11 Thresholds of spatially coupled K-SAT

It is interesting to consider the spatially coupled version of the K-SAT model. The same cavity theory can be applied and the RSB equations solved with the appropriate boundary conditions. this allows to determine the dynamical and condensation thresholds of the spatially coupled model (see table **??**). The numerical observations suggest that the condensation threshold remains invariant in the limit of an infinite chain. This is consistent with its interpretation as a singularity of the entropy. In fact one can prove by the interpolation method that the entropy of the infinite coupled chain and underlying uncoupled model are the same, and therefore α_c is the same for both models, namely $\lim_{L\to+\infty}\alpha_c(w,L) = \alpha_c$. On the other hand it is observed that the dynamical threshold saturates towards the condensation threshold in the limit of an infinite chain and a large coupling range, namely $\lim_{w\to+\infty}\lim_{L\to+\infty}\alpha_d(w,L) = \alpha_c$. These results are conssistent with the interpretation of the dynamical threshold as an algorithmic barrier and the condensation threshold as a static phase transition threshold.

16.12 Notes

Problems

16.1 Population dynamics for $x = 1$.
16.2 Population dynamics for the generagl 1RSB solution.

17 Survey Propagation Guided Decimation Algorithm

In this Chapter we turn to one of the most fascinating aspects of the K-SAT problem. Namely we want to devise good algorithms that are able to find with positive probability solutions of random formulas up to - or at least very close to - the satisfiability threshold $\alpha_s(K)$. In particular we also want to concretely compute $\alpha_s(K)$.

Belief propagation guided decimation algorithms find solutions of random K-SAT formulas for low constraint densities. However, as experimentally observed in Chapter 9, such algorithms fail slightly below the dynamical (zero temperature) threshold α_d. The cavity method gives some intuition for the reasons of this failure. When $\alpha_d < \alpha < \alpha_c$ the marginals computed from belief propagation form an average over all marginals of extremal Bethe states and cannot be used to reliably fix variables that we decimate. For $\alpha > \alpha_c$ the situation is even worse because long range point-to-point correlations appear and belief propagation does not even yield a correct average of marginals. Ideally, in order to find solutions of a random formula for $\alpha > \alpha_d$ one would like to sample directly from Bethe extremal measures. While it is not really clear how this can be achieved, one may hope to use the informaton contained in the cavity method marginals of the level-one model - the *surveys* - to achieve something similar. In this chapter we start from this insight to derive an algorithm, that goes under the name of *survey propagation guided algorithm*, which successfully finds solutions above the dynamical threshold. The algorithm is again a decimation algorithm but now the decisions on the value that a variable should take are based on the surveys. It should be stressed that this algorithm does not represent a correct sampling of the uniform measure over K-SAT solutions and it is believed that it somehow finds "special" solutions. The problem of sampling from the uniform measure over solutions of a random formula is not addressed here.

In order to use the cavity method marginals of the level-one model to find solutions then one should apply the formalism of Chapter 16 at zero temperature. However this is not quite enough. One difficulty is that we weighted the extremal Bethe states by their free energy, which at zero temperature and in the satisfiable phase reduces to their entropy (in the satisfiable phase their internal energy vanishes since all constraints are satisfied). So when the temperature vanishes, in the satisfiable phase the extremal Bethe states are weighted by their entropy. One sometimes refers to this limit as the *entropic cavity method*. The entropic

cavity method correctly captures the uniform distribution over solutions and correctly predicts the dynamical and condensation thresholds, but it leaves out all states that dont have maximal entropy as well as solutions within them. If the goal is to construct an algorithm which finds *some* solution the entropic method may not be the best one. Even more so, close to the satisfiability threshold where the number of entropically dominant states diminishes. This leads us to consider a variant of the cavity method, sometimes called *energetic cavity method*, where extremal states are weighted by their *internal energy* instead of their free energy. At zero temperature and in the satisfiable phase the internal energy vanishes and the corresponding level-one model equally counts *all* states, not only those with maximal entropy. This allows to define an *energetic complexity* function - the log of the total number of extremal states - and identify $\alpha_s(K)$ as the density where this complexity becomes negative.

In the next section 17.1 we formulate the energetic cavity method. This leads to the a new set of message passing equations, the *survey propagation* equations, derived in **??**. This formalism is applied to given instances of K-SAT formulas in section 17.3 and to random formulas in section 17.4 where we also compute the energetic complexity and satisfiability threshold. In the last section 17.5 we come back to specific K-SAT instances and discuss the survey propagation guided decimation algorithm. As we willl see this algorithm works for constraint densities that come rather close to the satisfiability threshold.

17.1 Energetic cavity method

Recall the Hamiltonian formulation of the K-SAT problem in Chapter **??**. Solutions of a given K-SAT formula are the minimizers of a cost function (3.53). In this section we wish to keep the formalism quite general so we consider cost functions (or Hamiltonians) of the form

$$\mathcal{H}(\underline{x}) = \sum_{a=1}^{m} E_a(x_{\partial a}) \tag{17.1}$$

where $E_a(x_{\partial a})$ is the energy cost of an assignment $\{x_i, i \in \partial a\}$ of the variables attached to function node a. We will assume that $\min E_a(x_{\partial a}) = 0$ for all $a = 1, \cdots, m$. We seek zero energy minimisers, in other words assignments $\underline{x}$ which simultaneously satisfy $E_a(x_{\partial a}) = 0$ for all $a = 1, \cdots, m$.

Here we develop the energetic cavity method by directly tackling this minimisation problem. As will become clear later on the energetic cavity method can be viewed as a suitable zero temperature limit of the general cavity method developed in Chapter 16.

Min-sum formalism

Just as the finite temperature cavity method is based on the sum-product equations, the starting point of the energetic cavity method are the min-sum equations which we already encountered several times. These are a set of message passing equations relating "energy costs"

$$\begin{cases} E_{i\to a}(x_i) = \sum_{b\in\partial i\setminus a} \hat{E}_{b\to i}(x_i) - C_{i\to a} \\ \hat{E}_{a\to i}(x_i) = \min_{\sim x_i} \left\{ E_a(x_{\partial a}) + \sum_{j\in\partial a\setminus i} E_{j\to a}(x_j) \right\} - \hat{C}_{a\to i} \end{cases} \tag{17.2}$$

Here we adjust the "normalization constants" $C_{i\to a}, \hat{C}_{a\to i}$ so that $\min E_{i\to a}(x_i) = \min \hat{E}_{a\to i}(x_i) = 0$. Min-sum equations are the stationary point conditions of a *Bethe energy functional*,

$$E_{\mathrm{Bethe}}[\{E_{i\to a}(\cdot), \hat{E}_{a\to i}(\cdot)\}] = \sum_i E_i + \sum_a E_a - \sum_{ai} E_{ai} \tag{17.3}$$

with contributions from variable and function nodes, and edges,

$$E_i = \min_{x_i}\{\sum_{b\in\partial i} \hat{E}_{b\to i}(x_i)\} \tag{17.4}$$

$$E_a = \min_{x_{\partial a}}\{E_a(x_{\partial a}) + \sum_{i\in\partial a} E_{i\to a}(x_i)\} \tag{17.5}$$

$$E_{ai} = \min_{x_i}\{E_{i\to a}(x_i) + \hat{E}_{a\to i}(x_i)\} \tag{17.6}$$

Note that we can shift the constant factors $C_{i\to a}$ and $\hat{C}_{a\to i}$ without affecting the value of the Bethe energy functional (17.3).

From the messages we can also compute a "marginal energy cost"

$$E_i(x_i) = \sum_{a\in\partial i} \hat{E}_{a\to i}(x_i) - \min\{\sum_{a\in\partial i} \hat{E}_{a\to i}(x_i)\} \tag{17.7}$$

These marginal energy costs can be used to guide a decimation algorithm in order to find minimizers of the Hamiltonian.

As already pointed out in previous chapters (in the context of compresive sensing) the min-sum formalism summarized above can be derived from the sum-product equations and Bethe free energy functional by taking a suitable zero temperature limit. The essential point is to represent the messages at low temperatures as $\mu_{i\to a}(x_i) \propto e^{-\beta E_{i\to a}(x_i)}$ and $\mu_{a\to i}(x_i) \propto e^{-\beta \hat{E}_{a\to i}(x_i)}$.

Landscape, complexity and level-one energetic model

When the factor graph is a tree the message passing equations have one valid solution and the Bethe energy yields the exact ground state energy, i.e. $\min \mathcal{H}(\underline{x}) = E_{\mathrm{Bethe}}$. At the same time the marginal energy cost (17.7) gives the exact excitation energy, i.e. $E_i(x_i) = \min_{\sim x_i} \mathcal{H}(\underline{x}) - \min \mathcal{H}(\underline{x})$.

Here we are interested in cases where the factor graph is not a tree and the

— figure —

Figure 17.1 Cartoon of the energy landscape. The horizontal axis represents the space of energy costs $\{E_{i\to a}(\cdot), \hat{E}_{a\to i}(\cdot)\}$ and the vertical axis is the Bethe energy (17.3).

min-sum equations may have numerous solutions, possibly exponentially many in n, which requires a statistical description in the spirit of the level-one model of the previous chapter. The Bethe energy functional is viewed as an effective energy landscape over the space of messages $\{E_{i\to a}(\cdot), \hat{E}_{a\to i}(\cdot)\}$, with numerous minima, maxima and saddles. We consider models such that the energy landscape has a large number of minima at vanishing energy as illustrated Figure 17.1 we illustrates such an energy landscape for a model in its satisfiable phase where there are a large number of minima of vanishing energy. We introduce a counting function - the *energetic complexity* - which will enable us to count the minima of vanishing energy. For $\epsilon > 0$ we set

$$e^{n\Sigma_{\mathrm{Bethe}}(\epsilon)}\, d\epsilon = \sum_{\text{stat points}} \delta(n\epsilon - E_{\mathrm{Bethe}}(\{E_{i\to a}(\cdot), \hat{E}_{a\to i}(\cdot)\}))\, d\epsilon. \tag{17.8}$$

where the sum is over the solutions of (17.2), in other words stationary points of (17.3). The energetic complexity *counts* such stationary points which fall in the energy band $[n\epsilon, n(\epsilon+d\epsilon)]$. As $\epsilon \to 0$ we expect that the minima are exponentially more numerous than maxima and saddles and therefore we expect that only minima contribute to

$$\lim_{\epsilon\to 0} \Sigma_{\mathrm{Bethe}}(\epsilon).$$

The ensemble average of this complexity at zero energy allows to distinguish between a SAT and UNSAT phases. Indeed we expect that it is non-negative if and only if solutions exist (for typical formulas).

To make this discussion more concrete let us briefly ilustrate the situation for random K-SAT. We will find that the average zero-energy complexity vanishes for $\alpha < \alpha_{\mathrm{SP}}$ (called the survey propagation threshold), jumps to a positive value at α_{SP} and then monotonically decreases until it becomes negative at a density $\alpha_s(K)$ and looses any meaning. The interpretation is that there exists only one minimum of zero energy for the Bethe energy functional as long as $\alpha < \alpha_{\mathrm{SP}}$, then an exponential number of such minima appear until the satisfiability threshold

— figure —

Figure 17.2 The average energetic complexity of the 4-SAT ensemble of random formulas.

$\alpha_s(K)$ where no solutions exist and zero energy minima disappear. Figure 17.2 shows the average energetic complexity of the 4-SAT ensemble as a function of α.

In order to compute $\Sigma_{\text{Bethe}}(\epsilon)$ we introduce the Laplace transform of (17.1)

$$\begin{aligned}\Xi(y) &= \int_0^{+\infty} d\epsilon\, e^{-n(y\epsilon - \Sigma_{\text{Bethe}}(\epsilon))} \\ &= \sum_{\text{stat points}} e^{-yE_{\text{Bethe}}[\{E_{i\to a}(\cdot),\hat{E}_{a\to i}(\cdot)\}]}\,. \end{aligned} \tag{17.9}$$

When the parameter $y \to +\infty$ the sum on the right hand side is dominated by the minima of vanishing energy, therefore

$$\lim_{\epsilon\to 0} \Sigma_{\text{Bethe}}(\epsilon) = \lim_{y\to+\infty} \frac{1}{n} \ln \Xi(y) \tag{17.10}$$

More generally, we can compute the complexity at any energy level, in other words count the number of stationary points in any energy band $[n\epsilon, n(\epsilon + d\epsilon)]$. For $n \to +\infty$ (17.9) formally implies

$$\begin{aligned} -\lim_{n\to+\infty} \frac{1}{n} \ln \Xi(y) &= \min_{\epsilon\geq 0}(y\epsilon - \Sigma_{\text{Bethe}}(\epsilon)) \\ &= y\epsilon_*(y) - \Sigma_{\text{Bethe}}(\epsilon_*(y)). \end{aligned} \tag{17.11}$$

where $\epsilon_*(y)$ is the solution of $y = \Sigma'_{\text{Bethe}}(\epsilon_*)$.

Clearly, (17.9) is a new statistical mechanical model for a system whose "microscopic degrees of freedom" are the energy costs satisfying the min-sum equations (the stationary points of the Bethe energy functional), "Hamiltonian" given by the Bethe functional, and "temperature" y. According to the general thermodynamic relations the "free energy" $-y^{-1} \ln \Xi(y)$ is given by the "internal energy" $\epsilon_*(y)$ minus y times the "entropy" $\Sigma_{\text{Bethe}}(\epsilon_*(y))$. At zero "temperature" $y \to +\infty$ (and in a SAT phase) the internal energy vanishes, therefore $\lim_{y\to+\infty} \Xi(y)$ reduces to the zero energy entropy $\Sigma_{\text{Bethe}}(\epsilon = 0)$. Note that in this model the energetic complexity is nothing else than a "Boltzman entropy".

It is also illuminating to clarify the connection between the energetic level-one model and the more general one introduced in 16. In the level one model of the prvious chapter the extremal Bethe states are given the Gibbs weights $e^{-\beta x F_{\rm Bethe}[\underline{\mu},\underline{\hat{\mu}}]}$. In the zero temperature limit $\mu_{i\to a}(x_i) \propto e^{-\beta E_{i\to a}(x_i)}$, $\mu_{a\to i}(x_i) \propto e^{-\beta \hat{E}_{a\to i}(x_i)}$ and $\lim_{\beta\to+\infty} F_{\rm Bethe} = E_{\rm Bethe}$. We see that the partition function (17.9) is recovered from (16.19) by letting $\beta \to +\infty$, and $x \to 0$ such that $\beta x = y$ is fixed. The energetic cavity method appears as a special case of the general cavity method developped in the previous chapter in a limit of very low temperatures where we weight the extremal states according to their energy instead of free energy. In this way when $y \to +\infty$ the partition function $\Xi(y)$ counts "zero energy extremal states" (or ground states) all with the same weight.

Factor graph representation of the energetic level-one model

The new partition function (17.9) can be represented as a factor graph model in essentially the same manner than in section 16.4 for the general level-one model. Here it will be sufficient to be brief.

To take into account the constraint of summing over solutions of min-sum equations in (17.9) we introduce two indicator functions (these are analogous to (16.21) and we abuse notation by using the same symbols)

$$\begin{cases} \Delta_{i\to a} = \mathbb{1}\big(E_{i\to a}(\cdot) = \sum_{b\in\partial i\setminus a} \hat{E}_{b\to i}(\cdot) - C_{i\to a}\big), \\ \hat{\Delta}_{a\to i} = \mathbb{1}\big(\hat{E}_{a\to i}(x_i) = \min_{\sim x_i}\big\{E_a(x_{\partial a}) + \sum_{j\in\partial a\setminus i} E_{j\to a}(x_j)\big\} - \hat{C}_{a\to i}\big). \end{cases} \tag{17.12}$$

Then, proceeding exactly as in (16.22)-(16.26) (and abusively using similar notations) the partition function (17.9) can be written in the factorized form

$$\Xi(y) = \sum_{\{E_{i\to a}(\cdot),\hat{E}_{a\to i}(\cdot)\}} \prod_i \psi_i \prod_a \psi_a \prod_{ia} \psi_{ia} \tag{17.13}$$

with

$$\psi_i = e^{-yE_i} \prod_{a\in\partial i} \Delta_{i\to a}, \qquad \psi_a = e^{-yE_a} \prod_{i\in\partial a} \hat{\Delta}_{a\to i}, \qquad \psi_{ia} = e^{+yE_{ia}} \tag{17.14}$$

where we recall that E_i, E_a and E_{ia} are the three contributions (17.4)-(17.6) to the Bethe energy functional. The factor graph of the model is the same as the one on the right of Figure 16.3.

17.2 Survey propagation equations and complexity of the energetic model

Now that we have formulated the energetic level-one model in the factor graph language (17.13) we can use our usual machinery to derive sum product equations and a Bethe free energy functional.

Survey propagation equations

The analysis is essentially identical to that of Section 16.5 and it is possible to reduce the sum-product equations on the new factor graph (Fig. 16.3) to a set of message passing equations with messages flowing on the edges of the original factor graph. Messages are distributions over energy costs $Q_{i\to a}(E_{i\to a})$ and $\hat{Q}_{i\to a}(\hat{E}_{a\to i})$. In the present context it is usual to call them *surveys* and the set of resulting equations linking them *survey propagation equations*. The later are analogous to (16.28),

$$\begin{cases} \widehat{Q}_{a\to i}(\widehat{E}_{a\to i}) \propto \sum_{\underline{E}} \hat{\Delta}_{a\to i} e^{-y(E_a - E_{ai})} \prod_{j\in\partial a\setminus i} Q_{j\to a}(E_{j\to a}) \\ Q_{i\to a}(E_{i\to a}) \propto \sum_{\underline{\widehat{E}}} \Delta_{i\to a} e^{-y(E_i - E_{ai})} \prod_{b\in\partial i\setminus a} \widehat{Q}_{b\to i}(\widehat{E}_{b\to i}). \end{cases} \tag{17.15}$$

We leave it as an exercise for the reader to show that the exponents in the "reweighting factors" have the following explicit expressions

$$\begin{cases} E_a - E_{ai} = \hat{C}_{a\to i} = \min_{x_{\partial a}} \left\{ E_a(x_{\partial a}) + \sum_{j\in\partial a\setminus i} E_{j\to a}(x_j) \right\} \\ E_i - E_{ai} = C_{i\to a} = \min_{x_i} \sum_{b\in\partial i\setminus a} \hat{E}_{b\to i}(x_i) \end{cases} \tag{17.16}$$

where $\hat{C}_{a\to i}$ and $C_{i\to a}$ are the "normalization constants" in the min-sum equations (17.2).

Bethe free energy and complexity of the energetic model

The survey propagation equations are stationnary point equations of a Bethe free energy functional. As usual on a tree this functional would be an exact expression for $n^{-1}\ln\Xi(y)$). Here we are not on a tree, but one may hope that the Bethe functional forms teh basis of an exact 1RSB like formula for $n^{-1}\mathbb{E}[\ln\Xi(y)]$ (when $n\to+\infty$) where the expectation is over the ensemble of random instances. This is believed to be the case in the applicatioj to K-SAT.

It should be clear to the reader (by now) that the expression for the relevant Bethe functional is analogous to (16.30)-(16.33), that is

$$\mathcal{F}_{\rm Bethe}(\underline{Q},\underline{\widehat{Q}};y) = \sum_{i\in V}\mathcal{F}_i + \sum_{a\in C}\mathcal{F}_a - \sum_{ai\in E}\mathcal{F}_{ai} \tag{17.17}$$

where

$$\mathcal{F}_i(\{\widehat{Q}_{b\to i}\}_{b\in\partial i}) = -y^{-1}\ln\Big\{\sum_{\underline{\widehat{\mu}}} e^{-yE_i}\prod_{b\in\partial i}\widehat{Q}_{b\to i}\Big\}, \tag{17.18}$$

$$\mathcal{F}_a(\{Q_{j\to a}\}_{j\in\partial a}) = -y^{-1}\ln\Big\{\sum_{\underline{\mu}} e^{-yE_a}\prod_{j\in\partial a} Q_{j\to a}\Big\}, \tag{17.19}$$

$$\mathcal{F}_{ai}(Q_{i\to a},\widehat{Q}_{a\to i}) = -y^{-1}\ln\Big\{\sum_{\mu,\widehat{\mu}} e^{-yE_{ai}} Q_{i\to a}\widehat{Q}_{a\to i}\Big\}. \tag{17.20}$$

In view of (17.10), the Bethe complexity for zero energy is found from

$$\Sigma_{\rm Bethe}(\underline{Q},\widehat{\underline{Q}}) = -\lim_{y\to+\infty} y\mathcal{F}_{\rm Bethe}(\underline{E},\widehat{\underline{E}};y) \tag{17.21}$$

17.3 Survey propagation for K-SAT instances

We use our usual notations introduced for the K-SAT problem. The alphabet is binary $x_i \in \{0,1\}$ and we switch to the spin language $s_i = (-1)^{x_i}$. Full edges (variable i appears un-negated in clause a) have $J_{ia} = +1$ and dashed edges (variable i appears negated in clause a) have $J_{ia} = -1$. Recall also that $\partial_{\pm i}$ is the set of constraints a such that $J_{ia} = \pm 1$. The cost functions E_a defining the Hamiltonian (17.1) are

$$E_a(s_{\partial a}) = \prod_{i\in\partial a} \frac{1}{2}(1 + J_{ia}s_i) \tag{17.22}$$

We keep in mind for later use that a constraint a is satisfied if for at least one node $i \in \partial a$ we have $s_i = -J_{ia}$. Similarly it is unsatisfied if for all $i \in \partial a$ we have $s_i = +J_{ia}$.

Energy costs $E_{i\to a}(s_i)$, $\hat{E}_{a\to i}(s_i)$ appearing in the min-sum equations are functions of a binary variable $s_i = \pm 1$ and normalized such that their minimum vanishes. It is not very hard to see that any such function can be parametrized as

$$E_{i\to a}(s_i) = |h_{i\to a}| + h_{i\to a}J_{ia}s_i, \qquad \hat{E}_{a\to i}(s_i) = |\hat{h}_{a\to i}| + \hat{h}_{a\to i}J_{ia}s_i \tag{17.23}$$

where $h_{i\to a}$, $\hat{h}_{a\to i}$ are for the moment arbitrary numbers ("fields"). With this parametrization the min-sum equations become (exercise)

$$\begin{cases} h_{i\to a} = \sum_{b\in\partial i\setminus a} J_{bi}J_{ai}h_{b\to i} \\ \hat{h}_{a\to i} = \prod_{j\in\partial a\setminus i}(1-\theta(h_{j\to a})) \end{cases} \tag{17.24}$$

where $\theta(u) = 0$, 1for $u \le 0$, $u > 0$ the Heaviside function. Obviously messages $h_{a\to i}$ live in the discrete alphabet $\{0,1\}$; but what about $h_{i\to a}$?. We notice in the second min-sum equation that only the sign of $h_{j\to a}$ really matters. In other words, introducing the "sign function" $\mathrm{sgn}(u) = -1,\ 0,\ +1$ for $u < 0$, $u = 0$, $u > 0$, (17.24) are equivalent to

$$\begin{cases} h_{i\to a} = \mathrm{sgn}\{\sum_{b\in\partial i\setminus a} J_{bi}J_{ai}\hat{h}_{b\to i}\} \\ \hat{h}_{a\to i} = \prod_{j\in\partial a\setminus i}(1-\theta(h_{j\to a})) \end{cases} \tag{17.25}$$

The message alphabet is thus entirely discrete, namely $h_{i\to a} \in \{-1,0,+1\}$ and $\hat{h}_{a\to i} \in \{0,1\}$, and this reduction is a crucial simplifying feature of the energetic cavity method.

Equations (17.25) are sometimes called "warning propagation" equations because the messages can be nicely interpreted as "warnings". For example $\hat{h}_{a\to i} = +1$ is equivalent to $E_{a\to i}(s_i = J_{ia}) = +2$, $E_{a\to i}(s_i = -J_{ia}) = 0$. Thus $\hat{h}_{a\to i} = +1$ is interpreted as a "warning": given the information a receives from $j \in \partial a \setminus i$ it warns i to satisfy it (i.e. set $s_i = -J_{ia}$) otherwise this would incur an energy cost. When $\hat{h}_{a\to i} = +0$, $E_{a\to i}(s_i = \pm J_{ia}) = 0$ so a warns i that both choices $s_i = \pm J_{ia}$ are indifferent and incur an energy cost. Similarly $\hat{h}_{i\to a} = -1$ is equivalent to $E_{i\to a}(s_i = J_{ia}) = 0$, $E_{i\to a}(s_i = -J_{ia}) = +2$; thus given the information i receives from $b \in \partial a \setminus i$ it warns a that it is forced to not satisfy it (i.e. $s_i = J_{ia}$). Finally for $\hat{h}_{i\to a} = 0$ we have $E_{i\to a}(s_i = \pm J_{ia}) = 0$ so i warns a that it can take a random value, and for $\hat{h}_{i\to a} = +1$ we have $E_{i\to a}(s_i = J_{ia}) = +2$, $E_{i\to a}(s_i = -J_{ia}) = 0$ and i warns a that it is forced to satisfy it. With these interpretations the second warning propagation equation in (17.25) expresses that if a receives a warning from at least one $j \in \partial a \setminus i$ that j can satisfy a (i.e. $h_{j\to a} = +1$) then a can warn i that it is free to take any value (i.e. $h_{a\to i} = 0$). The first equation expresses that i is forced not to satisfy a (i.e. $\hat{h}_{i\to a} = -1$) when i receives from $b \in \partial i \setminus a$ **complete the sentence properly**.

Since the essential object of interest is the zero-energy complexity (because it controls the zero temperature phase diagram) we are interested in the $y \to +\infty$ limit of the survey propagation equations (17.15). In this limit only terms such $C_{i\to a} = 0$ and $\hat{C}_{a\to i} = 0$ survive. In fact (for K-SAT) $\hat{C}a \to i = 0$ anyway so this does not yield an extra constraint in the sum. Indeed in (see (17.16)),

$$\hat{C}_{a\to i} = \min_{s_{\partial a}} \big\{ E_a(s_{\partial a}) + \sum_{j\in\partial a\setminus i} E_{j\to a}(s_j) \big\}$$

once we fix $s_j = J_{ja}\mathrm{sgn}h_{j\to a}$ for all $j \in \partial a$ı, we can still fix $s_i = -J_i a$ to achieve $\hat{C}_{a\to i} = 0$. For the other constant we have

$$\begin{aligned}
\hat{C}_{i\to a} &= \min_{s_i} \big\{ \sum_{b\in\partial a\setminus i} \hat{E}_{b\to i}(s_i) \big\} \\
&= \min_{s_i} \big\{ \sum_{b\in\partial a\setminus i} |\hat{h}_{b\to i}| + s_i \sum_{b\in\partial a\setminus i} J_{bi}\hat{h}_{b\to i} \big\} \\
&= \sum_{b\in\partial a\setminus i} |\hat{h}_{b\to i}| - |\sum_{b\in\partial a\setminus i} J_{bi}\hat{h}_{b\to i}|
\end{aligned}$$

where the last equality is obtained by setting $s_i = -J_{bi}$ to achieve the minimum. In order to satisfy $C_{i\to a} = 0$ we can allow $\hat{u}_{b\to i} = 1$ on a certain number of edges with the *same sign* and $\hat{u}_{b\to i} = 0$ on the remaining edges that have both signs.

We can now work out the explicit form of the survey propagation equations (17.15) for K-SAT instances in the limit $y \to +\infty$. Consider the surveys $\hat{Q}_{a\to i}(\hat{h}_{a\to i})$. These must satisfy $\hat{Q}_{a\to i}(1) + \hat{Q}_{a\to i}(0) = 1$ and we will keep track

only of $\hat{Q}_{a\to i}(1)$. We have

$$\begin{aligned}\hat{Q}_{a\to i}(1) &= \sum_{\underline{h}} \mathbb{1}\big(1 = \prod_{j\in\partial a\setminus i}(1-\theta(h_{j\to a}))\big) \prod_{j\in\partial a\setminus i} Q_{j\to a}(h_{j\to a}) \\ &= \prod_{j\in\partial a\setminus i} Q_{j\to a}(-1) \end{aligned} \tag{17.26}$$

This equation says that the probability that a warns i that it must satisfy it is equal to the probability that all $j \in \partial a \setminus i$ warn that they are forced to un-satisfy a (here these are probabilities over the set of fixed points of the warning propagation equations which have zero Bethe free energy and the warnings are treated as independent). Consider now the three other surveys $Q_{i\to a}(-1), Q_{i\to a}(0), Q_{i\to a}(+1)$, namely the probabilities that i warns a it must un-satisfy it, is free to take any value, must satisfy it. For $Q_{i\to a}(0)$ we have

$$\begin{aligned} Q_{i\to a}(0) &\propto \sum_{\underline{\hat{h}}} \mathbb{1}\big(0 = \mathrm{sgn}\big\{\sum_{b\in\partial i\setminus a} J_{bi}J_{ai}\hat{h}_{b\to i}\big\}\big) \\ &\quad \times \mathbb{1}\big(0 = \sum_{b\in\partial a\setminus i} |\hat{h}_{b\to i}| - |\sum_{b\in\partial a\setminus i} J_{bi}\hat{h}_{b\to i}|\big) \prod_{b\in\partial i\setminus a} \hat{Q}_{b\to i}(\hat{h}_{b\to i}) \\ &= \sum_{\underline{\hat{h}}} \mathbb{1}\big(0 = \mathrm{sgn}\big\{\sum_{b\in\partial i\setminus a} J_{bi}J_{ai}\hat{h}_{b\to i}\big\}\big) \\ &\quad \times \mathbb{1}\big(0 = \sum_{b\in\partial a\setminus i} |\hat{h}_{b\to i}|\big) \prod_{b\in\partial i\setminus a} \hat{Q}_{b\to i}(\hat{h}_{b\to i}) \\ &= \prod_{b\in\partial i\setminus a} \hat{Q}_{b\to i}(0) \\ &= \prod_{b\in\partial i\setminus a} (1-\hat{Q}_{b\to i}(1)) \end{aligned} \tag{17.27}$$

In words the probability that i warns it is free to take any value equals the probability that all constraints $b \in \partial i \setminus a$ warn they are satisfied. We leave it to the reader to work out in detail the two other cases,

$$\begin{aligned} Q_{i\to a}(1) &\propto \sum_{\underline{\hat{h}}} \mathbb{1}\big(1 = \mathrm{sgn}\Big\{\sum_{b\in\partial i\setminus a} J_{bi}J_{ai}\hat{h}_{b\to i}\Big\}\Big) \\ &\quad \times \mathbb{1}\big(0 = \sum_{b\in\partial a\setminus i} |\hat{h}_{b\to i}| - |\sum_{b\in\partial a\setminus i} J_{bi}\hat{h}_{b\to i}|\big) \prod_{b\in\partial i\setminus a} \hat{Q}_{b\to i}(\hat{h}_{b\to i}) \\ &= \Big\{\prod_{b\in\partial i\setminus a: J_{bi}\neq J_{ai}} (1-\hat{Q}_{b\to i}(1))\Big\}\Big\{1 - \prod_{b\in\partial i\setminus a: J_{bi}=J_{ai}} (1-\hat{Q}_{b\to i}(1))\Big\}, \end{aligned} \tag{17.28}$$

$$
\begin{aligned}
Q_{i\to a}(-1) \propto & \sum_{\underline{\hat{h}}} \mathbb{1}\big(-1 = \mathrm{sgn}\big\{ \sum_{b\in\partial i\setminus a} J_{bi}J_{ai}\hat{h}_{b\to i}\big\}\big) \\
& \times \mathbb{1}\big(0 = \sum_{b\in\partial a\setminus i} |\hat{h}_{b\to i}| - |\sum_{b\in\partial a\setminus i} J_{bi}\hat{h}_{b\to i}|\big) \prod_{b\in\partial i\setminus a} \hat{Q}_{b\to i}(\hat{h}_{b\to i}) \\
= & \Big\{ \prod_{b\in partial i\setminus a: J_{bi}=J_{ai}} (1-\hat{Q}_{b\to i}(1)) \Big\} \Big\{ 1 - \prod_{b\in\partial i\setminus a: J_{bi}\neq J_{ai}} (1-\hat{Q}_{b\to i}(1)) \Big\}
\end{aligned}
\tag{17.29}
$$

Equation (17.28) says the probability i warns that it satisfies a equals the probability that i is forced to satisfy clauses with $J_{bi} = J_{ai}$ and free with respect to clauses with $J_{bi} \neq J_{ai}$. Similarly equation (17.29) says the probability i warns that it un-satisfies a equals the probability that i is forced to satisfy clauses with $J_{bi} \neq J_{ai}$ and free with respect to clauses with $J_{bi} = J_{ai}$.

The four equations (17.26)-(17.29) are the survey propagation equations for K-SAT when $y \to +\infty$ and can be used in the satisfiable phase. The three equations (17.27), (17.28), (17.29) must be normalized so that $Q_{i\to a}(-1) + Q_{i\to a}(0) + Q_{i\to a}(1) = 1$. A bit of thought shows that (17.26) is already correctly normalized (hence the equality line in the first line).

17.4 Energetic complexity and satisfiability threshold

With a bit more work (see exercises) one can derive from (17.21) the Bethe formula for the energetic complexity at zero energy. The resulting expression is

$$
\Sigma_{\mathrm{Bethe}}(\underline{Q},\underline{\hat{Q}}) = \sum_i \Sigma_i + \sum_a \Sigma_a - \sum_{ai} \Sigma_{ai} \tag{17.30}
$$

with

$$
\Sigma_i = \log\Big\{ \prod_{a\in\partial_+ i} (1-\hat{Q}_{a\to i}(1)) + \prod_{a\in\partial_- i} (1-\hat{Q}_{a\to i}(1)) - \prod_{a\in\partial i} (1-\hat{Q}_{a\to i}(1)) \Big\} \tag{17.31}
$$

$$
\Sigma_a = \log\Big\{ 1 - \prod_{j\in\partial a} Q_{j\to a}(-1) \Big\} \tag{17.32}
$$

$$
\Sigma_{ai} = \log\big\{ 1 - Q_{i\to a}(-1)\hat{Q}_{a\to i}(1) \big\} \tag{17.33}
$$

It is easy to check that the stationary point condition for $\Sigma_{\mathrm{Bethe}}(\underline{Q},\underline{\hat{Q}})$ is equivalent to (17.26)-(17.29).

Let us now give the 1RSB expression for the average energetic complexity. We will be brief since we have already encountered this type of formulation several times by now. Fix a trial distribution $\mathcal{Q}^{\mathrm{u}}(\cdot)$ for a real valued random variable

$Q^{\rm u}$ and take $K-1$ iid copies $Q^{\rm u}_1,\cdots,Q^{\rm u}_{K-1}$. Define the random variable

$$\hat{Q}=\prod_{k=1}^{K-1}Q^{\rm u}_k \tag{17.34}$$

with induced distribution $\hat{\mathcal{Q}}(\cdot)$. Let p and q two Poisson$(\alpha K/2)$ integers and $p+q$ iid copies of $\hat{Q}$, denoted $\hat{Q}^+_1,\cdots\hat{Q}^+_p$ and $\hat{Q}^-_1,\cdots\hat{Q}^-_q$, as well as K iid copies of $Q^{\rm u}$ denoted $Q^{\rm u}_1,\cdots,Q^{\rm u}_K$. Consider the random variable

$$\begin{aligned}\Sigma=&\log\Big\{\prod_{\ell=1}^{p}(1-\hat{Q}^+_\ell)+\prod_{\ell=1}^{q}(1-\hat{Q}^-_\ell)-\prod_{\ell=1}^{p}(1-\hat{Q}^+_\ell)\prod_{\ell=1}^{q}(1-\hat{Q}^-_\ell)\Big\}\\&+\log\Big\{1-\prod_{k=1}^{K}Q^{\rm u}_k\Big\}-\log\Big\{1-Q^{\rm u}_1\hat{Q}^+_1\Big\}\end{aligned} \tag{17.35}$$

The 1RSB energetic complexity is a function of the constraint density given by

$$\Sigma_{\rm 1RSB}(\alpha)=\sup_{\mathcal{Q}(\cdot)}\mathbb{E}[\Sigma] \tag{17.36}$$

where the expectation is over $\underline{Q}^{u,\pm}$, $\underline{\hat{Q}}$, and p, q.

Of course (17.34) is the distributional form of the corresponding equation for specific instances (17.26). The other three distributional forms of (17.27), (17.28), (17.29) are nothing else than the stationary point conditions for the variational expression in the 1RSB formula (17.36). These take the form

$$Q^*\overset{\rm distr}{=}\frac{1}{N}\prod_{\ell=1}^{p}(1-\hat{Q}^+_\ell)\prod_{\ell=1}^{q}(1-\hat{Q}^-_\ell), \tag{17.37}$$

$$Q^{\rm u}\overset{\rm distr}{=}\frac{1}{N}\Big\{\prod_{\ell=1}^{p}(1-\hat{Q}^+_\ell)\Big\}\Big\{1-\prod_{\ell=1}^{q}(1-\hat{Q}^-_\ell)\Big\}, \tag{17.38}$$

$$Q^{\rm s}\overset{\rm distr}{=}\frac{1}{N}\Big\{\prod_{\ell=1}^{q}(1-\hat{Q}^+_\ell)\Big\}\Big\{1-\prod_{\ell}^{p}(1-\hat{Q}^-_\ell)\Big\} \tag{17.39}$$

with a normalisation N such that $Q^*+Q^{\rm u}+Q^{\rm s}=1$.

To compute $\Sigma_{\rm 1RSB}(\alpha)$ one solves equations (17.34) and (17.37)-(17.39) by a population dynamics method and deduces the average of Σ in (17.35). There is always (for all α) a trivial fixed point $\hat{\mathcal{Q}}(\hat{Q})=\delta(0)$ and $\mathcal{Q}^{\rm s}(Q^{\rm s})=\mathcal{Q}^{\rm u}(Q^{\rm u})=\delta(0)$, $\mathcal{Q}^*(Q^*)=\delta(Q^*-1)$. The *survey propagation threshold* $\alpha_{\rm SP}$ is by definition the maximum value of α such that the trivial solution is unique. For $\alpha<\alpha_{\rm SP}$ we immediately see from (17.36) that the 1RSB energetic complexity vanishes. In this regime we expect the Bethe energy has a single zero energy minimum. When $\alpha>\alpha_{\rm SP}$ the complexity is given by the non-trivial fixed point. We find a strictly positive complexity for $\alpha_{\rm SP}<\alpha<\alpha_s$, meaning that the Bethe energy has exponentially many (in n) minima of vanishing energy. At α_s the complexity becomes negative which means that the zero energy minima disappear. The

K	$\alpha_{\rm SP}$	α_s
3	3.93	4.267
4	8.30	9.93

Table 17.1 Survey propagation and satisfiability thresholds predicted by the energetic cavity method.

threshold α_s is identified as the satisfiability threshold. Table 17.1 gives the numerical values of the thresholds for the first few values of K.

For large K we can make an analytical asymptotic analysis and derive expansions for the survey propagation and satisfiability thresholds. Here we briefly derive the leading term of these expansions but with more work it is possible to compute higher order terms (see the notes). First we set $\hat{X} = \ln(1-\hat{Q})$. Equation (17.38) becomes

$$Q^{\rm u} \stackrel{\rm distr}{=} \frac{e^{\sum_{\ell=1}^{p}\hat{X}_\ell^+} - e^{\sum_{\ell=1}^{p}\hat{X}_\ell^+}e^{\sum_{\ell=1}^{q}\hat{X}_\ell^-}}{e^{\sum_{\ell=1}^{p}\hat{X}_\ell^+} + e^{\sum_{\ell=1}^{q}\hat{X}_\ell^-} - e^{\sum_{\ell=1}^{p}\hat{X}_\ell^+}e^{\sum_{\ell=1}^{q}\hat{X}_\ell^-}}.$$

In the large K limit the law of large numbers (or Wald's theorem) implies that the sums in the exponentials all concentrate around $(\alpha K/2)\mathbb{E}[\hat{X}]$ (here $\alpha K/2$ is the average of p and q). Therefore, setting $\mathbb{E}[\hat{X}] = \hat{x}$, $Q^{\rm u}$ concentrates on $(1 - e^{\frac{\alpha K}{2}\hat{x}})/(2 - e^{\frac{\alpha K}{2}\hat{x}})$. Applying again the law of large numbers to (17.34) we get

$$\hat{x} = \ln\left\{1 - \left(\frac{1 - e^{\frac{\alpha K}{2}\hat{x}}}{2 - e^{\frac{\alpha K}{2}\hat{x}}}\right)^{K-1}\right\} \approx -\left(\frac{1 - e^{\frac{\alpha K}{2}\hat{x}}}{2 - e^{\frac{\alpha K}{2}\hat{x}}}\right)^{K-1}.$$

The last approximation is justified a posteriori: when we calculate the order of magnitude of the fixed point solutions we find that $(\cdots)^{K-1} = O(2^{-K})$ (for example this is evident for the trivial fixed point $\hat{x} = 0$). It is convenient to set $\hat{y} = -\frac{\alpha K}{2}\hat{x}$ and scale the constraint density as $\alpha = 2^K\hat{\alpha}$ so the last equation becomes

$$\hat{y} = \hat{\alpha}\left(\frac{e^{\hat{y}} - 1}{e^{\hat{y}} - \frac{1}{2}}\right)^{K-1}. \tag{17.40}$$

The survey propagation integral equations (17.34), (17.37)-(17.39) have been reduced to a simple one dimensional fixed point equation over $\mathbb{R}$. Note $\hat{y} = -\frac{\alpha K}{2}\mathbb{E}[\ln(1-\hat{Q})]$ so absence of warnings $\hat{Q} = 0$ means $\hat{y} = 0$ and presence of warnings mean $\hat{y} > 0$. Proceeding with similar approximations for the complexity we get in the large K limit

$$\Sigma_{\rm 1RSB}(\alpha) \approx \sup_{\hat{y}}\left\{\ln(2e^{-\hat{y}} - e^{-2\hat{y}}) - 2K\hat{\alpha}\left(\frac{e^{\hat{y}} - 1}{2e^{\hat{y}} - 1}\right) + \hat{y}\frac{e^{\hat{y}} - 1}{2e^{\varphi} - 1}\right\} \tag{17.41}$$

One can check for consistency that the stationary points of (17.41) are given by the fixed point equation (17.40). It remains to find solutions of the fixed point

equation and deduce the complexity. Evidently (17.40) has a trivial fixed point $\varphi = 0$ for all $\hat{\alpha}$ and the corresponding complexity vanishes. To find the non-trivial fixed points we plot the curve $\hat{\alpha}(\varphi)$ directly obtained from (17.40). This is a convex function with a unique minimum at $\varphi_{\rm SP} \approx \ln(\frac{1}{2}K \ln K)$, $\hat{\alpha}_{\rm SP}(K) \approx \frac{\ln K}{K}$. Fixing $\hat{\alpha} > \hat{\alpha}_{\rm SP}(K)$ we find two fixed point solutions, one (locally) stable and the other unstable. The stable solution equals $\varphi_{\rm st} \approx K\hat{\alpha} + \varphi_{\rm SP}$ for $\varphi >> \varphi_{\rm SP}$. This is the solution which yields a maximal complexity for $\hat{\alpha} > \hat{\alpha}_{\rm SP}(K)$. We find $\Sigma_{\rm 1RSB}(\alpha) \approx \ln 2 - \hat{\alpha}$ for $\hat{\alpha} > \hat{\alpha}_{\rm SP}$ which implies $\hat{\alpha}_s \approx \ln 2$. Summarizing, to leading order we find

$$\alpha_{\rm SP} \approx 2^K \frac{\ln K}{K}, \qquad \alpha_s \approx 2^K \ln 2$$

It is noteworthy that the 1RSB prediction for the satisfiability threshold has been recently turned into a rigorous result for K large enough (but finite). The proof uses the interpolation bounds to deduce an upper bound on α_s and ideas from the energetic cavity method for a lower bound.

THEOREM 17.1 *Let* $\alpha_s \equiv \sup\{\alpha | \Sigma_{\rm 1RSB}(\alpha) \geq 0\}$ *where* $\Sigma_{\rm 1RSB}(\alpha) \geq 0\}$ *is defined through* (17.34), (17.35), (17.36). *Let* $\mathbb{P}[-]$ *the uniform probability distribution over formulas* $F \in \mathcal{F}(n, K, \alpha)$. *We have* $\lim_{n\to+\infty} \mathbb{P}[\text{F is satisfiable} = \mathbb{1}(\alpha < \alpha_s)$.

In particular, this theorem finally settles the question of the existence of a sharp threshold in the thermodynamic limit discussed in Chapter 1. It is in fact rather surprizing that the existence question has been settled only through a complete characterisation of the threshold.

We end this section by mentioning that the energetic cavity method can be developed for the spatially coupled K-SAT ensemble introduced in Chapter **??**. The thresholds depend on the size of the coupling window w, the length L of the chain and of course K. One finds a threshold saturation phenomenon

$$\lim_{w\to+\infty} \lim_{L\to+\infty} \alpha_{\rm SP}(K, w, L) = \alpha_s(K),$$

(here α_s the usual satisfiability threshold of the standard uncoupled ensemble). Besides, using an interpolation method (in the spirit of the one that proves the existence of the free energy in Chapter **??**) one can rigorously prove for all w

$$\lim_{L\to+\infty} \alpha_s(K, w, L) = \alpha_s(K).$$

17.5 Survey propagation guided decimation algorithm

In chapter 9 we showed how to use used belief propagation marginals to guide a decimation algorithm for finding solutions of specific instances of K-SAT formulas. We can proceed analogoulsy and develop a new decimation algorithm where

the decisions (fix and decimate a variable) are taken thanks to survey propagation marginals. This turns out to be a fruitful idea which allows to find solutions close to the satisfiability threshol well inside a region constraint densities that belief propagation guided decimation cannot probe.

Recall that from the min-sum message passing equations (17.2) we can compute the "marginal energy costs" (17.7). The latter are vectors (here the alphabet is discrete) with non-negative components with at least one which vanishes. For K-SAT the alphabet is furthermore binary, so we can always parametrize the marginal energy costs as

$$E_i(s_i) = |h_i| + h_i s_i \tag{17.42}$$

Replacing the parametrization of messages (17.23) in (17.7) it follows that

$$h_i = \sum_{a \in \partial i} h_{a \to i} J_{ia} \tag{17.43}$$

When $h_i > 0$ we have $E_i(+1) > 0$, $E_i(-1) = 0$ and there is no energy cost if $s_i = -1$ (or $x_i = 1$); when $h_i < 0$ we have $E_i(+1) = 0$, $E_i(-1) > 0$ and there is no energy cost if $s_i = +1$ (or $x_i = 0$); and when $h_i = 0$ we have $E_i(+1) = E_i(-1) = 0$ and there is no cost for $s_= \pm 1$ (or $x_i = 0, 1$).

From survey propagation we can compute the fraction of zero energy minima of the Bethe neergy (zero energy solutions of the min-sum equations) with a marginal energy cost parametrized by h_i. This fraction that we call $Q_i(h_i)$ is just the marginal of the level-one energetic model and can be computed from messages $\hat{Q}_{a \to i}(\hat{h}_{a \to i})$ by the usual rules of message passing. The rules for $y \to +\infty$ can easily be guessed from the interpretation of min-sum messages in terms of messages (their formal deduction is left to the resder as an exercise). A variable is free to take any value $s_i = \pm 1$ when it receives no warnings $\hat{h}_{a \to i} = 0$ from all neighboring constraints, so

$$Q_i(h_i = 0) \propto \prod_{a \in \partial i} (1 - \hat{Q}_{a \to i}(1)) \tag{17.44}$$

A variable should take the value $s_i = -1$ when it receives at least one warning $\hat{h}_{a \to i} = 1$ from constraints $a \in \partial_+ i$ (such that $J_{ai} = +1$) and no warning $\hat{h}_{a \to i} = 0$ from constraints $a \in \partial_- i$ (such that $J_{ai} = -1$), thus

$$Q_i(h_i > 0) \propto \left\{ 1 - \prod_{a \in \partial i_+} (1 - \hat{Q}_{a \to i}(1)) \right\} \left\{ \prod_{a \in \partial i_-} (1 - \hat{Q}_{a \to i}(1)) \right\}. \tag{17.45}$$

Similarly we shoul have $s_i = +1$ when there is at least one warning $\hat{h}_{a \to i} = 1$ from constraints $a \in \partial_- i$ (such that $J_{ai} = -1$) and no warning $\hat{h}_{a \to i} = 0$ from constraints $a \in \partial_+ i$ (such that $J_{ai} = 1$),

$$Q_i(h_i < 0) \propto \left\{ 1 - \prod_{a \in \partial i_-} (1 - \hat{Q}_{a \to i}(1)) \right\} \left\{ \prod_{a \in \partial i_+} (1 - \hat{Q}_{a \to i}(1)) \right\} \tag{17.46}$$

The survey propagation guided decimation algorithm can now be formulated.

The main idea is to compute the survey messages and marginals. The variable

Algorithm 4: SPGD algorithm

1. Take a fixed instanceof size n.
2. Run $t_{\max}$ iterations of SP message passing equations starting from a random initial condition $\hat{Q}_{a\to i} \in [0,1]$. If the iterations do not converge, return fail. If they converge Compute the biases $|Q_i(1) - Q_i(0)|_0$.
3. If all biases are smaller than some specified samll number $\delta > 0$ call BP decimation or Walksat.
4. Else fix a variable with the largest bias to $x_i = +1$ if $Q_i(h_i > 0) > Q_i(h_i < 0)$ or $x_i = 0$ if $Q_i(h_i > 0) < Q_i(h_i < 0)$.
5. Decimate the variable and reduce the formula. Return to 2 until all variables are eliminated.

with the maximal bias $|Q_i(h_i > 0) - Q_i(h_i < 0)|$ is fixed appropriately, namely $s_i = \mathrm{sgn}(Q_i(h_i < 0) - Q_i(h_i > 0)$, and the formula reduced. Reducing teh formula means that the satisfied constraints are eliminated and the unsatisfied ones are shortened. The process is iterated as long as there is a large bias. When no such bias appears we call BPGD or Walksat. Note that the algorithm is effective only for $\alpha > \alpha_{\mathrm{SP}}$. Indeed for $\alpha < \alpha_{\mathrm{SP}}$ iterations are always attracted by the trivial fixed point $\hat{Q}_{a\to i}(1) = 0$ so $Q_i(h_i = 0) = 1$ and $Q_i(h_i < 0) = Q_i(h_i > 0) = 0$ and BPGD or Walksat are used. Note that the algorithm is effective only for $\alpha > \alpha_{\mathrm{SP}}$. Indeed for $\alpha < \alpha_{\mathrm{SP}}$ iterations are always attracted by the trivial fixed point $\hat{Q}_{a\to i}(1) = 0$ so $Q_i(h_i = 0) = 1$ and $Q_i(h_i < 0) = Q_i(h_i > 0) = 0$. In practice when we start the algorithm at $\alpha_{\mathrm{SP}} < \alpha < \alpha_s$ we find biases and the formula gets reduced. At some point the density of the reduced formula is smaller than some sort of effective survey propagation threshold of a "reduced ensemble of formulas" and there is no more bias (and we call BPGD or Walksat).

The complexity of this algorithm is $O(t_{\max} n^2)$. This is given by the product of the complexity of each run of SP $O(t_{\max} n)$ times the number of recursions or decimation steps $O(n)$. For a variant of the algorithm decimates a fraction fn with $0 < f < 1$ of variables at a time (the fraction fn of variables with largest biases) one has to recurse $O(n/fn)$ times so that the complexity is reduced to $O(t_{\max} n/f)$. Also, experimentally one finds that convergence of SP messages is reached for $t_{\max} = \log n$ even for densities larger than the satisfiability threshold.

Figure 17.3 shows the empirical probabilities of success and convergence over 100 instances of size $n = 10^4$, 1000 iterations and convergence criterion $\delta = 10^{-2}$ where δ is the difference between two successive messages). In practice we observe that SPGD finds solutions of large instances for $\alpha_{\mathrm{SPGD}}(K = 3) \approx 4.25$ and $\alpha_{\mathrm{SPGD}}(K = 4) \approx 9.6$. The effectiveness of the algorithm is apparent when one compares these values with the one's found with BPGD also with the SAT-UNSAT threshold.

Decimation algorithms are in general very hard to analyse except when the

— figure —

Figure 17.3 Empirical probabilities of success and convergence over 100 instances of size $n = 10^4$, 1000 iterations and convergence criterion $\delta = 10^{-2}$.

decimation step preserves some uniform randomness property of the ensemble of formulas, as is the case for unit clause propagation. To date it is not clear how to rigorously analyse the survey propagation guided decimation algorithm. However an interesting general result of Sudan and Gamarnik hints at some necessary condition that this decimation algorithm satisfies in order to be successful beyond the dynamical or survey propagation threshold. Roughly speaking the result of Sudan and Gamarnik states that, when there exist exponentially many clusters of solutions, any decimation rule based on a computational tree of depth $O(1)$ cannot succeed at finding solutions. Survey propagation guided decimation evades this result because presumably one explores a neighbourhood of size $t_{\max} = O(\ln n)$ (we say "presumably" because numerically this is not so obvious to assess). This suggests that survey propagation messages should "converge" (remain smaller than $\delta << 1$) only after $O(\ln n)$ iterations, not $O(1)$ iterations as is the case with belief propagation.

17.6 Notes

Problems

17.1 1
17.2 2

Bibliography

Aji, S. M. & McEliece, R. J. (2000), 'The generalized distributive law', *IEEE Trans. Inform. Theory* **46**(2), 325–343.

Amic, C. & Luck, J. (1995), 'Zero-temperature error-correcting code for a binary symmetric channel', *J. Phys. A: Math. Gen.* **28**, 135–47.

Baxter, R. (1982), *Exactly Solved Models in Statistical Mechanics*, Academic Press.

Berlekamp, E. R. (1984), *Algebraic Coding Theory*, revised edn, Aegean Park Press, Walnut Creek, CA, USA.

Berrou, C., Glavieux, A. & Thitimajshima, P. (1993), Near Shannon limit error-correcting coding and decoding, *in* 'Proc. of ICC', Geneva, Switzerland, pp. 1064–1070.

Bethe, H. A. (1935), 'Statistical theory of superlattices', *Proc. Roy. Soc.* **A150**, 552–75.

Beurling, A. (1938), 'Sur les intégrales de fourier absolument convergentes et leur application à une transformation fonctionnelle', *Proc Scandi Math Congr, Helsinki, Finland* .

Blahut, R. E. (2003), *Algebraic Codes for Data Transmission*, Cambridge Univ. Press.

Boettcher, C. (1973), *Theory of Electric Polarization (Second edition). Dielectrics in Static Fields.*, Elsevier.

Bollabás, B. (1998), *Modern Graph Theory*, Springer Verlag, New York, NY, USA.

Bolthauzen, E. (2014), 'An iterative construction of solutions of the tap equations for the sherringtonkirkpatrick model', *Communications in Mathematical Physics* **325**, 333366.

Bragg, W. & Williams, E. (1934), 'The effect of thermal agitation on atomic arrangements in alloys', *Proceedings of the Royal Society A* **145**, 699–730.

Brillouin, L. (1956), *Science and Information Theory*, Academic Press.

Brush, S. (1983), *Statistical Physics and the Atomic Theory of Matter: From Boyle and Newton to Landau and Onsager*, Princeton series in physics, Princeton University Press.

Brush, S. G. (1967), 'History of the Lenz-Ising model', *Reviews of Modern Physics* **39**, 883–893.

Candès, E. (2006*a*), Compressive sampling, *in* 'Proc. Int. Congress Math.'.

Candès, E. (2006*b*), 'The restricted isometry property and its implications for compressive sensing', *C. R. Acad Sci, Sér 1* **346(9-10)**, 589–592.

Candès, E., Romberg, J. & Tao, T. (2006), 'Stable signal recovery from incomplete and inaccurate measurements', *Comm. Pure Appl. Math.* **59(8)**, 1207–1223.

Candès, E. & Tao, T. (2006), 'Decoding by linear programming', *IEEE Trans. Inform. Theory* **51(12)**, 4203–4215.

Caratheodory, C. (1907), '¨ber den variabilitätsbereich der koeffizienten von potenzreihen, die gegebene werte nicht annehmen', *Math Ann* **64**, 95–115.

Chaikin, P. & Lubensky, T. (2007), *Principles of condensed matter physics*, Cambridge University Press.

Chen, S. & Donoho, D. (1995), 'Examples of basis pursuit', *Proceedings Wavelet Applications in Signal and Image Processing III* .

Cook, S. A. (1971), The complexity of theorem proving procedures, *in* 'Proc. of STOC', pp. 151–158.

de Almeida, J. R. L. & Thouless, D. J. (1978), 'Stability of the sherrington- kirkpatrick solution of a spin glass model', *J. Phys. A* **11**, 983.

Dembo, A. & Montanari, A. (2010), 'Ising models on locally tree-like graphs', *Ann. Appl. Probab.* **20 (2)**, 565–592.

Ding, J., Sly, A. & Sun, N. (2014), 'Proof of the satisfiability threshold for large k', *arXiv:1411.0650 [math.PR]* .

Dobrushin, R. (1965), 'Existence of a phase transition in the two-dimensional and three-dimensional ising models', *Dokl. Akad. Nauk SSSR 160 1046–1048 (Russian); translated as Soviet Physics Dokl. 10 (1965) 111-113* .

Donoho, D. (2006), 'Compressed sensing', *IEEE Trans. Inform. Theory* **52(4)**, 1289–1306.

Edwards, S. F. & Anderson, P. W. (1975), 'Theory of spin glasses', *Journal of Physics F: Metal Physics* **5**, 965–974.

Eldar, Y. & Kutyniok, G., eds (2012), *Compressed sensing*, Cambridge University Press.

Fisher, K. & Hertz, J. (1991), *Spin Glasses*, Cambridge Studies in Magnetism, Cambridge University Press.

Forney, Jr., G. D. (2001), 'Codes on graphs: Normal realizations', *IEEE Trans. Inform. Theory* **47**(2), 520–548.

Frey, B. (1998), *Graphical Models for Machine Learning and Digital Communication*, Adaptive Computation and Machine Learning series, MIT press, Cambridge.

Friedgut, E. (1999), 'Sharp thresholds of graph properties, and the k-sat problem', *Journal of the American Mathematical Society* **12**, 1017–1054.

Fu, Y. & Anderson, P. (1986), 'Applications of statistical mechanics to np-complete problems in combinatorial optimization', *Journal of Physics* **A19**, 1605.

Gallager, R. G. (1962), 'Low-density parity-check codes', *IRE Trans. Inform. Theory* **8**, 21–28.

Gallager, R. G. (1963), *Low-Density Parity-Check Codes*, MIT Press, Cambridge, MA, USA.

Gallavotti, G. (1999), *Statistical Mechanics, A Short Treatise*, Texts and Monographs in Physics, Springer.

Garey, M. R. & Johnson, D. S. (1979), *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, CA, USA.

Giurgiu, A., Macris, N. & Urbanke, R. (2016), 'Spatial coupling as a proof technique and three applications', *IEEE Trans. Inf. Theor.* **62**(10), 5281 – 5295.

Goff, S. L., Glavieux, A. & Berrou, C. (1994), Turbo-codes and high spectral efficiency modulation, *in* 'Proc. of ICC', New Orleans, LA, pp. 645–649.

Goldenfeld, N. (1993), *Lectures on phase transitions and the renormalization group*, Addison Wesley.

Griffiths, R. (1964), 'Peierls proof of spontaneous magnetization in a two-dimensional ising ferromagnet', *Physical Review* .

Guerra, F. (2001), 'Sum rules for the free energy in the mean field spin glass model', *Fields Institute Communications* **30**, 161.

Guerra, F. & Toninelli, F. L. (n.d.), 'Quadratic replica coupling in the sherrington-kirkpatrick mean field spin glass model', *Journal of Mathematical Physics* **43**, 3704–3716.

Hopfield, J. (1982), 'Neural networks and physical systems with emergent collective computational abilities', *Proc. Natl. Acad. Sci. USA* **79**(8), 2554–2558.

Huang, K. (1987), *Statistical Physics*, John Wiley and Sons.

Jaynes, E. T. (1957), 'Information theory and statistical mechanics', *Physical Review* **106**, 620–630.

Jordan, M. (1999), *Learning in Graphical Models*, Adaptive Computation and Machine Learning series, MIT press, Cambridge.

Kabashima, Y. (2003), 'A cdma multiuser detection algorithm on the basis of belief propagation', *Journal of Physics A: Mathematical and general* **36**, 11111?1112.

Kabashima, Y. & Saad, D. (1998), 'Belief propagation vs. tap for decoding corrupted messages', *Europhysics Letters* **44**, 668–674.

Kabashima, Y. & Saad, D. (2004), 'Statistical mechanics of low-density parity check codes', *J. Phys. A* **37**, R1–R43. Invited paper.

Kac, M. (1968), Mathematical mechanisms of phase transitions, *in* 'Statistical Mechanics of Phase Transitions and Superfluidity (Ed. M. Chr etilin, E. Gross, S. Dresser)'.

Kadanoff, L. (2009), 'More is the same; phase transitions and mean field theories', *Journal of Statistical Physics* **137**, 777–797.

Kim, J. H. & Pearl, J. (1983), A computational model for causal and diagnostic reasoning in inference systems, *in* 'IJCAI', pp. 190–193.

Kirkpatrick, S. & Selman, B. (1994), 'Critical behavior in the satisfiability of random boolean expressions', *Science* **264**, 1297–1301.

Kschischang, F. R., Frey, B. J. & Loeliger, H.-A. (2001), 'Factor graphs and the sum-product algorithm', *IEEE Trans. Inform. Theory* **47**(2), 498–519.

Landau, L. (1937), *Phys. Z. Sow. 11, 26545. English Translation: Collected Papers of Landau (1965)*, Pergamon Press.

Lebowitz, J. L. & Penrose, O. (1966), 'Rigorous Treatment of the Van Der Waals-Maxwell Theory of the Liquid-Vapor Transition', *Journal of Mathematical Physics* **7**, 98–113.

Lin, S. & Costello, Jr., D. J. (2004), *Error Control Coding*, 2nd edn, Prentice Hall, Englewood Cliffs, NJ, USA.

Loeliger, H.-A. (2004), 'An introduction to factor graphs', *Signal Process.* **21**(2), 28–41.

Loyd, S. (2000), 'Ultimate physical limits to computation', *Nature* **406**, 1047–1054.

Luby, M., Mitzenmacher, M., Shokrollahi, A. & Spielman, D. A. (2001), 'Improved low-density parity-check codes using irregular graphs', *IEEE Trans. Inform. Theory* **47**(2), 585–598.

Luby, M., Mitzenmacher, M., Shokrollahi, A., Spielman, D. A. & Stemann, V. (1997), Practical loss-resilient codes, *in* 'Proc. of the 29th annual ACM Symposium on Theory of Computing', pp. 150–159.

Lukke, H. (1999), 'The origins of the sampling theorem', *IEEE Communications magazine* **37(4)**, 1–3.

Ma, S. (1976), *Modern Theory of Critical Phenomena*, Benjamin Reading, Massachusetts.

MacKay, D. J. C. (1999), 'Good error correcting codes based on very sparse matrices', *IEEE Trans. Info. Theory* **45**(2), 399–431.

MacKay, D. J. C. (2003), *Information Theory, Inference, and Learning Algorithms*, Cambridge Univ. Press.

MacKay, D. J. C. & Neal, R. M. (1996), 'Near Shannon limit performance of low density parity check codes', *Electron. Lett.* **32**(18), 1645–1646. Reprinted in *Electron. Lett.*, **33** (1997), pp. 457–458.

Macris, N. (2007*a*), 'Griffith-kelly-sherman correlation inequalities: A useful tool in the theory of error correcting codes', *Information Theory, IEEE Transactions on* **53**, 664–683.

Macris, N. (2007*b*), 'Sharp bounds on generalized exit functions', *IEEE Transactions on Inform. Theory* **53**(7), 2365 – 2375.

Macris, N. & Korada, S. (2010), 'Tight bounds on the capacity of binary input random cdma systems', *Information Theory, IEEE Transactions on* **56**, 5590–5613.

Mao, Y. & Kschischang, F. R. (2005), 'On factor graphs and the Fourier transform', *IEEE Trans. Inform. Theory* **51**, 1635–1649.

Maxwell, J. C. (1875), 'On the dynamical evidence of the molecular constitution of bodies', *Nature* **11**, 357–359, 374–377.

McCoy, B. & Wu, T. (1973), *The Two-Dimensional Ising Model*, Harvard University Press, Cambridge Massachusetts.

Méasson, C., Montanari, A., Richardson, T. J. & Urbanke, R. (2009), 'The generalized area theorem and some of its consequences', *IEEE Trans. Inf. Theor.* **55**(11), 4793–4821.

Méasson, C., Montanari, A. & Urbanke, R. (2004), Maxwell's construction: The hidden bridge between maximum-likelihood and iterative decoding, *in* 'Proc. of the IEEE Int. Symposium on Inform. Theory', Chicago, IL, USA, p. 225.

Méasson, C., Montanari, A. & Urbanke, R. (2008), 'Maxwell's construction: The hidden bridge between maximum-likelihood and iterative decoding', *IEEE Transactions on Inform. Theory* **54**, 5277 – 5307.

Mézard, M. & Montanari, A. (2009), *Information, Physics, and Computation*, Oxford University Press.

Mézard, M., Parisi, G. & Virasoro, A. (1987*a*), *Spin Glass Theory and Beyond*, World Scientific.

Mézard, M., Parisi, G. & Virasoro, M. A. (1987*b*), *Spin-Glass Theory and Beyond*, World Scientific Publ.

Mitchell, D., Selman, B. & Levesque, H. (1992), Hard and easy distributions for sat problems, *in* 'Proc. of the Tenth Natl. Conf. on Artificial Intelligence', pp. 459–465.

Monasson, R. & al (1999), 'Computational complexity from characteristic phase transitions', *Nature* **400**, 133–137.

Monasson, R. & Zecchina, R. (1997), 'Statistical mechanics of the random k-sat model', *Phys. Rev. E* **56**, 1357–70.

Montanari, A. (2012), Graphical models concepts in compressed sensing, *in* 'Compressed Sensing: theory and Applications', Cambridge, pp. 394–438.

Moore, C. & Mertens, S. (2011), *The Nature of Computation*, Oxford University Press.

Morita, T. (1979), 'Variational principle for the distribution function of the effective field for the random ising model in the bethe approximation', *Physica A: Statistical Mechanics and its Applications* **98**, 566–572.

Nishimori, H. (1980), 'Exact results and critical properties of the ising model with competing interactions', *J. Phys. C: Solid State Phys.* **13**(21), 4071–4076.

Nishimori, H. (1993), 'Optimal decoding for error-correcting codes', *J. Phys. Soc. Japan* **62**, 29735.

Nishimori, H. (2001), *Statistical Physics of Spin Glasses and Information Processing: An Introduction*, Oxford Science Publ.

Onsager, L. (1936), 'Electric moments of molecules in liquids', *Journal of the American Chemical Society* **58(8)**, 14861493.

Onsager, L. (1944), 'Crystal statistics. i. a two-dimensional model with an order-disorder transition', *Physical Review, Series II* **65**, 117–149.

Onsager, L. (1952), 'The spontaneous magnetization of a two-dimensional ising model', *Physical Review, Series II* **85**, 808816.

Opper, M. & Winter, O. (1996*a*), 'A mean field algorithm to bayes learning in feedforward networks', *Neural Information Processing Systems (NIPS)* .

Opper, M. & Winter, O. (1996*b*), 'A mean field approach to bayes learning in feedforward networks', *Physical review letters* **76**, 1964.

Panchenko, D. (2013), *The SherringtonKirkpatrick Model*, Springer Monographs in Mathematics.

Papadimitriou, C. & Steiglitz, K. (1982), *Combinatorial Optimization: Algorithms and Complexity*, Englewood Cliffs, N.J:Prentice Hall, Inc.

Parisi, G. (1980), 'The order parameter for spin glasses: a function on the interval 0-1', *J. Phys. A: Math. Gen.* **13**, 1101–1112.

Pearl, J. (1982), Reverend bayes on inference engines: A distributed hierarchical approach, *in* 'Proceedings of the Second National Conference on Artificial Intelligence AAAI-82', Pittsburgh, PA. Menlo Park, California, pp. 133–136.

Pearl, J. (1988), *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann Publ., San Mateo, CA, USA.

Peierls, R. (1936), 'The spontaneous magnetization of a two-dimensional ising model', *Proc. Camb. Philos. Soc.* **32**, 477.

Penrose, O. (2005), *Foundations of Statistical Mechanics: a deductive treatment*, Dover.

Richardson, T. & Urbanke, R. (2007), *Modern Coding Theory*, Cambridge University Press.

Ruelle, D. (1969), *Statistical Mechanics: Rigorous Results*, W. A. Benjamin Inc, New York.

Ruján, P. (1993), 'Finite temperature error-correcting codes', *Phys. Rev. Lett.* **70**(19), 2968–2971.

Schroeder, D. (2000), *An Introduction to Thermal Physics*, Princeton series in physics, Addison-Wesley.

Shafer, G. R. & Shenoy, P. P. (1990), 'Probability propagation', *Ann. Math. Art. Intel.* **2**, 327–352.

Shannon, C. E. (1948), 'A mathematical theory of communication', *Bell System Tech. J.* **27**, 379–423, 623–656.

Sherrington, D. & Kirkpatrick, S. (1975), 'Solvable model of a spin-glass', *Physical Review Letters* **35 (26)**, 1792–1796.

Simon, B. (1993), *The Statistical Mechanics of Lattice Gases*, Princeton series in physics, Princeton University Press.

Sourlas, N. (1989), 'Spin-glass models as error-correcting codes', *Nature* **339**(29), 693–695.

Sourlas, N. (1994), 'Spin glasses, error-correcting codes and finite-temperature decoding', *Europhys. Lett.* **25**, 159–164.

Stanley, H. (1971), *Mean field theory of magnetic phase transitions. Introduction to phase transitions and critical phenomena*, Oxford University Press.

Talagrand, M. (2000), 'Replica symmetry breaking and exponential inequalities for the sherringtonkirkpatrick model', *Annals of Probability* **28(3)**, 1018–1062.

Talagrand, M. (2003), *Spin Glasses: A Challenge for Mathematicians: Cavity and Mean Field Models*, Springer, New York, NY, USA.

Talagrand, M. (2011), *Mean Field Models for Spin Glasses, volumes I and II*, A series of Modern Surveys in Mathematics, Srpinger.

Tanaka, T. (2002), 'A statistical-mechanics approach to large-system analysis of cdma multiuser detectors', *Information Theory, IEEE Transactions on* **48**, 2888–2910.

Tanner, R. M. (1981), 'A recursive approach to low complexity codes', *IEEE Trans. Inform. Theory* **27**(5), 533–547.

Thompson, C. (1988), *Classical Equilibrium Statistical Mechanics*, Clarendon Press.

Thouless, D. J., Anderson, P. W. & Palmer, R. G. (1977), 'Solution of a 'solvable model of a spin glass", *Philosophical Magazine* **35**, 593–601.

Tibshirani, R. (1996), 'Regression shrinkage and selection with the lasso', *J. Roy. Stat. Soc.* **B 58**, 267–288.

Toninelli, F. L. (n.d.), 'About the almeida-thouless transition line in the sherrington-kirkpatrick mean-field spin glass model,', *Europhysics letters* **60**, 764–767.

Van der Waals, J. D. (1873), *Over de continuiteit van den gas-en vloeisoftoestand*, Leiden: Sijthoff.

Van der Waals, J. D. & Rowlinson, J. S. (1988), *On the Continuity of the Gaseous and Liquid States - Edited with an introduction by J. S. Rowlinson*, Dover.

Weiss, P. (1907), 'L'hypothse du champ molculaire et la proprit ferromagntique', *J. Phys. Theor. Appl.* **6 (1)**, 661690.

Wiberg, N., Loeliger, H.-A. & Kötter, R. (1995), 'Codes and iterative decoding on general graphs', *Eur. Trans. Telecomm. (ETT)* **6**, 513–526.

Yedidia, J. S., Weiss, Y. & Freeman, W. T. (2003), 'Understanding belief propagation and its generalizations', *Exploring Artificial Intelligence in the New Millennium* (ISBN 1558608117), 239–236. Chap. 8.