This is to certify that the
thesis entitled

COMPLEX NETWORK PROBLEMS IN PHYSICS
COMPUTER SCIENCE AND BIOLOGY

presented by

RADU IONUT COJOCARU

has been accepted towards fulfillment
of the requirements for the

Ph.D. degree in Physics and Astronomy

Major Professor's Signature

12/16/2006

Date

MSU is an Affirmative Action/Equal Opportunity Institution

COMPLEX NETWORK PROBLEMS IN PHYSICS, COMPUTER SCIENCE AND BIOLOGY

By

Radu Ionut Cojocaru

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Physics and Astronomy

2006

ABSTRACT

COMPLEX NETWORK PROBLEMS IN PHYSICS, COMPUTER SCIENCE AND BIOLOGY

By

Radu Ionut Cojocaru

There is a close relation between physics and mathematics and the exchange of ideas between these two sciences are well established. However until few years ago there was no such a close relation between physics and computer science. Even more, only recently biologists started to use methods and tools from statistical physics in order to study the behavior of complex system. In this thesis we concentrate on applying and analyzing several methods borrowed from computer science to biology and also we use methods from statistical physics in solving hard problems from computer science.

In recent years physicists have been interested in studying the behavior of complex networks. Physics is an experimental science in which theoretical predictions are compared to experiments. In this definition, the term prediction plays a very important role: although the system is complex, it is still possible to get predictions for its behavior, but these predictions are of a probabilistic nature. Spin glasses, lattice gases or the Potts model are a few examples of complex systems in physics.

Spin glasses and many frustrated antiferromagnets map exactly to computer science problems in the NP-hard class defined in Chapter 1. In Chapter 1 we discuss a common result from artificial intelligence (AI) which shows that there are some problems which are NP-complete, with the implication that these problems are difficult to solve. We introduce a few well known hard problems from computer science (Satisfiability, Coloring, Vertex Cover together with Maximum Independent Set and Number Partitioning) and then discuss their mapping to problems from physics.

In Chapter 2 we provide a short review of combinatorial optimization algorithms and their applications to ground state problems in disordered systems.We discuss the cavity method initially developed for studying the Sherrington-Kirkpatrick model of spin glasses. We extend this model to the study of a specific case of spin glass on the Bethe lattice at zero temperature and then we apply this formalism to the K-SAT problem defined in Chapter 1.

The phase transition which physicists study often corresponds to a change in the computational complexity of the corresponding computer science problem. Chapter 3 presents phase transitions which are specific to the problems discussed in Chapter 1 and also known results for the K-SAT problem. We discuss the replica method and experimental evidences of replica symmetry breaking.

The physics approach to hard problems is based on replica methods which are difficult to understand. In Chapter 4 we develop novel methods for studying hard problems using methods similar to the message passing techniques that were discussed in Chapter 2. Although we concentrated on the symmetric case, cavity methods show promise for generalizing our methods to the un-symmetric case.

As has been highlighted by John Hopfield, several key features of biological systems are not shared by physical systems. Although living entities follow the laws of physics and chemistry, the fact that organisms adapt and reproduce introduces an essential ingredient that is missing in the physical sciences. In order to extract information from networks many algorithm have been developed. In Chapter 5 we apply polynomial algorithms like minimum spanning tree in order to study and construct gene regulatory networks from experimental data. As future work we propose the use of algorithms like min-cut/max-flow and Dijkstra for understanding key properties of these networks.

ACKNOWLEDGMENTS

A Ph.D. dissertation is far from being a one person work, although finally it comes to a single name on the cover. There are so many people I am thankful for getting this far.

First, and foremost, I would like to thank my adviser and probably the most influential person on my future career, Phillip Duxbury, who introduced me to combinatorial optimization problems, taught me almost everything I know about them, guided my questions and helped me to develop some ideas I run into. His generosity and his character to discuss physics in particular and anything else in general made our discussions some of the best in my life and it is very hard to find someone else to thank for something like that.

I would like to thank Wolfgang Bauer, Jonathan Hall, Carlo Piermarocchi and Stuart Tessmer who, as members of my advisory committee, offered precious suggestions and criticism.

The Department of Physics provided generous financial support in my first two years, which made my life in East Lansing comfortable. To that end, I thank our graduate secretary Debbie Simmons, CMP secretary Cathy Cords and secretary to the Chair of our department, Lisa Ruess.

I would like to thank my office mates Chris Farrow, Chip Fay, Jiwu Liu, Dan Olds and one of my former office mate, now Dr. Erin McGarrity, for many interesting discussions from the physics realm and not only.

I surely couldn't be here without my dear physics professors from Romania: Andrei Medar, Vasile Constantin or Raluca Rebedea (the last two passed away but not in my heart).

Definitely the most important person that I want to thank is my wife Nadia. Her support and incredible understanding during the graduate years and especially in the last one year made this work possible. I would like to thank my son Alex who, although he is only three years and half, he understood that sometimes instead of spending evenings together I had to come at school and work. Also many thanks to my one month and a half son Paul who teaches me something new every day. Being successful on both plans, family and school, is not an easy job so this is why I also thank *totea* Lida who traveled thousands of miles to help us, my wife, kids and I.

I would be remiss if I did not mention the people with whom I enjoyed my free time since I was almost a kid. These include my friends Cristian Cocheci, Marian Maruta and Claudiu Soare.

Last but not least, I would also like to thank my mother and my sister Georgiana for keeping me sane and for encouragements especially when I didn't have my best days.

This work is dedicated to my **father**.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

Introduction

You are a chief of protocol for the embassy ball. The crown prince instructs you either to invite Peru or to exclude Qatar. The queen asks you to invite either Qatar or Romania or both. The king, in a spiteful mood, wants to snub either Romania or Peru or both. Is there a guest list that will satisfy the whims of the entire royal family?

1.1 Computational Complexity for Physicists

This contrived little puzzle is an instance of a problem that lies near the root of theoretical computer science. It is called the satisfiability problem (or *SAT*) and it was the first member of the well know class called the *nondeterministic polynomial* or NP class (in the next section we define a few problems that are in NP, the SAT problem being one of them).

Computational complexity is a part of computer science which deals with classifying problems according to the computational resources required to solve them. There are two basic classes: the polynomial class (or the P class) and the nondeterministic polynomial (or NP) class.

The NP term was introduced in the early $1970's$ and describes the abyss of inherent intractability that programmers face as they try to solve larger and more complex problems. Problems that belong to the nondeterministic-polynomial NP class seem intrinsically hard, but after 35 years of attempts nobody proved that they are necessarily difficult. Characteristic for the NP class is that if we can guess the answer, than we can check its correctness in polynomial time (*i.e.* efficiently). For the above example, the checking procedure is straight forward: given a proposed labeling, we just have to substitute the specified invite (or *true*) and do not invite (or *false*) for the three variables (P from Peru, Q from Qatar and R from Romania) and make sure that the formula is true. There are many examples of problems with competing objectives (like the one above) that appear in both physics (e.g. the spin glasses (*SG* problem)) and in computer science (e.g. scheduling or planning). These kinds of problems with many degrees of freedom have been analyzed for more than half a century by the computer science community and more recently by physicists. The SAT problem is a member of an even more exclusive class called NP-complete. Completeness is a key property to the entire set of NP-complete problems: if a polynomial time algorithm could be found for any NP - complete problem, then the algorithm could be adapted to all problems in NP. The SAT problem was the first problem shown by Stephen Cook [46] to belong to the NP-complete class.

For the other class, the polynomial class, we say that a problem belongs to the P class if it can be solved by a polynomial-time algorithm. Unfortunately, the converse assumption is not true: just because no one found a polynomial-time algorithm to solve a problem doesn't mean that the problem is not in class P. As Hayes [58] states: *It remains possible (though unlikely) that we are simply attacking them by clumsy methods, and if we could dream up a clever algorithm they would all turn out to be easy*. There are thousands of other NP-complete problems that are suspended in this computational limbo, and in the next section we introduce some of them.

1

1

A

a

i

T

ta

n

w

in

as

A

ca

th

alg

eit

me

hav

we

NP

the

1.2 Examples of key problems from the NP-complete class

1.2.1 The SAT problem

An instance of the satisfiability problem is defined in terms of N Boolean variables, and a set of M constraints between them, where each constraint takes the special form of a *clause*. A clause is the logical *OR* of some variables or their negations. The notation that we use is the following: a variable x_i, with $i \in \{1, 2, \cdots, N\}$, takes value in $\{0, 1\}$ with 1 corresponding to *true* and 0 corresponding to *false*; the negation of x_i is $\overline{x}_i \equiv 1 - x_i$. A variable or its negation is called a literal, denoted with l_p^i (*i.e* l_p^i denotes either x_i or $\overline{x}_i$). A clause a, with $a \in \{1, 2, \cdots, M\}$, involving K_a variables is a constraint which forbids exactly one among the 2^{K_a} possible assignments to these K_a variables. The clause a is written as $C_a = l_p^1 \vee l_p^2, \cdots, \vee l_p^K$. An instance of the satisfiability problem can be written as:

$$F = C_1 \wedge C_2 \wedge \cdots \wedge C_M \tag{1.1}$$

called a *conjunctive normal form (CNF)*. Given the formula F, the question is whether there exists an assignment of the variables x_i, such that the formula F is true. An algorithm solving the satisfiability problem must be able, given the formula F, to either answer YES (the formula is then said to be *SAT* and provide such an assignment), or to answer NO in which case the formula is called *UNSAT*.

The restriction of the satisfiability problem by requiring that all the clauses in F have the same length $K_a = K$, is called the *K-satisfiability* problem (or K-SAT). As we already mentioned the satisfiability problem was the first problem shown to be NP-complete. For $K_a \leq 2$ the problem is solved in polynomial time but for $K_a \geq 3$ the problem belongs to the NP-complete class.

An optimization problem is associated to the decision version of satisfiability: given a formula F, one is asked to find an assignment which violates the smallest number of clauses. This is called the MAX-SAT problem.

1.2.2 The Coloring Problem

The Graph Coloring problem is a well known problem in combinatorics and in statistical physics. The problem is simply stated but is very difficult to solve either analytically or numerically. Given a graph, or a lattice, and given a number q of available colors, the problem consists in finding a coloring of vertices such that no edge has the two ending vertices of the same color. The possibility of finding such a solution, depends on the way the graph is constructed and also on the number of colors. The minimally needed number of colors is the *chromatic number* of the graph.

In modern computer science, graph coloring is taken as one of the most widely used benchmarks for the evaluation of algorithm performance. The interest in coloring stems from the fact that many real-world combinatorial optimization problems have component sub-problems which can be easily represented as coloring problems (for example, a classical application is the scheduling of registers in the central processing unit of computers). The q-coloring problem of random graphs is an active field of research in discrete mathematics and represents the natural evolution of the percolation theory initiated by Erdös and Rényi in the 50's [33].

One point of contact between computer science and random graph theory arises from the observation that for large random graphs, there exists a critical average connectivity beyond which the graphs become uncolorable with probability going to one as the graph size goes to infinity. The precise value of the critical connectivity depends on the number of allowed colors and on the ensemble of random graphs under consideration. Graphs generated close to, but below their critical connectivity are hard to color.

1.2.3 Vertex Cover and Maximum Independent Set

In order to have a more intuitive understanding for defining the vertex cover (VC) problem we will use an example in the same way we did for defining the satisfiability problem. Imagine that a director of a museum situated in a large park with numerous paths wants to put guards on crossroads to observe any path, but in order to economize costs the director has to use as few guards as possible. Let N be the number of crossroads and let $X \leq N$ be the number of guards. Then there are C_X^N possibilities of placing the guards, but most configurations will lead to unobserved paths. Deciding whether there exists any perfect solution, or finding one, can take a time that grows exponentially with N.

The mathematical formulation of the VC problem can be stated as follows: Consider an undirected graph $G = (V, E)$ with N vertices $i \in V = 1, 2, \cdots, N$ (the crossroads in our example) and edges $(i, j) \in E \subset V \times V$ (the paths in the considered example). A vertex cover is a subset $V_{vc} \subset V$ of vertices such that for all edges $(i, j) \in E$ there is at least one of its endpoints i or j in V_{vc} (the path is observed). We call the vertices that are in V_{vc} *covered*, whereas the vertices in its complement $V \setminus V_{vc}$ are called *uncovered*. The decision version is whether a VC of fixed cardinality exists or not.

Maximum independent set is trivially related to the minimum VC. First we define the independent set as a subset of vertices which are pairwise disconnected in the graph. Any set $V \setminus V_{vc}$ thus forms an independent set, and maximal independent sets are complementary to minimal vertex cover.

1.2.4 Maximum Cut

In order to define the Max Cut we first need to define the notion of a *cut*. In graph theory, a cut is a partition of the vertices of a graph into two sets. More formally, let $G(V,E)$ denote a graph. A cut is a partition of the vertices V into two sets S and T. Any edge $(i,j) \in E$ with $i \in S$ and $j \in T$ (or $i \in T$ and $j \in S$, in case of a directed graph) is said to be crossing the cut and it is called a *cut edge*. The size of a cut is the total number of edges crossing the cut. In weighted graphs, the size of the cut is defined to be the sum of weights of the edges crossing the cut. If we consider non-negative weights $w_{ij} = w_{ji}$ on the edges $(i,j) \in E$, the maximum cut problem Max Cut is that of finding the set of vertices S that maximizes the weight of the edges in the cut (S,T) that is, the weight of the edges with one endpoint in S and the other in $T \equiv \overline{S}$. For simplicity, we usually set $w_{ij} = 0$ for $(i,j) \notin E$ and denote the weight of a cut by $w(S,T) = \sum_{i \in S, j \in T} w_{ij}$. The Max Cut problem is one of the Karp's original NP-complete problems [64] and has long been known to be NP-complete even if the problem is unweighted; that is, if $w_{ij} = 1$ for all $(i,j) \in E$ [46]. The Max Cut problem is solvable in polynomial time for some special classes of graphs (e.g. if the graph is planar [54]). Besides its theoretical importance, the MAX CUT problem has applications in circuit layout design and statistical physics [8].

1.2.5 Number Partitioning

The number partition problem (NPP) is another problem of Garey and Johnson's [46] six basic NP-complete problems that lie at the heart of NP-completeness. It is defined as follows: given a sequence of positive real numbers $a_1, a_2, ..., a_N$ the NPP consists of partitioning them into two disjoint sets A_1, A_2 such that the difference $|\sum_{a_j \in A_1} a_j - \sum_{a_j \in A_2} a_j|$ is minimized. Despite its simplicity, the NPP was shown to belong to the NP-complete class.

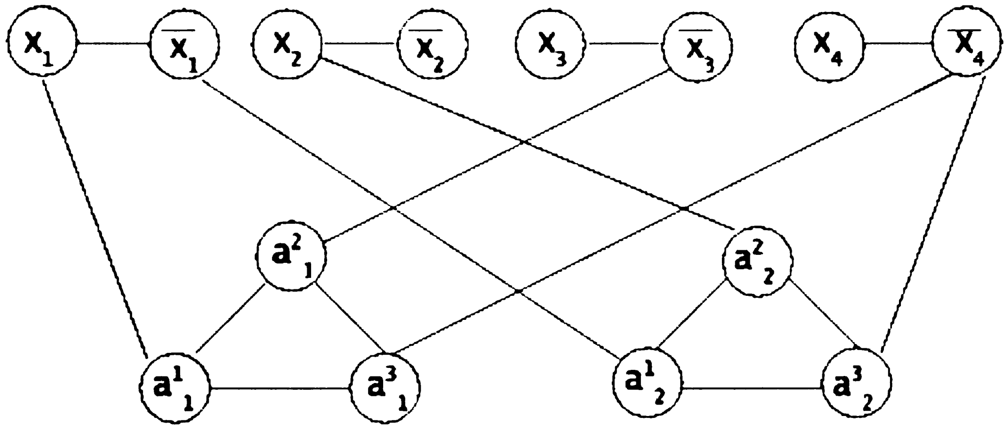


Figure 1.1: Vertex Cover instance resulting from the $3 - SAT$ instance

The decision version of NPP is: given a fixed k determine if there is a partition of $A = A_1 \bigcup A_2$ such that

$$|\sum_{a_j \in A_1} a_j - \sum_{a_j \in A_2} a_j| \leq k \qquad (1.2)$$

The four problems that we defined in this section (SAT, Coloring, VC and NPP) are among the six celebrated and most used problems in proving NP-completeness (the other two are the traveling salesman problem TSP and Hamiltonian circuit HC).

1.3 Mappings

In this section we use the reduction methodology to provide NP-completeness proofs for some of the problems discuses above.

1.3.1 Mapping 3-SAT to VC

The proof of the NP-completeness of the vertex cover problem works by reducing $3 - SAT$ to VC in polynomial time. First, we show $VC \in NP$: It is very easy to decide for a given subset V' of vertices, whether all edges are covered, *i.e.* whether V' is a vertex cover, by just iterating over all edges. Hence, it remains to show that $3 - SAT$ is polynomially reducible to VC. Let $F = C_1 \wedge, \cdots, C_m$ be a $3 - SAT$ formula with variables $X = x_1, \cdots, x_n$ and $C_p = l_p^1 \vee l_p^2 \vee l_p^3$ for all p where each literal is a variable ($l_p^i = x_j$) or a negated variable ($l_p^i = \overline{x}_j$). We have to create a graph G and a threshold K, such that G has a VC of size lower than or equal to K, iff F is satisfiable. For this purpose, we set:

(i) $V_1 \equiv \{v_1, \overline{v}_1, \cdots, v_n, \overline{v}_n\}$ with $(\mid V_1 \mid = 2n)$ and $E_1 \equiv \{(v_1, \overline{v}_1), (v_2, \overline{v}_2), \cdots, (v_n, \overline{v}_n)\}$, *i.e.* for each variable occurring in F we create a pair of vertices and an edge between them. To cover the edges in E_1, we have to include at least one vertex per pair in the covering set. In this part of the graph, each cover corresponds to an assignment of the variables with the following idea behind it: If variable $x_i = 1$, then v_i should be covered, while if $x_i = 0$ then $\overline{v_i}$ is to be covered.

(ii) For each clause in F we introduce three vertices connected in a triangle: $V_2 \equiv \{a_1^1, a_1^2, a_1^3, a_2^1, a_2^2, a_2^3, \cdots, a_m^1, a_m^2, a_m^3\}$ and $E_2 \equiv \{(a_1^1, a_1^2), (a_1^2, a_1^3), (a_1^3, a_1^1), (a_2^1, a_2^2), (a_2^2, a_2^3), (a_2^3, a_2^1), \cdots, (a_n^1, a_n^2), (a_n^2, a_n^3), (a_n^3, a_n^1)\}$. For each clause, we have to include at least two vertices in a VC. In a cover of minimum size, the uncovered vertex corresponds to a literal which is satisfied.

(iii) Finally, for each position i in a clause p, vertex a_p^i is connected with the vertex representing the literal l_p^i appearing at that position of the clause: $E_3 \equiv \{a_p^i, v_j \mid p = 1, \cdots, m;\ i = 1,2,3\ \ if\ \ l_p^i = x_j\} \cup \{a_p^i, \overline{v}_j \mid p = 1, \cdots, m;\ i = 1,2,3\ \ if\ \ l_p^i = \overline{x}_j\}$ Hence, E_3 contains edges each connecting one vertex from V_1 with one vertex from V_2.

(iv) The graph G is the combination of the above introduced vertices and edges: $G = (V, E),\ \ V = V_1 \cup V_2,\ \ E = E_1 \cup E_2 \cup E_3$.

(v) For a SAT formula, the size of the minimum vertex cover con-

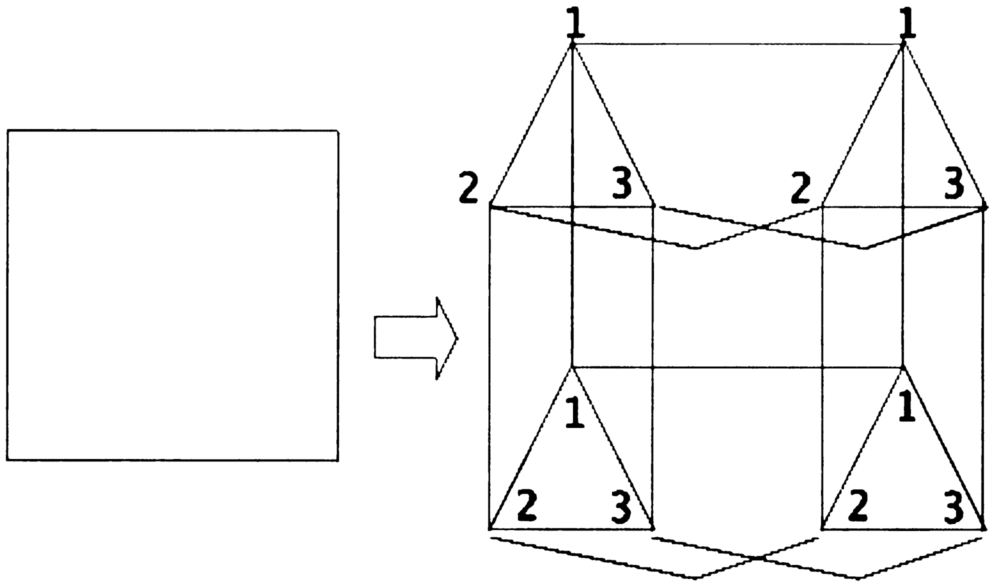


Figure 1.2: Vertex Cover instance resulting from the 3-Coloring instance

structed has cardinality $n + 2m$. As a small example we consider $F = (x_1 \vee \overline{x}_3 \vee \overline{x_4}) \wedge (\overline{x}_1 \vee x_2 \vee \overline{x}_4)$. The resulting graph $G(V, E)$ is displayed in Fig. 1.1.

1.3.2 Mapping 3-Coloring to VC

In Fig.(1.2) the mapping of the VC problem to 3-Coloring problem is illustrated. For each vertex of the square we introduce a sub-lattice with size 3 (*i.e.* a triangle); if we would like to color the square with q colors, then at each vertex of the square we would introduce a clique of size q. Each of these triangles (or cliques) correspond to one color. In order to ensure that two neighbors are never colored the same, we connect these two nodes with an edge. We continue this process for each of the four cliques/triangles and if we can find a MIS with size equal to the number of nodes in the lattice, then our original square is colorable. If not, the cost function (energy) is equal to the number of uncolorable sites.

1.4

One
vari
miz
ues
(ma
cal
star
exa
vap
alth
pro
mo
ext

1.4

In
spi
var
que
the

1.4 Mapping of NP-complete problems to statistical physics

One of the main goals for an optimization problem is to find the configuration of variables to minimize (maximize) a multi-variable function. A *combinatorial optimization problem* corresponds to the case when the variables take only discrete values under some combinatorial constraints. The function that we want to minimize (maximize) $f(x_1, x_2, ..., x_n)$ is called the cost function. The principal goal of statistical mechanics is to understand the macroscopic properties of many-body systems starting from the knowledge of interactions between microscopic elements. As an example we can consider the case of water which can exist in three different states: vapor (gas), water (liquid) or ice (solid) which look very different from each other although the microscopic elements are the same molecules of H_2O. Macroscopic properties of these three phases are quite different from each other because intermolecular interactions drastically change the macroscopic behavior according to external conditions like for example temperature or pressure.

1.4.1 The SAT problem

In order to map the $K - SAT$ problem onto random diluted systems we introduce spin variables $S_i = 1$ if the Boolean variable x_i is *true* and $S_i = -1$ if the Boolean variable x_i is *false*. The structure of the clauses is taken into account by an $M \times M$ quenched matrix $C_{l_i} = -1(1)$ if $x_i(\overline{x}_i)$ belongs to clause l and 0, otherwise. Then, the energy cost function (the number of violated clauses) is given by:

$$E[C,S] = \sum_{l=1}^{M} \delta(\sum_{i=1}^{N} C_{l_i} S_i - K) \tag{1.3}$$

sub
sym
nu
the
is i
var
erg
act
las
ion
ne
erg
the
poi
wh

tho
ter
In
the
wh

1.4

Fro
sta
For

subject to the constraints $\sum_{l=1}^{M} C_{l_i} S_i = K$ and $\sum_{i=1}^{N} C_{l_i}^2 = K, \quad l = 1 \ldots M$. The symbol δ denotes the delta function. The energy cost turns out to be equal to the number of violated clauses in that the quantity $\sum_{i=1}^{N} C_{l_i} S_i$ equals K if and only if all the Boolean variables in the clause l take the values opposite to the desired ones, *i.e* if the clause itself is false. The constraints ensure that the number of Boolean variables in any clause is exactly K. From a physical point of view, the K-SAT energy function is similar to the Hamiltonian of spin glasses. These systems, characterized by a frozen-in structural disorder, have been intensively studied in the last twenty five years. Spin glasses are materials weakly diluted with magnetic ions. The random positions of the ions induce random (in sign and strength) magnetic interactions. The lack of homogeneity results in an extremely complex energy landscape. In particular, the enormous number of meta-stable states makes the low-temperature behavior very unusual and interesting from a fundamental point of view. In the $K = 3$ case, the disorder is induced by the random clauses which make the problem more and more frustrated as the ration M/N increases.

The ground state properties of the cost function given by Eq.(3.8) will reflect those of K-SAT ($E_{GS} = 0$) and MAX-K-SAT ($E_{GS} > 0$). In Eq.(3.8), K may be interpreted as the number of *neighbors* to which each spin is coupled inside a clause. In Chapter 3 we study the ground state properties of the cost function (3.8) using the replica approach and in Chapter 4 we develop a novel geometrical technique which turns out to be much simpler than the replica technique.

1.4.2 Coloring

From the physics point of view, the q-coloring problem corresponds to the ground state configuration of a Potts anti-ferromagnetic with q-state variables [114], [12]. For most lattices such a system is frustrated and displays all the equilibrium and out-of-equilibrium features of spin glasses. The Hamiltonian is,

$$E = \sum_{i,j} b_{ij} \delta_{x_i x_j} \tag{1.4}$$

where $b_{ij} = 1$ (0) if an edge is present (absent) between sites i and j. The Potts anti-ferromagnetic is an example of a physical system with geometrical frustration [114]. Optimizing the color configuration with q colors is equivalent to finding the ground state of the q-state Potts anti-ferromagnetic on the same graph.

1.4.3 Spin Glasses

Spin was first discovered in the context of the emission spectrum of alkali metals. In 1924 Pauli introduced what he called a *two-valued quantum degree of freedom* associated with the electron in the outermost shell. This allowed him to formulate the Pauli exclusion principle, stating that no two electrons can share the same quantum numbers. Spin glasses (also called amorphous magnets) are magnetic substances in which the interaction among the spins is sometimes ferromagnetic, sometimes anti-ferromagnetic.

The rigorous formulation of the Ising spin glass problem is the following: we have N variables s_i where s_i can take only two values: $+1$ or -1 (Ising spin); for a given set of J_{ij} we are interesting in minimizing the function:

$$H_J\{s\} = -\sum_{i>j} J_{ij} s_i s_j \tag{1.5}$$

The function H is called the cost function or the Hamiltonian (or energy). From the point of view of complexity theory this problem is NP-complete [7], which means that very likely there is no algorithm that can find the minimum (called *the ground state*) in polynomial time.

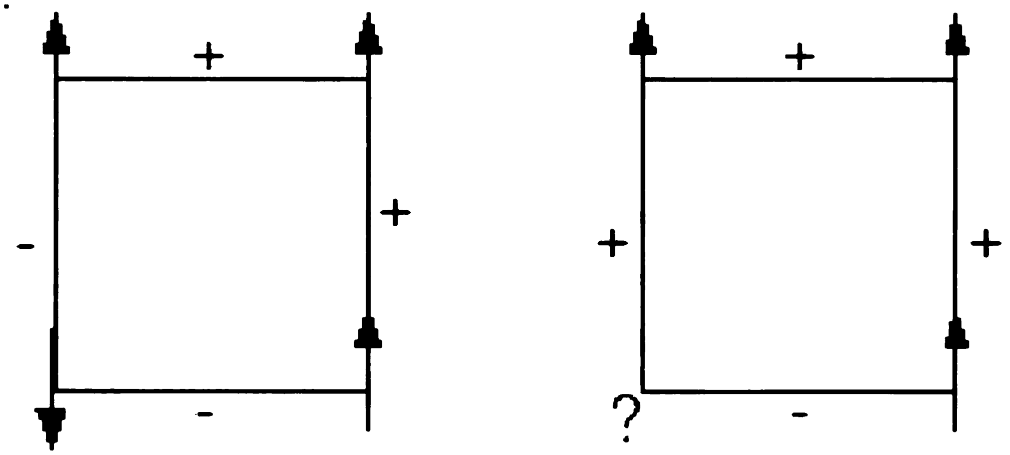


Figure 1.3: Frustration in a square lattice: on the left hand side an unfrustrated plaquette is shown while on the right hand side a frustrated plaquette is shown.

A spin glass has two properties: *frustration* and *quenched disorder*.

The term frustration refers to this inability to satisfy all the bonds To illustrate the physics of frustration, consider a two-dimensional Ising model on a square lattice with nearest-neighbor couplings J_{ij} which can take on only the values $\pm J$. We examine its energetics at a level of a single square of 4 spins and their 4 mutual couplings. Such an elementary unit of lattice is called a *plaquette*. If the number of negative bonds is even, then it is always possible to find a pair of spin configurations which satisfy all the bonds. One simply chooses one of the spins to be, for example up and then move, say, clockwise around the loop determining the value of the next spin to be that of the previous one multiplied by the sign of the bond connecting them. Then there will be no conflicting instructions coming from the originally-fixed spin when we get all the way around the loop to the starting point again.

If the number of negative bonds is odd like in Fig.(1.3), there will be a conflict when one gets all the way arround the loop. The bond connecting the last spin and the originally fixed spin will not be satisfied. If one tries to satisfy it by flipping either of these two spins, one will break another bond instead.

In frustrated systems the individual entities that construct the model (spins, for the spin glass system) feel some sort of frustration in the literal sense. Toulouse [107] introduced the concept of frustration for this plaquette that is specific to spin gasses.   In mathematical terms, quenched disorder is harder to analyze than its annealed counterpart since the thermal and the disorder averaging play very different roles. In fact the problem is so hard that few techniques to approach it are known, most of them relying on approximations. The most used is replica theory (which we describe in Chapter 3), a technique based on a mathematical analytical continuation (known as the replica trick) which although it gives results in accord with experiments in a large range of problems, is not a rigorous mathematical procedure and is still the subject of research.

If we perform for example a Monte Carlo simulation on quenched disordered systems we have to overcome large energy barriers between the various minima. As a consequence, the relaxation times become very large and it is well known in the community of computational physicists that investigating disordered systems at equilibrium is almost impossible for large systems. So an important aspect is that the system size has to be relatively small. Another important observation known as "large sample-to-sample fluctuations" is that different samples, *i.e.* different disorder realizations can have completely different properties. This observation originates in the lack of self-averaging in some physical observable so we can have rare events (*i.e.* disorder configurations with small probability) that have strong impact on averaged quantities like susceptibilities or autocorrelations. So when disordered systems are investigated we have to sample a huge number of disordered configurations and to be modest in system size.

Ising spin glasses map to one of the optimization problems defined in the previous section, namely Max-Cut [8]. Of course, due to the completeness property, a spin glass can be mapped to all other NP-complete problems so finding a solution to one of them would solve the Ising spin glass problem.

1.4.4 Number partitioning

A partition can be encoded by Ising spins $s_i = \pm 1 \; : s_i = 1 \;\; if \;\; a_i \in A_1$ and $s_i = -1$, otherwise. So, we search for the Ising spin configurations $s = (s_1, s_2, ..., s_N)$ that minimizes the energy or cost function:

$$E(s) = |\sum_{j=1}^{N} a_j s_j| \tag{1.6}$$

Despite its simplicity, the NPP was shown to belong to the NP-complete class no deterministic polynomial algorithm that solves all the instances of this problem in polynomial time, is currently available. The fact that the NPP problem is frustrated can easily be understood by squaring Eq.(1.6), so that the problem of minimizing the energy E becomes equivalent to finding the ground state of Ising Hamiltonian:

$$H = E^2 = \frac{1}{2} \sum_i \sum_{j>i} a_i a_j s_i s_j \tag{1.7}$$

In statistical mechanics, this is an infinite range Ising spin glass with Mattis-like, anti-ferromagnetic couplings $J_{ij} = -a_i a_j$ [44].

1.4.5 Maximum Independent Set and Vertex Cover

Maximum independent set (MIS) is a problem of broad interest in both the statistical physics and computer science communities. Finding a maximum independent set MIS, or the minimum vertex cover MVC to which it is trivially related, is one of the six fundamental NP-complete problems [46], and is NP-complete even on planar graphs.

As we already defined in the previous section, an independent set (IS) in a graph is a set of vertices such that no two independent sites share an edge. The MIS is an IS that contains the maximum number of sites [46] . In statistical physics, an MIS corresponds to the maximum packing state of a hard core lattice gas [56] . The hard core lattice gas Hamiltonian for the MIS is

$$H = J \sum_{ij} \epsilon_{ij} n_i n_j - \mu \sum_{i} n_i \tag{1.8}$$

where $n_i = 1$ if a site is part of the MIS and $n_i = 0$ if a site is part of the minimum vertex cover MVC . In order to find the MIS, we take the limit $J \longrightarrow \infty$ to ensure that no bond has an MIS site at both of it's ends and $\epsilon_{ij} = 1$ if a bond exists between sites i and j. The chemical potential μ weights the cardinality of an independent set. In order to find the MIS, we take the limit $\beta\mu \longrightarrow \infty$, with $\beta = 1/k_B T$ k_B being the Boltzmann's constant and $J/\mu \longrightarrow \infty$ to ensure that the independent set (hard core) condition is preserved. Here $\beta = 1/k_B T$ where k_B is Boltzmann's constatnt and T is the temperature.

1.5 Novel Techniques for Attacking NP-C Problems: The Satisfiability Problem on a DNA computer

Remarkably, the new computer science predicts that quantum computers will be able to perform certain computational tasks in phenomenally fewer steps than any conventional *classical* computer. Moreover, quantum effects allow unprecedented tasks to be performed, such as teleportation of information, breaking supposedly unbreakable codes, generating true random numbers, and communicating with messages that expose the presence of eavesdropping.

Although attacking NP-complete problems using quantum computing is not our focus of research in this thesis, it is worth mentioning that Lidar [73] developed an algorithm which allows one to construct a superposition of qubit states, such that each state uniquely codes for a single configuration of Ising spins. A central feature of the algorithm is that the quantum probability of each state in the superposition is exactly equal to the thermodynamic weight of the corresponding configuration. When a measurement is performed, it causes the superposition to collapse into a single state. The probabilities of measuring states are ordered by the energies of the corresponding spin configurations, with the ground state having the highest probability. Therefore, statistical averages needed for calculations of thermodynamic quantities obtained from the partition function are approximated in the fastest converging order in the number of measurements. Unlike Monte Carlo simulations on a classical computer, consecutive measurements on a quantum computer are totally uncorrelated.

Research at the intersection of the biological and computational sciences holds the potential to enable a number of important advances for both communities. Computational models of cell regulatory networks and biochemical signaling cascades are being constructed to elucidate the inner working of living cells. Biologists are utilizing these models to explore the logical implication of alternative competing hypothesis, to design drugs that are highly selective for specific targets, and to control the behavior of cells in response to external inputs. Similarly, advances in the biological sciences are being used to drive innovation in the design of new computing architectures based on biomolecules. The inherent ability of DNA and RNA nucleotides to perform very big computations is being exploited to solve hard problems, as outlined in the next section.

1.5.1 The Satisfiability Problem on a DNA computer

Recently, the vast parallelism in molecular computation proved the capability of attacking NP-complete problems that have resisted conventional methods. For molecular computations have been proposed different methods, [3], [74], [11], [102], [52], [100], [38] few of them being applied experimentally with promising results [3], [100], [52], [37], [53], [39]. The $3 - SAT$ problem, became the benchmark for testing the performance of DNA computers, after Lipton [74] demonstrated that it was well suited to take advantage of the parallelism specific to molecular computation. A group led by Smith [37] used surface-based chemistry to solve a four-variable (16 possible truth assignments) instance of the problem. Yoshida and Suyama [53] also solved a four-variable instance using a DNA program implementing a breadth first search. Sakamoto *et al.* [35] solved a six-variable (64 possible truth assignments) problem using hairpin DNA. A group led by Landweber [39] used RNA to solve an instance of a nine-variable (512 possible truth assignments) satisfiability problem related to the "Knights Problem" in chess.

In the following we describe an experiment done by Adleman [3]. A 20-variable ($1,048,576$ possible truth assignments) instance of the $3 - SAT$ problem is solved using a simple DNA computer. This computational problem is the largest solved using non-electronic means. The architecture used is related to the Sticker Model [103] which uses two basic operations for computation: separation based on subsequence and application of stickers. We use only separations which are carried out using oligonucleotide probes immobilized in polyacrylamide gel-filled glass modules where information-carrying DNA strands are moved through the modules by electrophoresis. In the module are kept strands with subsequences complementary to those of the immobilized hybridized probes. By running electrophoresis at temperature higher than the melting temperature, the capture strands are released from the probes and then transported using electrophoresis to new modules for further separations.

We focus on the computational part and we do not emphasize technical details about the experiment. The input was a a formula with 20-variables 24-clause

3-conjunctive normal form

$$F = (\bar{x}_3 \vee \bar{x}_{16} \vee x_{18}) \wedge (x_5 \vee x_{12} \vee \bar{x}_9) \wedge \cdots (x_8 \vee \bar{x}_7 \vee \bar{x}_{15}) \wedge (\bar{x}_8 \vee x_{16} \vee \bar{x}_{10})$$

In order to make the computation more challenging, this formula was chosen to have a unique truth assignment (*i.e.* a unique solution):

$$\begin{gathered} x_1 = F,\ x_2 = T,\ x_3 = F,\ x_4 = F,\ x_5 = F,\ x_6 = F,\ x_7 = T, \\ x_8 = T,\ x_9 = F,\ x_{10} = T,\ x_{11} = T,\ x_{12} = T,\ x_{13} = F,\ x_{14} = F, \\ x_{15} = T,\ x_{16} = T,\ x_{17} = T,\ x_{18} = F,\ x_{19} = F,\ x_{20} = F \end{gathered} \tag{1.9}$$

To represent all possible truth assignments, a Lipton encoding [74] was used. For each of the 20 variables x_k, $k = 1, \cdots, 20$, two distinct 15 base value sequences were designed: one representing true (T), X_k^T and one representing false (F), X_k^F. Below are shown few examples of sequences written 5′ to 3′:

$$\begin{gathered} X_1^T = TTA\ CAC\ CAA\ TCT\ CTT,\ X_1^F = CTC\ CTA\ CAA\ TTC\ CTA, \\ X_2^T = ATT\ TCC\ AAC\ ATA\ CTC,\ X_2^F = AAA\ CCT\ AAT\ ACT\ CCT, \\ \ldots \\ X_{19}^T = ACC\ CAT\ TAC\ TAC\ CAT,\ X_{19}^F = ACC\ CAT\ TAC\ TAC\ CAT, \\ X_{20}^T = ACA\ CAA\ ATA\ CAC\ ATC,\ X_{20}^F = CAA\ CCA\ AAC\ ATA\ AAC. \end{gathered} \tag{1.10}$$

Each of the 2^{20} truth assignments was represented by a library sequence of 300 bases consisting of the ordered concatenation of one value sequence for each variable. Single-stranded DNA molecules with library sequences were termed *library strands* and a collection of all library strands duplexed with complements was termed a *full library*. For each of the 40 sequences $\bar{X}_k^Z$, $k = 1, \cdots, 20$, $Z = T$ *or* F, 5′-end Acrydite-modified oligonucelotides were obtained and used as probes during separation operations.

To test the full library, PCR amplifications were run with primer sets: $< X_1^T, \bar{X}_k^T >$, $< X_1^T, \bar{X}_k^F >$, $< X_1^F, \bar{X}_k^T >$, $< X_1^F, \bar{X}_k^F >$, $< X_1^T, \bar{X}_1^F >$, $< X_k^T, \bar{X}_k^F >$ for various k. Gel analysis of the resulting products showed bands of the expected lengths as it is illustrated in Fig.(1.5). In Fig.(1.4) it is shown the computer consisted of an electrophoresis box with a hot chamber and a cold chamber, a glass library module filled with polyacrylamide gel containing covalently bound full library, and for each of the 24 clauses of formula F a glass *clause module* filled with polyacrylamide gel containing covalently bound probes and designed to capture only library strands encoding truth assignments satisfying that clause [101].

The computational protocol that was used is as follows:

Step 1: Insert the library module into the hot chamber of the electrophoresis box and the first clause module into the cold chamber of the box. Begin electrophoresis. The library strands melt off their Acrydite-modified complements in the library module and migrate to the first clause module. Library strands encoding truth assignments satisfying the first clause are captured in the capture layer, while library strands encoding non-satisfying assignments run through the capture layer and continue into the buffer reservoir.

Step 2: Remove both modules from the box. Discard the module from the hot chamber. Wash the box and add new buffer. Insert the module from the cold chamber into the hot chamber and the module for the next clause into the cold chamber.Begin electrophoresis. During Step 2, library strands melt off their Acrydite-modified probes in the clause module located in the hot chamber and migrate to the clause module in the cold chamber. Library strands encoding truth assignments satisfying the clause associated with the module in the cold chamber will be captured, while library strands encoding non-satisfying assignments will run through the capture layer and continue into the buffer reservoir.

Step 3: Repeat Step 2 for each of the remaining 22 clauses. At the end of Step 3, the final (24th) clause module will contain only those library strands which have been captured in all 24 clause modules and hence encode truth assignments satisfying each clause of formula F and therefore formula F itself.

Step 4: Extract the answer strands from the final clause module, PCR-amplify, and "read" the answer.

For assigning truth values to variables x_1 and x_{20}, 1μl aliquots of $10-$, $20-$, $30-$, $40-$, $50-$, $60-$, and 100- fold dilutions of the answer stock were PCR-amplified with primer sets:

$$< X_1^T, \bar{X}_{20}^T >, < X_1^T, \bar{X}_{20}^F >, < X_1^F, \bar{X}_{20}^T >, < X_1^F, \bar{X}_{20}^F > .$$

Gel analysis of the PCR products for $10-$, $20-$, $30-$, $40-$, 50-fold dilutions showed no bands except for primer set: $< X_1^F, \bar{X}_{20}^F >$. These primer sets gave only a band corresponding to 300 bp. Based on this, x_1 and x_{20} were assigned to be false. Analysis of the PCR products for the 60- and 100-fold dilutions showed no bands for any primer set. For assigning truth values to the variables $x_2, x_3 \cdots x_{19}$, and as a redundant test for the truth value of x_{20}, a 1μl aliquot of the 50-fold dilution of the answer stock was PCR-amplified with primer sets: $< X_1^T, \bar{X}_k^T >$, $< X_1^T, \bar{X}_k^F >$, $< X_1^F, \bar{X}_k^T >$, $< X_1^F, \bar{X}_k^F >$, where $k = 2, 3, \cdots, 20$. According to Fig.(1.6) gel analysis showed that in each case only one combination of primers gave a band and this band was of the expected length, compared with the one from Fig.(1.5). On this basis, truth-values were assigned to each variable. These experimentally derived truth values corresponded to the unique satisfying truth assignment for formula F given by Eq. (1.9).

To summarize, in this chapter we gave the computer science definition and their counterpart in statistical physics version of four optimization problems. Many ideas from statistical physics methods (like replica technique and the cavity method) have been applied to these problems and will be discussed more detailed in the next chapter. On the other hand, methods from computer science have proven useful in statistical physics, recent examples being belief propagation or exact methods like branch and cut in the study of spin glasses. We also described a novel procedures to attack the NP-complete problems based on the vast parallelism computation feature: an experiment using a DNA computer was introduced which illustrates that biological molecules can be used also for distinctly non-biological purposes. A minimalistic approach was taken in this experiment: a 20-variable instance of a 3-SAT problem was solved using (except during input and output) DNA Watson-Crick pairing and melting as the sole operation. Though computational theory would predict it, nonetheless it is remarkable that this basic molecular interaction could sustain such a complex computation.

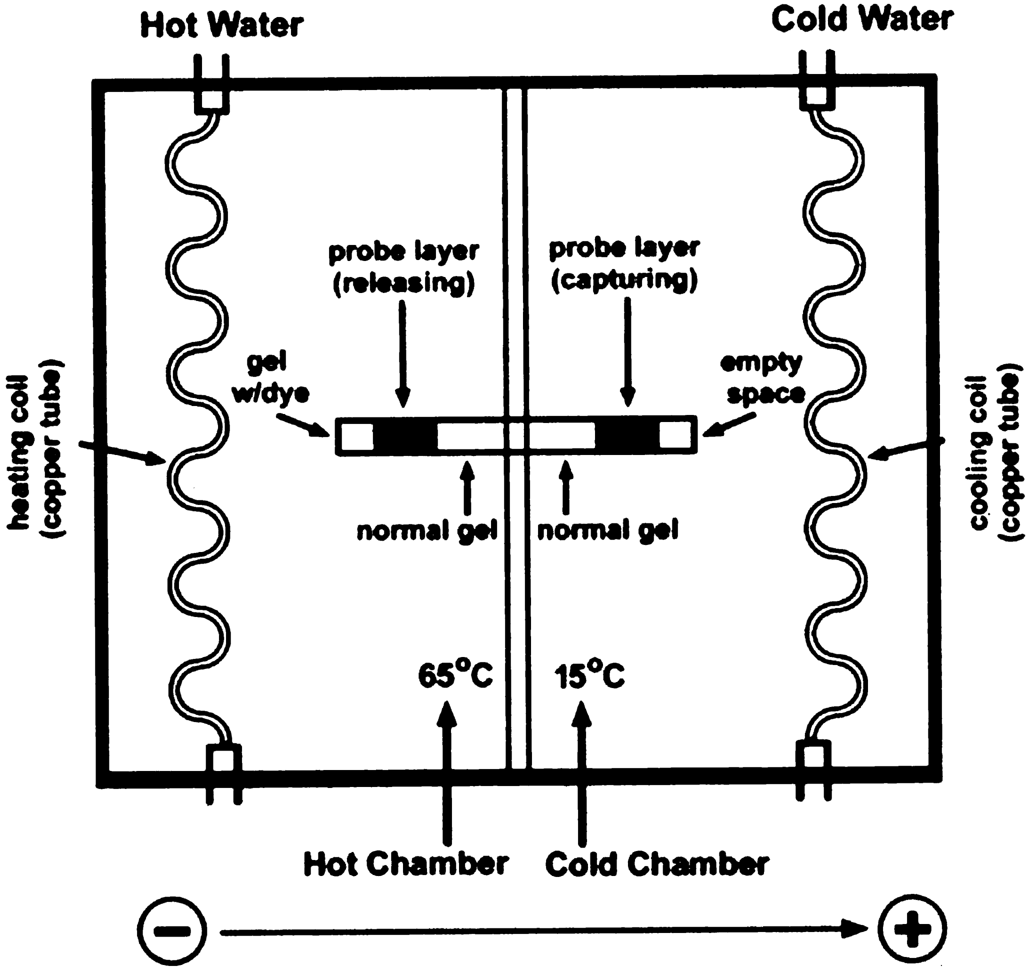


Figure 1.4: The DNA computer [101]

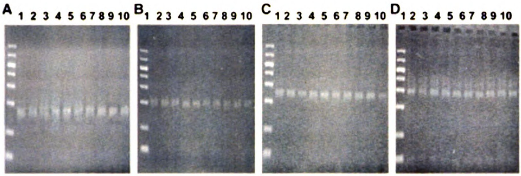


Figure 1.5: Analysis of the full library. Purified full library was *PCR*-amplified under standard conditions for 15 cycles. *PCR* products were analyzed on 4% agarose gels. Lanes 1 and 2 correspond to primer set $< X^T_{1'}, X^T_k >$, lanes 3 and 4 correspond to primer pair $< X^T_{1'}, X^T_k >$, lanes 5 and 6 correspond to primer pair $< X^T_{1'}, X^F_k >$, lanes 7 and 8 correspond to primer pair $< X^F_{1'}, X^F_k >$, lanes 9 and 10 correspond to primer pair $< X^F_{1'}, X^F_k >$, where: $(A)\ k = 11;\ \ (B)\ k = 14;\ \ (C)\ k = 17;\ \ (D)\ k = 20$. Molecular weight markers are on the leftmost lane of each gel.

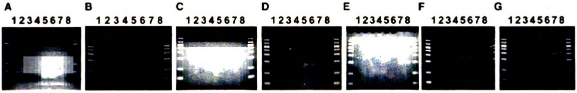


Figure 1.6: Readout of the answer: $1 - \mu l$ aliquots of a 50-fold dilution of the answer stock were *PCR*-amplified under standard conditions for 25 cycles. *PCR* products were analyzed on 4% agarose gels. Lanes 1 and 2 correspond to primer set $< X^T_{1'}, X^T_k >$, lanes 3 and 4 correspond to primer pair $< X^T_{1'}, X^T_k >$, lanes 5 and 6 correspond to primer pair $< X^F_{1'}, X^T_k >$, lanes 7 and 8 correspond to primer pair $< X^F_{1'}, X^F_k >$ where where: $(A)\ k = 2;\ \ (B)\ k = 5;\ \ (C)\ k = 8;\ \ (D)\ k = 11\ (E)\ k = 14\ (F)\ k = 17\ (G)\ k = 20$. Molecular weight markers are on the leftmost lane of each gel.

Chapter 2

Algorithms

There are many useful algorithms in both, the P and NP classes. In this chapter we introduce some of these algorithms and some of their connections with statistical physics. We also introduce new algorithms for solving (at least approximatively) hard problems as they are related to the methods described in Chapter 3 and 4.In Chapter 5 we use these algorithms on applications to gene networks.

Many difficult ground state problems like the Random Field Ising Model (RFIM), Spin Glasses in two dimensions, domain walls in random bond magnets, arrays of directed polymers or rigidity percolation are only a few problem from physics which are exactly solvable in polynomial time.

We present three basic classes of polynomial network algorithms: (i) flow, (ii) minimal path and (iii) minimum spanning tree. These algorithms are *greedy* in the sense that the algorithms make the choice which is the best at that time. Greedy algorithms solve some problems exactly and provide approximations to hard computational problems.

2.1 Minimum Spanning Tree

A spanning tree T of graph G with N vertices is a connected acyclic subgraph that spans all the nodes so the spanning tree has $N-1$ edges. Given an undirected graph $G=(N,E)$ with N nodes and E edges, with each each edge having a weight (or cost) c_{ij} with $(i,j)\in E$, we want to find a spanning tree called the *minimum spanning tree* which has the smallest total weight, measured as the sum of weights (costs) of the edges in the spanning tree.

2.1.1 Applications

The minimum spanning tree problem arises in a number of applications: (i) designing physical systems, (ii) optimal message passing, (iii) reducing data storage and (iv) cluster analysis. Two applications of minimum spanning tree, namely designing physical systems and cluster analysis are now presented.

Designing Physical Systems. We need to design a network that will connect components of a system that are geometrically dispersed and we are interested in the cheapest possible connection, a minimum spanning tree. This problem arises in the following settings:

- Connect terminals and cable amongst the panels of electrical equipment in such a way to use the least possible length of wire.

- Construct a pipeline network to connect a number of towns using the smallest possible total length of pipes.

- Construct a digital computer system, composed of high-frequency circuitry, when it is important to minimize the length of wires between different components to reduce both capacitance and delay line effects.

Cluster Analysis

Data points within a particular group of data, or cluster are more closely related to each other than data points not in a cluster. Suppose that we are interested in finding a partition of a set of n points in two-dimensional Euclidian space into clusters. A popular method for solving this problem, which we describe in the next section is Kruskal's algorithm. At each intermediate iteration, Kruskal's algorithm maintains a forest and adds edges in non-decreasing order of their weight. The nodes spanned by the trees at intermediate steps can be seen as different clusters and these clusters are often good solutions for the clustering problem. Kruskal's algorithm can be thought of as providing n partitions: the first partition contains n clusters, each cluster containing a single point, and the last partition contains just one cluster with all the points connected in the minimum spanning tree.

2.1.2 Kruskal's Algorithm and Prim's Algorithm

In the following we will describe two algorithms for solving the minimum spanning tree problem: Kruskal's algorithm and Prim's algorithm. The two algorithms are *greedy* in the sense that at each step the best choice is made. The greedy strategy generally does not find globally optimal solutions, but it does find the globally optimal solution for the minimum spanning tree problem.

Growing a minimum spanning tree.

Assume that we have a connected, undirected graph $G = (V, E)$ with a weight function $w \rightarrow R$ and we want to find a minimum spanning tree for G. The algorithm manages a set A that is always a subset of some minimum spanning tree. At each step, an edge (u, v) is determined that can be added to A without violating this invariant, in the sense that $A(u, v)$ is also a subset of a minimum spanning tree. We call such an edge a *safe edge* for A, since it can be safely added to A without destroying the invariant. Two notions that we need to introduce in order to understand the algorithm are the definition of *cut* and of *light edge.*

Definition: A *cut* $(S, V - S)$ of an undirected graph $G = (V, E)$ is a partition of V. An edge is a *light edge* crossing a cut if its weight is the minimum of any edge crossing the cut.

Kruskal's algorithm.

Kruskal's algorithm finds a safe edge to add to the growing forest by finding of all edges that connect any two trees in the forest, an edge (u, v) of least weight. Let C_1 and C_2 denote the two trees that are connected by (u, v). Since (u, v) must be a light edge connecting C_1 to some other tree, then (u, v) must be a safe edge connecting C_1 to some other tree. The total running time of Kruskal's algorithm is $O(E \lg E)$ [23], page 505 .

Prim's algorithm

Prim's algorithm operates more like Dijkstra's algorithm for finding shortest paths in a graph. In Prim's algorithm, the edges in the set A always form a single tree. At each step, a light edge connecting a vertex in A to a vertex in $V - A$ is added to the tree so the algorithm adds only edges that are safe for A. Then, when the algorithm terminates, the edges in A form a minimum spanning tree.

2.2 Shortest Path

In a shortest path problem, we are given a weighted, directed graph $G = (V, E)$, with weight function $w \longrightarrow \mathbf{R}$ mapping edges to real-valued weights. The *weight* of a path $p =< v_0, v_1, \ldots, v_k >$ is the sum of the weights of its constituent edges

$$w(p) = \sum_{i=1}^{k} w(v_{i-1}, v_i) \tag{2.1}$$

We define the *shortest path weight* from u to v by

$$\delta(u,v) = \begin{cases} min\{w(p) : u \mapsto v\} & \text{if there is a path from u to v} \\ \infty & \text{otherwise} \end{cases}$$

A *shortest path* from vertex u to vertex v is then defined as any path p with weight $w(p) = \delta(u,v)$. Shortest-path algorithms typically exploit the property that a shortest path between two vertices contains other shortest paths within it.

Dijkstra's algorithm, named after its discoverer, dutch computer scientist Edsger Dijkstra, is an algorithm that solves the single-source shortest path problem for a directed graph with nonnegative edge weights. For example, if the vertices of the graph represent cities and edge weights represent driving distances between pairs of cities connected by a direct road, Dijkstra's algorithm can be used to find the shortest route between two cities.

Dijkstra algorithm is used for finding the minimum path, which works by growing outward from the staring node s in a similar way to breadth-first-search algorithm [96], [23]. At each step Dijkstra's algorithm chooses to advance its growth front to the next unlabeled site which has the smallest distance from the starting node. The minimal path problem is closely related to a polymer in a random medium and hence to growth problems. In the minimal-path problem, one chooses the site at the growth front which has the minimum-cost path to the source, while in the minimus spanning tree problem one chooses the minimum-cost edge.

The Bellman-Ford algorithm solves the single-source shortest-path problem in a more general case when edge weights can be negative. The algorithm returns a boolean value indicating whether or not there is a negative weight cycle that is reachable from the source. If there is such a cycle, the algorithm indicates that no solution exists and if there is no such a cycle, the algorithm produces the shortest paths and their weights. Shortest path provides a first approximation to key regulatory paths in gene networks, as discussed in Chapter 5.

2.3 Flow Algorithms

In the same way that we can model a road map as a directed graph in order to find the shortest path from one point to another, we can also interpret a directed graph as a *flow netwok*. In graph theory, *a network flow* is an assignment of flow to the edges of a directed graph (called a flow network in this case) where each edge has a capacity such that the amount of flow along an edge does not exceed its capacity. In addition we have the restriction that the amount of flow into a node equals the amount of flow out of it, except if it is a source which only has outgoing flow, or a sink which has only incoming flow.

A flow network can be used to simulate traffic in a road system, fluids in pipes, currents in an electrical circuit, flow in a porous media or anything similar in which something travels through a network of nodes. An important feature of flow algorithms is the discreteness imposed by requiring integer flows. In particular, a flow of one unit along a path can be used to model a non-intersecting polymer.

We introduce the *maximum flow* problem as the simplest problem concerning flow networks and it asks what is the greatest rate at which material can be shipped from the source to the sink without violating any capacity constraints.

2.3.1 Flow Networks

Consider a graph $G(V,E)$ with nodes V and edges E, and special nodes source s (in-degree 0) and sink t (out-degree 0), let $f(u,v)$ be the flow from node u to node v, and $c(u,v)$ the capacity (maximum flow possible). A network flow is a real function $f : V \times V \to R$ with the following three properties for all nodes u and v:

(i) Capacity constraints: $f(u,v) \leq c(u,v)$. The flow along an edge cannot exceed its capacity.

(ii) Skew symmetry: $f(u,v) = -f(v,u)$. The net flow from u to v must be the opposite of the net flow from v to u.

(iii) Flow conservation: $\sum_{v \in V} f(u,v) = 0$, unless $u = s$ or $u = t$. The net flow to a node is zero, except for the source, which produces flow, and the sink, which consumes flow.

To define the maximum flow we need to introduce two concepts: *the residual capacity* and *the augmenting path*. The residual capacity of an edge is $c_f(u,v) = c(u,v) - f(u,v)$. This defines a residual network denoted $G_f(V,E_f)$, giving the amount of available capacity. Note that there can be an edge from u to v in the residual network, even though there is no edge from u to v in the original network. Since flows in opposite directions cancel out, decreasing the flow from v to u is the same as increasing the flow from u to v. An augmenting path is a path $(u_1, u_2, \ldots, u_k)$, where $u_1 = s, u_k = t$, and $c_f(u_i, u_{i+1}) > 0$, which means it is possible to send more flow along this path.

The max-flow min-cut theorem is a statement in optimization theory about maximal flows in flow networks and it states that the maximal amount of a flow is equal to the capacity of a minimal cut. Suppose G(V,E) is a finite directed graph and every edge (u,v) has a capacity $c(u,v)$ (a non-negative real number). Further assume two vertices, the source s and the sink t, have been distinguished. As we already defined, a *cut* is a split of the nodes into two sets S and T, such that s is in S and t is in T, so there are 2^{V-2} possible cuts in a graph. The capacity of a cut (S,T) is the sum of the capacity of all the edges crossing the cut, from the region S to the region T: $c(S,T) = \sum_{u \in S, v \in T | (u,v) \in E} c(u,v)$

Theorem: Min-cut/Max-flow The following three conditions are equivalent:

(i) f is a maximum flow in G

(ii) The residual network G_f contains no augmenting paths.

(iii) $f = c(S,T)$ for some cut (S,T). At first glance, finding the minimum cut looks like a hard problem because it seems to require searching over all possible cuts, which is worse than exponential in the number of nods in the graph. However simple polynomial algorithms exist.

In many of the physics applications, the minimum cut is important because it corresponds to the interface structure in interface problems, and the domain structure of random magnets. The residual graph contains the minimum cut structure which means that it contains information about complex ground states morphologies. The residual graph and its minimum cut structure are used in two additional ways: testing the sensitivity of cuts to small perturbations [4] and finding all the cuts in degenerate systems [10].

The algorithms that we introduced in Sections 2.1 – 2.3 are useful for solving *easy* (*i.e.* polynomial) problems. In the next section we introduce algorithms for solving approximately *hard* (*i.e.* NP-complete) problems.

2.4 Message Passage Techniques

We first explain the principles that describe *belief propagation*, which is an efficient way to approximate inference problems based on passing local messages. Inference problems arise in computer vision, error-correcting codes and last but not least in statistical mechanics. We make a survey of inference problems from artificial intelligence, computer vision, digital communications and statistical physics. Algorithms of these type are used in Chapter 4.

2.4.1 Brief Survey of Inference Problems

An example from Artificial Intelligence (AI): Bayesian Networks

In the artificial intelligence literature Bayesian Networks are the most popular type of graphical model. They are used in expert systems involving different domains like medical diagnoses, heuristic search and so on.

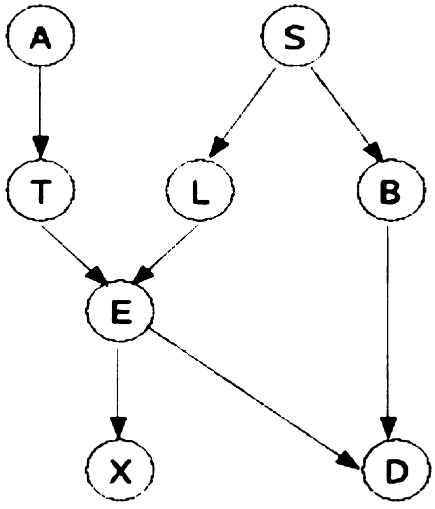


Figure 2.1: The fictional *Asia* Bayesian network [72]

We will take an example from medicine. Let us assume that we want to construct a machine that automatically gives the diagnosis for patients. For each patient we have information about symptoms and test results and we want to *infer* the probability that a given disease or set of diseases is causing the symptoms. We also assume that we know the statistical dependencies between different symptoms, test results and disease. So let us consider the following example:

(a) A recent trip to Asia (A) increases the chance of tuberculosis (T); (b) Smoking (S) is a risk for both lung cancer (L) and bronchitis (B); (c) The presence of either (E) tuberculosis and lung cancer is indicated by an X-ray (X) result;

(d) Dyspnoea (D) can be caused by bronchitis (B) or either (E) tuberculosis (T) or lung cancer (L). In Fig. (2.1) each node represents a variable that can be in a discrete number of states and call x_i the variable representing different possible states of node i. Associated with each arrow is a conditional probability. For example, $p(x_L \mid x_S)$ is the conditional probability for a patient having lung cancer given that he does or he does not smoke; for this link we say that the S-node is the parent of the L-node because x_L is conditionally dependent on x_S according to figure. In this example, the overall joint probability that the patient has some combination of symptoms, test-results and disease is the product of all the probabilities of the parent nodes and all the conditional probabilities:

$$\begin{aligned} p(x) = p(x_A)p(x_S)p(x_T \mid x_A)p(x_L \mid x_S)p(x_B \mid x_S) \\ p(x_E \mid x_T, x_L)p(x_D \mid x_B, x_E)p(x_X \mid x_E) \end{aligned} \tag{2.2}$$

More generally, a Bayesian network is a directed acyclic graph of N random variables x_i that defines a joint probability function:

$$p(x_1, x_2, \cdots, x_N) = \prod_{i=1}^{N} p(x_i \mid Parent(x_i)) \tag{2.3}$$

The goal is to compute certain marginal probabilities (in our case we want to calculate the probability that a patient has a certain disease). From a mathematical point of view, marginal probabilities are defined in terms of sums over all the possible states of all the other nodes in the system. We refer to marginal probabilities that we compute approximately as *beliefs* and denote the belief at node i by $b(x_i)$. If we have some information about the nodes (for example we know that the patient does not smoke) then we are able to fix the corresponding variable and we don't have to sum over the unknown states of that node. Such a node we call an *observable node* otherwise we call it an *hidden node*. For small Bayesian networks, we can do marginalization sums directly, but unfortunately, the number of terms will grow exponentially with the number of hidden nodes in the network. Using Belief Propagation algorithms we can compute marginal probabilities, at least approximately, in a time that grows only linearly with the number of nodes in the system.

2.4.2 An Example from Computer Vision: Pairwise Random Markov Fields (PRMF)

In computer vision problems [98] we want to infer the details of whatever is out there based on the data that we are given. We assume that we observe some quantities about the image y_i, and we want to infer some other quantities about the underlying scene x_i. We also assume that there exists a statistical dependency between x_i and y_i at each position i and we write this as a joint compatibility function $\Phi_i(x_i, y_i)$ (often called the "evidence" for x_i). Nodes i are arranged in a two-dimensional grid, and scene variables x_i should be compatibile with nearby scene variables x_j, as represented by a compatibility function $\Psi_{ij}(x_i, x_j)$, where Ψ_{ij}, connects only neighbor positions. The overall joint probability of a scene x_i and an image y_i reads:

$$p(x,y) = \frac{1}{Z} \prod_{(ij)} \Psi_{(ij)}(x_i, x_j) \prod_i \Phi_i(x_i, y_i) \tag{2.4}$$

with Z a normalization constant and the first product is over nearest neighbors (ij) on the square lattice. Fig. (2.2) shows a graphical description where the filled-in circles represents the "observed" image nodes y_i and the empty circles represents the "hidden scene nodes", x_i. Our goal is to calculate the belief $b(x_i)$ for all positions i and in this way to be able to infer something about the underlying scene. A direct computation of marginal probabilities would take exponential time so we need a faster heuristic algorithm like the belief propagation algorithm.

2.4.3 Mapping to Statistical Physics

The PRMF decribed earlier can de transformed into a form which physicists recognise. Define the interaction J_{ij} between the variables at neighboring nodes by $J_{ij} = \ln \Psi_{ij}(x_i j)$ and the field at each node by $h_i(x_i) = \ln \Phi(x_i, y_i)$. If we define

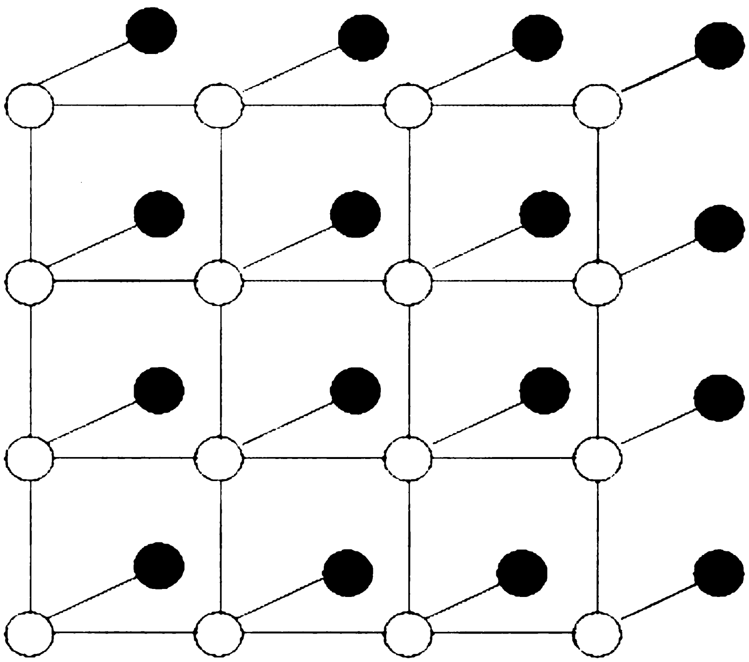


Figure 2.2: A square lattice Pairwise Random Markov Field

the energy as:

$$E(x) = -\sum_{(ij)} J_{ij}(x_i x_j) - \sum_{i} h(x_i) \tag{2.5}$$

and we use Boltzmann's law

$$p(x) = \frac{1}{Z}\, Exp\left(\frac{-E(x_i)}{T}\right) \tag{2.6}$$

where Z is the normalization constant called the *partition function,* we see that our pairwise *MRF* corresponds to statistical field theory at $T = 1$. If the number of states at each node is exactly two, the model reduce to the Ising model. For this case we change variables from Boolean variables $x_i = 0 \quad (1)$ to spin variables $S_i = -1 \quad (1)$ and if we restrict to J_{ij} interactions which have a symmetric form than we have a spin glass energy function [84]:

$$E(s) = -\sum_{(ij)} J_{ij}\, s_i\, s_j - \sum_{i} h_i\, s_i \tag{2.7}$$

In the context of the Ising model, the inference problem of computing beliefs $b(x_i)$ can be mapped onto the physics problem of computing local magnetizations:

$$m_i = b(s_i = 1) - b(s_i = -1) \tag{2.8}$$

The magnetization is equal to the marginal probability and the marginal probability is approximated by the beliefs.

2.4.4 Tanner Graphs and Factor Graphs

In the following we introduce a special type of graph namely *factor graph* to represent more than 2-body interactions. A factor graph is a bipartite graph that shows how a *global* function of many variables factors into a product of *local* functions and so goes beyong pairwise interactions [71].

Consider a set of N discrete-valued random variables $X_1, X_2, \cdots, X_N$, and consider x_i the possible realizations of random variables X_i. The joint probability function factors into a product of functions which has the general form:

$$p(\mathbf{x}) = \frac{1}{Z}\prod_a f_a(x_a) \tag{2.9}$$

with $\mathbf{x} = x_1, x_2, \cdots, x_N$ and a is an index which labels M functions $f_A, f_B, \ldots, f_M$; the function $f_a(\mathbf{x}_a)$ has arguments $\mathbf{x}_a$ that are some subsets of $x_1, x_2, \ldots, x_N$. A *factor graph* is a bipartite graph that expresses the factorization structure in Eq. (2.9). It has a *variable node* (which we draw as a circle) for each variable x_i, a factor node (which we draw as a square) for each function f_a, with an edge connecting variable node i to factor node a if and only if x_i is an argument of f_a. From now on we index variable nodes with letters starting with $i,\ j, \ldots$ and factor nodes starting with $a,\ b,\ ,\ldots$. Fig. (2.3) presents an example of the factor graph corresponding to

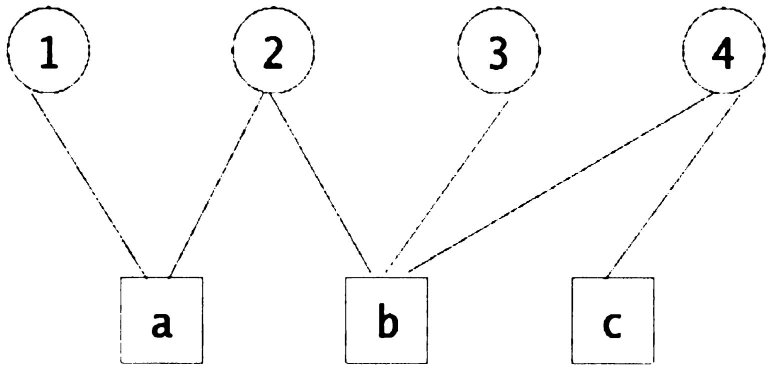


Figure 2.3: A factor graph representing the joint probability distribution given by Eq. (2.10)

$$p(x_1, x_2, x_3, x_4) = \frac{1}{Z} f_a(x_1, x_2) f_b(x_2, x_3, x_4) f_c(x_4) \tag{2.10}$$

The goal is to compute the marginal probability distributions. x_i are the states of variable node i; if S is a set of variable nodes then $\mathbf{x}_S$ denotes the states of the corresponding variable nodes. The marginal probability function obtained by marginalizing $p(x)$ onto the set of variable nodes S reads:

$$p_S(\mathbf{x}_S) = \sum_{x \setminus \mathbf{x}_S} p(\mathbf{x}) \tag{2.11}$$

where the summation over $x \setminus \mathbf{x}_S$ indicates that we sum over the states of all variable nodes which are *not* in set S. The belief propagation algorithm is a method for computing marginal probability functions [115], [113]. The computed marginal probability functions will be exact if the factor graph has no cycles, but the BP algorithm is still well-defined and empirically often gives good approximation results even when the factor graph does have cycles.

2.4.5 Standard Belief Propagation

In the next section we explain the standard belief propagation procedure on pairwise random markov fields. We assume the observed nodes y_i are fixed, write $\Phi_i(x_i)$ as a short-hand for $\Phi_i(x_i, y_i)$ and focus on the joint probability distribution for the unknown variables x_i:

$$p(x) = \frac{1}{Z} \prod_{(ij)} \psi_{ij}(x_i, x_j) \prod_i \Phi_i(x_i) \tag{2.12}$$

In the BP algorithm, we introduce variables $m_{ij}(x_j)$ which can be intuitively interpreted as a *message* from a hidden node i to a hidden node j about what state node j should be in (Fig. 2.4A). The belief at node i is proportional to the product of the local evidence at that node ($\Phi_i(x_i)$), and all the messages coming into node i:

$$b_i(x_i) = K\Phi_i(x_i) \prod_{j \in N(i)} m_{ji}(x_i) \tag{2.13}$$

where K is the normalization constant (the beliefs must sum to 1) and $N(i)$ is the set of nodes neighboring node i(Fig. 2.4B). The message update rule is:

$$m_{ij} \leftarrow \sum_{x_i} \Phi_i(x_i) \Upsilon_{ij}(x_i, x_j) \prod_{k \in N(i) \backslash j} m_{ki}(x_i) \tag{2.14}$$

and it is shown diagramatically in (Fig. 2.4C). On the right-hand-side of Eq. (2.14) we take the product over all messages going into node i except for the one coming from node j. Let us consider an example of a network with four hidden nodes like the one in Fig. (2.4).

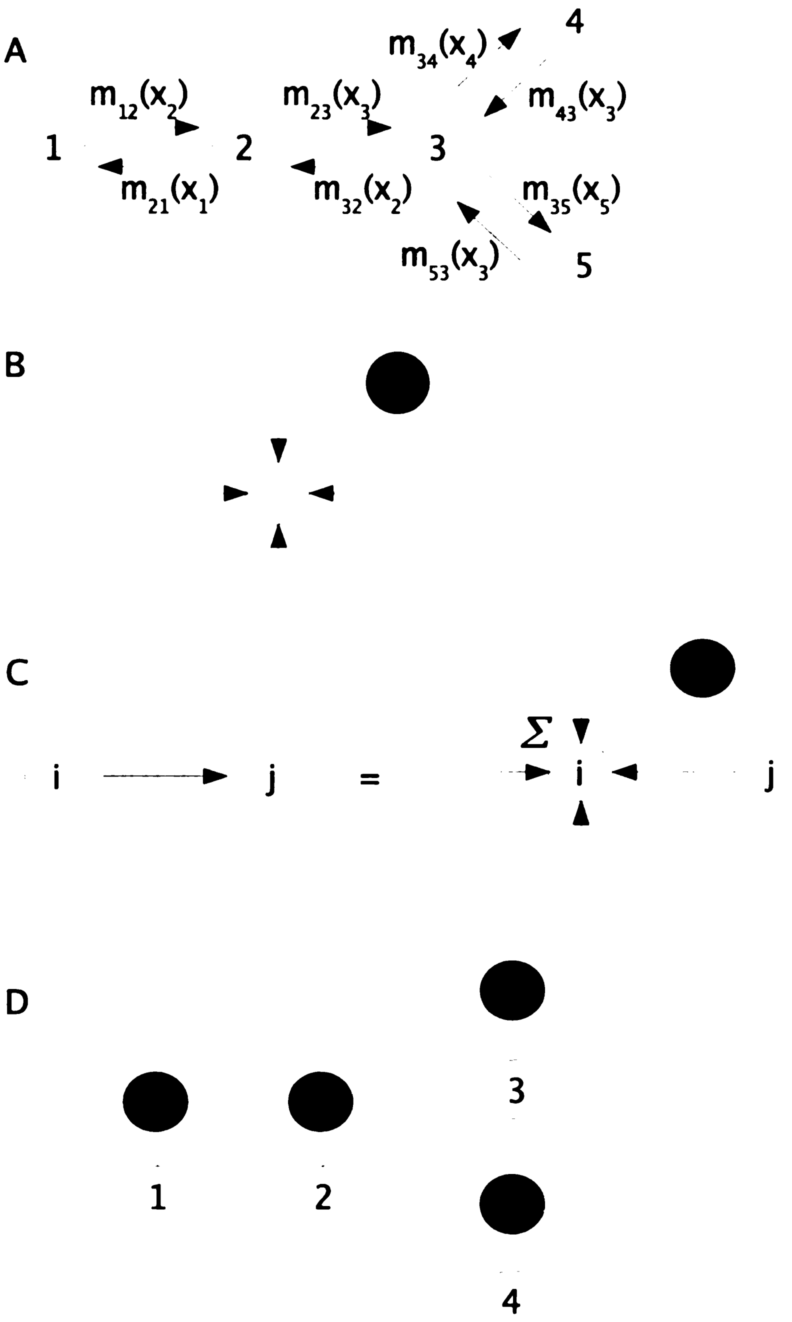


Figure 2.4: A. An illustration of the messages passed in Belief Propagation; B. A diagramatic representation of (2.13); C. A diagramatic representation of the BP message update rules 2.14. The summation symbol indicates that the summation is over all the states of node *i*; D. A pairwise MRF with four hidden nodes.

The belief at node 1 using the belief propagation rule (2.14) is:

$$b_1(x_1) = k\Phi_1(x_1)m_{21}(x_1) = k\Phi_1(x_1)\sum_{x_2}\Psi_{12}(x_1,x_2)\Phi_2(x_2)m_{32}(x_2)m_{42}(x_2)$$
$$= k\Phi_1(x_1)\sum_{x_2}\Psi_{12}(x_1,x_2)\Phi_2(x_2)\sum_{x_3}\Phi_3(x_3)\Psi_{23}(x_2,x_3)\sum_{x_4}\Phi_4(x_4)\Psi_{24}(x_2,x_4) \quad (2.15)$$

Reorganizing the sums, it is easy to see that the belief at node 1 (Fig. 2.4D) is the same as the exact marginal probabilities at node 1:

$$b_1(x_1) = k\sum_{x_2,x_3,x_4} p(\{x\}) = p_1(x_1) \quad (2.16)$$

Belief propagation gives the exact marginal probabilities for all nodes in any singly connected graph. In this case each message needs to be computed only once which means that the whole computation takes a time proportional to the number of links in the graph (so the whole computaion takes much less time than the exponentially large time required to compute marginal probabilities naively).

The two-node marginal probabilities p_{ij}, for two neighboring sites i and j, is obtained by marginalizing the joint probability function over every node, except the two nodes i and j:

$$p_{ij}(x_i,x_j) = \sum_{z_i:z_{ij}=(x_i,x_j)} p(\{z\}) \quad (2.17)$$

The belief equation for the two-nodes belief reads:

$$b_{ij} = k\psi_{ij}(x_i,x_j)\phi_i(x_i)\phi_j(x_j)\prod_{k\in N(i)\backslash j} m_{ki}(x_i)\prod_{l\in N(j)\backslash i} m_{lj}(x_j) \quad (2.18)$$

The BP algorithm, as defined in terms of the belief equations (2.13) and (2.18) and the message-update rules, does not make reference to the topology of the graph. This means that we can implement this algorithm on graphs with loops. On the other hand as Pearl warned [98], "if we ignore the existance of loops and permit the nodes to continue communicatiing to each other as if the network were singly connected, messages can circulate indefinitely around these loops , and the process may not converge to a stable equilibrium".

2.4.6 The Free Energy

In this section we introduce the Bethe approximation for the free energy and show that the fixed points of the BP algorithm corresponds to the stationary points of the Bethe free energy. For two joint probabilities $p(x)$ and $b(x)$, we can define a *distance* between $p(x)$ and $b(x)$ [115]:

$$D(b(\{x\}) \parallel p(\{x\})) = \sum_{x} b(\{x\}) \ln \frac{b(x)}{p(x)} \tag{2.19}$$

Distance D (known as Kulback-Liebler distance) does not have all the properties that we normally associate with distances: it is not symmetric and does not satisfy the triangle inequality. It is always non-negative and it is equal to zero if and only if the two probability functions $b(x)$ and $p(x)$ are equal.

In statistical physics we assume a Boltzmann distribution law:

$$p(x) = \frac{1}{Z} e^{-\frac{E(x)}{T}} \tag{2.20}$$

where T is an order parameter that defines a scale of units for the energy, and for simplicity we assume $T = 1$. From Eqs. (2.19) and (2.20) we find:

$$D(b(\{x\}) \parallel p(\{x\})) = \sum_{\{x\}} b(\{x\}) E(\{x\}) + \sum_{\{x\}} b(\{x\}) \ln b(\{x\}) + \ln Z \tag{2.21}$$

As we mentioned earlier, distance $D = 0$ ($b(\{x\}) = p(\{x\})$ when the quantity

$$G(b(\{x\})) = \sum_{\{x\}} b(\{x\})E(\{x\}) + \sum_{\{x\}} b(\{x\}) \ln b(\{x\}) = U(b(\{x\})) - S(b(\{x\})) \tag{2.22}$$

achives its minimal value

$$F = -\ln Z \tag{2.23}$$

called the Helmohltz free energy. In statistical physics language terms G, U, S and F defined in eqs. (2.22) and (2.23) are called *Gibbs energy, average energy, entropy* and *Helmholtz free energy.*

2.4.7 The Mean-Field Free Energy

Let us consider the case when joint probabilities have a particularly simple form: they are factorized over sites

$$b(\{x\}) = \prod_i b_i(x_i) \tag{2.24}$$

with constraint $\sum_i b_i(x_i) = 1$. With this mean-field approximation, the one-node beliefs are $b_i(x_i)$ and the two node beliefs are $b_{i,j}(x_i, x_j) = b_i(x_i)b_j(x_j)$. The energy configuration of a pairwise MRF is:

$$E(\{x\}) = -\sum_{(ij)} \ln \psi_{ij}(x_i, x_j) - \sum_i \ln \Phi_i(x_i) \tag{2.25}$$

The mean-field average energy and the mean-field entropy are:

$$U_{MF}(\{b_i\}) = -\sum_{(ij)} \sum_{x_i, x_j} b_i(x_i)b_j(x_j) \ln \psi_{ij}(x_i, x_j) - \sum_i \sum_{x_i} b_i(x_i) \ln \Phi_i(x_i) \tag{2.26}$$

respectively

$$S_{MF}(b_i) = -\sum_i \sum_{x_i} b_i(x_i) \ln b_i(x_i) \tag{2.27}$$

The mean-field Gibbs free energy $G_{MF} = U_{MF} - S_{MF}$ is a function of the full joint probability distribution while the mean-field free energy is only a function of one node belief. Since we know that G lies below G_{MF} it is reasonable to search for that configuration of b_i's that minimizes G_{MF} which in fact represents the standard variational justification for the mean-field theory.

2.4.8 The Bethe Free Energy

In the following we derive Gibbs free energy as a function of both the one-node belief $b_i(x_i)$ and the two-nodes belief $b_{i,j}(x_i, x_j)$ which obey the normalization conditions $\sum_{x_i} b_i(x_i) = \sum_{x_i, x_j} b_{i,j}(x_i, x_j) = 1$. In this case the average energy reads:

$$U = -\sum_{i\,j} b_{i\,j}(x_i, x_j) \ln \psi_{ij}(x_i, x_j) - \sum_i b_i(x_i) \ln \Phi_i(x_i) \tag{2.28}$$

The average energy when computed with the exact marginal probabilities $p_i(x_i)$ and $p_{ij}(x_i, x_j)$ will also have this form so if the one-node and two-node beliefs are exact, the average energy given by Eq. (2.28) will be exact. We could compute the entropy exactly if we could explicitly express the joint probability distribution $b(x)$ in terms of the one node and two-node beliefs. For a singly connected graph we can do this, case in which the joint probability distribution reads:

$$b(x) = \frac{\prod_{(i\,j)} b_{i\,j}(x_i, x_j)}{\prod_i \; b_i(x_i)^{q-1}} \tag{2.29}$$

where q_i represents the number of nodes neighboring node i. From eqs. (2.29) and (2.27) we have:

$$S_{Bethe} = -\sum_{(i\,j)} \sum_{x_i,\,x_j} b_{i\,j}(x_i, x_j) \ln b_{i\,j}(x_i, x_j) + \sum_i (q_i - 1) \sum_{x_i} b_i(x_i) \ln b_i(x_i) \quad (2.30)$$

So for a singly connected graph, the Bethe approximation of both the energy and the entropy will have the correct functional dependency of beliefs and the values of these beliefs that minimize the Bethe free energy $G_{Bethe} = U - S_{Bethe}$ will correspond to the exact marginal probabilities. For graphs with loops, the Bethe entropy and free energy will be only approximations [42]. In contrast to the mean-field free energy, the Bethe free energy in general is not an upper bound on the *true* free energy. If we introduce the local energies:

$$E_i(x_i) = -\ln \Phi_i(x_i) \quad (2.31)$$

$$E_{ij}(x_i, x_j) = -\ln \Psi_{ij}(x_i, x_j) - \ln \Phi_i(x_i) - \ln \Phi_j(x_j) \quad (2.32)$$

and using the marginalization constraints we obtain:

$$U = -\sum_{(i\,j)} \sum_{x_i, x_j} b_{i\,j}(x_i, xj) E_{i\,j}\,(x_i, x_j) + \sum_i (q_i - 1) \sum_{x_i} b_i(x_i) E_i(x_i) \quad (2.33)$$

which has exactly the same form as the entropy for Bethe approximation where we replaced the ln terms with the local energy E terms.

The Bethe free energy is equal to the exact Gibbs free energy for pairwise *MRF*'s when the graph has no loops so the Bethe free energy is minimal for the correct marginals. So when there are no loops the *BP* beliefs are the global minima of the Bethe free energy. It turns out that a set of beliefs gives a *BP* fixed point in any graph if and only if they are locally stationary points of the Bethe free energy [113]. The BP algorithm for graphical models with loops is not guaranteed to converge but because BP fixed points correspond to Bethe free energy minima, one can simply choose to minimize the Bethe free energy directly. Such free energy minimizations are slower than the BP algorithm, but at least they are guaranteed to converge.

The succes of BP algorithms is exciting because it means that many different types of problems that seem difficult to handle, involving graphs with many nodes and loops, can actually be handled using efficient and systematically correctable algorithms. These algorithms are much faster than Monte Carlo approaches, and the approximation to the free energy that they are effectively implementing are more accurate than mean-field approximations. These algorithms give a principal framework for propagating, in parallel, information and uncertainty between nodes in a network.

In the next section we will show that ideas from BP algorithms combined with ideas from statistical physics lead to a new class of algorithms, the *survey propagation* algorithms which turn out to be very effective in solving NP-complete problems.

2.5 Novel Techniques: Message Passing Algorithms and the Cavity Method

For the K-SAT problem, well below the threshold, a generic problem has many solutions, which tend to form one giant cluster; the set of all satisfying assignments form a connected cluster in which it is possible to find a path between two solutions that requires short steps only. So, greedy algorithms and other simple heuristics can readily identify a solution by a local search process.

Close to the critical thershold, however, the solution space breaks up into many smaller clusters and solutions from separate clusters are usually far apart. Clusters that correspond to partial solutions (*i.e.* satisfy some but not all of the constraints) are exponentially more numerous than the clusters of solutions and act as traps for local search algorithms.

2.5.1 The Message Passing Solution of SAT on a Tree

Survey propagation finds solutions to many problem instances in this hard region, including very large instance that cannot be solved using earlier methods. For example, for random $3 - SAT$ problem and close to the threshold, survey propagation algorithms solve instances up to 10^7 variables, whereas DPLL algorithms can solve instance with usually less than 100 variables. The message passing procedure resembles in some aspect the BP algorithm described earlier, but with few differences. The messages sent along the graph underlying the combinatorial problem are surveys of some elementary warning messages and are probability distributions parametrized in a simple way: they describe how the Boolean variables are expected to fluctuate from cluster to cluster. Once the iterative equations for such probability distributions have reached convergence, it is possible to identify the Boolean variables which can be fixed and then simplify the problem. The whole process is repeated on the simplified problem until a solution is found.

We describe the message passing procedure in the special case in which the factor graph is a tree. The algorithm uses elementary messages passed along the graph called cavity biases and cavity fields. We use the definition of these two type of messages introduced by Braunstein *et. al.* [17].

The basic elementary message passed from one functionj node a to a variable node i is a Boolean number $u_{ai} \in 0,1$ called a *cavity bias*. The basic elemntary message passed from one variable node j to a function node a is an integer number h_{ja} called a *cavity field*.

In order to compute the cavity field h_{ja}, the variable j considers the incoming cavity biases which it receives from all the function nodes b to which is connected, except $b = a$ (so from here the name *cavity*) and performs the sum:

$$h_{ja} = \left(\sum_{b \in V_+(j) \setminus a} u_{bj} \right) - \left(\sum_{b \in V_-(j) \setminus a} u_{bj} \right) \tag{2.34}$$

If j has no other neighbor than a, then $h_{ja} = 0$. In Eq. (2.34) we denoted by $V(i)$ the set of function nodes a to which it is connected by an edge, by $V_+(i)$ the subset of $V(i)$ consisting of function nodes a where the variable appears un-negated (the edge that connects a to i is a full line), and by $V_-(i)$ the subset of $V(i)$ consisting of function nodes a where the variable appears negated (the edge that connects a to i is a dashed line).

In order to compute the cavity bias u_{ai}, the function node a considers the incoming cavity fields which it receives from all of the variable nodes j to which is connected, except $j = i$. If there is at least one $j \in V(a) \setminus i$ such that $h_{ja} J_j^a \leq 0$ then $u_{ai} = 0$, otherwise $u_{ai} = 1$. If a has no other neighbor than i, then $u_{ai} = 1$. Therefore:

$$u_{ai} = \prod_{j \in V(a) \setminus i} \theta(h_{ja} J_j^a) \tag{2.35}$$

where $\theta(x)$ is the step function. Using ideas from survey propagation algorithms it was possible to have a more profound picture of the solution space for the SAT problem. It was found [85] that there exists two critical regimes for α. For $\alpha < \alpha_d \approx 3.921$ the set of satisfying assignment S_M is connected, that is one can find a short path in S_M to go from one solution to any other solution. For $\alpha_d < \alpha < \alpha_c \approx 4.267$, S_M becomes divided into subsets which are far apart in Hamming distance. If we denote with $N_{cl} \equiv \exp(\Sigma(\alpha))$ the number of such clusters and with $N_{int} \equiv \exp(S_{int}(\alpha))$ the number of solutions within each cluster, then in the language of statistical physics, the quantity Σ is called the complexity and S_{int} the

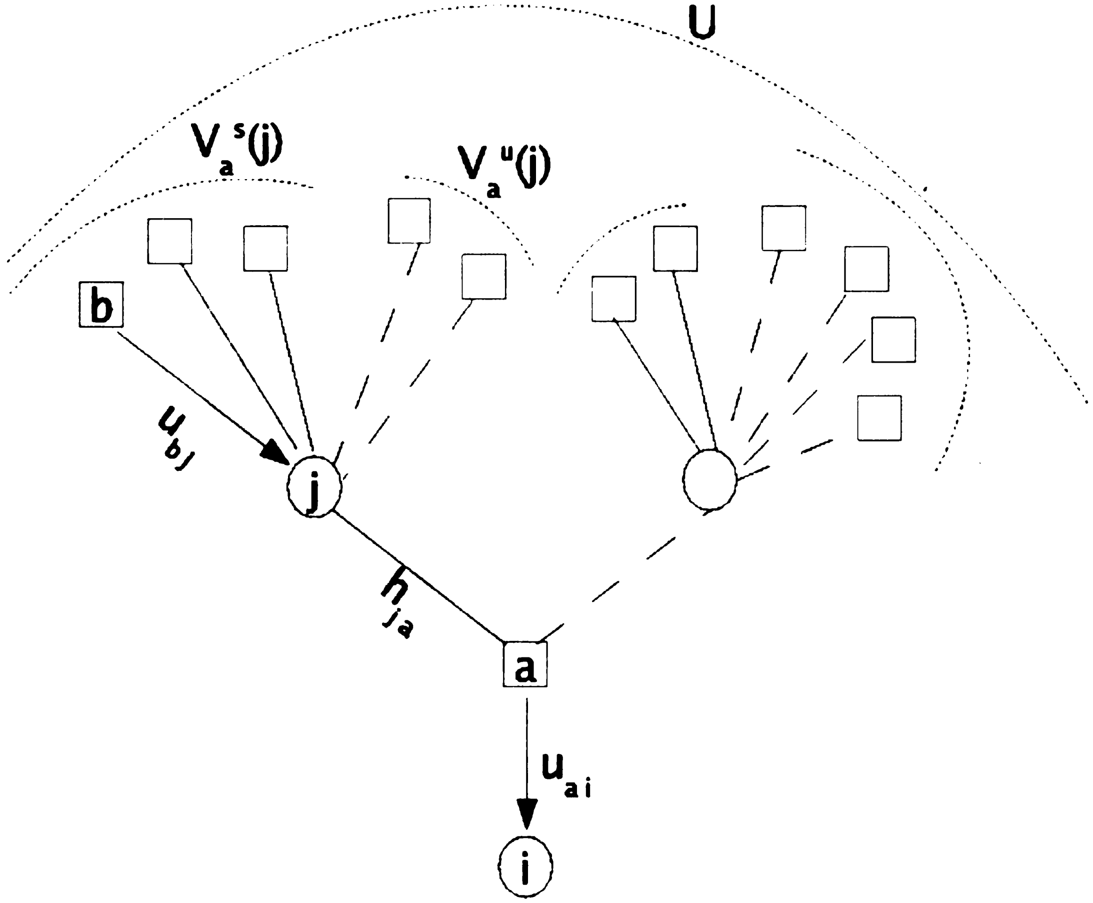


Figure 2.5: A function node *a* and its neighborhood. The survey of cavity bias cab be computed from the knowledge of the joint probability distribution for all the cavity-biases in the set U, so those coming onto all variables node *j* which are neighbors of *a*, except $j = i$.

internal entropy. Although there exists in this phase an exponentially large number of clusters, each containing an exponential number of solutions, it is very difficult to find a solution because of the proliferation of *metastable* states. A metastable cluster is a cluster of assignments which all have the same fixed number say C of violated clauses, and such that one cannot find at a small Hamming distance of this cluster an assignment which violates strictly less than C clauses.

This clustering phenomenon is particularly difficult to study in random systems but recent progress in statistical physics made possible to develop new methods like the cavity method [85] [86].

2.5.2 The Cavity Method

The cavity method, initially developed to study the Sherington Kirkpatrick model of spin glasses [68] is a powerful method to compute the properties of ground states in many condensed matter and optimization problems. The method is in principle equivalent to the replica method, but has more clearer interpretation that allows to determine solutions for problems which remain very difficult to understand in the replica formalism.

Let us consider a system of N Ising spins, $S_i = \pm 1,\ i \in 1, \ldots, N$, interacting with random couplings, with energy

$$H = -\sum_{<ij>} J_{ij} S_i S_j \tag{2.36}$$

The sum is over all links of a lattice. For each link (ij) the coupling J_{ij} is an independent random variable chosen with the same probability distribution $P(J)$. The aim is to compute in the infinite N limit, the value of the energy density of the global ground state, which is the configuration of Ising spins which minimizes the Hamiltonian given by Eq. (2.36). The ground state energy of a N spin system, averaged over the distribution of samples is denoted by E_N, so in other words we want to compute:

$$U = \lim_{N \to \infty} \frac{E_N}{N} \tag{2.37}$$

The cavity method is best exploited when the locally stucture of the underlying graph is a tree. There are various types of tree-like lattices, the most used by physicist are presented as follows:

- The Cayley tree: starting from a central site $i = 0$, one builds a first shell of $k+1$ neighbors. Then each of the first shell spins is connected to k new neighbors in the second shell etc. until one reaches the Lth shell which is the boundary. There is no overlap among the new neighbors, so that the graph is a tree.

- The random graph with fluctuating connectivity: for each pair of indices (ij), a link is present with probability c/N and absent with probability $1 - c/N$. The number of links connected to a point is a random variable with a Poisson distribution, its mean being equal to c.

- The random graph with fixed connectivity, equal to $k+1$, called the *Bethe lattice*. The space of allowed graphs are all graphs such that the number of links connected to each point is equal to $k+1$. The simplest choice is the case where every such graph has the same probability.

There are several ways to calculate the energy density given by eq.(2.37) and in the following will adopt a formalism developed by Mazard and Parisi [82].

Let us introduce an intermediate object which is a spin glass model with N spins, on a slightly different random lattice, where q randomly chosen *cavity* spins have only k neighbors, while the other $N - q$ spins all have $k+1$ neighbors as in part A of Fig.(2.6). We call such a graph a $G_{N,q}$ cavity graph. The cavity spins are fixed, their values are $s_1, \ldots, s_q$. The global ground state energy of the corresponding spin glass model depends on the values of the cavity spins.

The intermediate construction of $G_{N,q}$ turns out to be helpful. The basic operations which one can perform on cavity graphs are the following (Figures A, B, C from Fig.(2.6)):

- *Iteration*: By adding a new spin s_0 of fixed value into the cavity, connecting it to k of the cavity spins say $s_1, \ldots, s_k$, and optimizing the values of these k spins, we change the $G_{N,q}$ into a $G_{N+1,q-k+1}$ graph, so we have $\delta N = 1, \quad \delta q = -k+1$.

- *Link addition*: By adding a new link between two randomly chosen cavity spins s_1, s_2, and optimizing the values of these two spins we change a $G_{N,q}$ into a $G_{N,q-2}$ graph, so $\delta N = 0, \quad \delta q = -2$.

- *Site addition*: By adding a new spin s_0 into the cavity, connecting it to $k+1$ of the cavity spins say $s_1, \ldots, s_{k+1}$, and optimizing the values of the $k+2$ spins $s_1, \ldots, s_{k+2}$, we change a $G_{N,q}$ into a $G_{N+1,q-k-1}$ graph, so $\delta N = 1, \quad \delta q = -k-1$.

It is straightforward to see that if we start from a $G_{N,2(k+1)}$ cavity graph and perform $k+1$ link additions, we get a $G_{N,0}$ graph, *i.e.* the original spin glass problem with N spins. Starting from the same $G_{N,2(k+1)}$ cavity graph and performing two site additions we get a $G_{N+2,0}$ graph, *i.e.* the original spin glass problem with $N+2$ spins. Therefore the variation in the global ground state energy when going from N to $N+2$ sites, $(E_{N+2} - E_N)$, is related to the average energy shifts $\Delta E^{(1)}$ for a site addition and $\Delta E^{(2)}$ for a link addition, through:

$$E_{N+2} - E_N = 2\Delta E^{(1)} - (k+1)\Delta E^{(2)}. \tag{2.38}$$

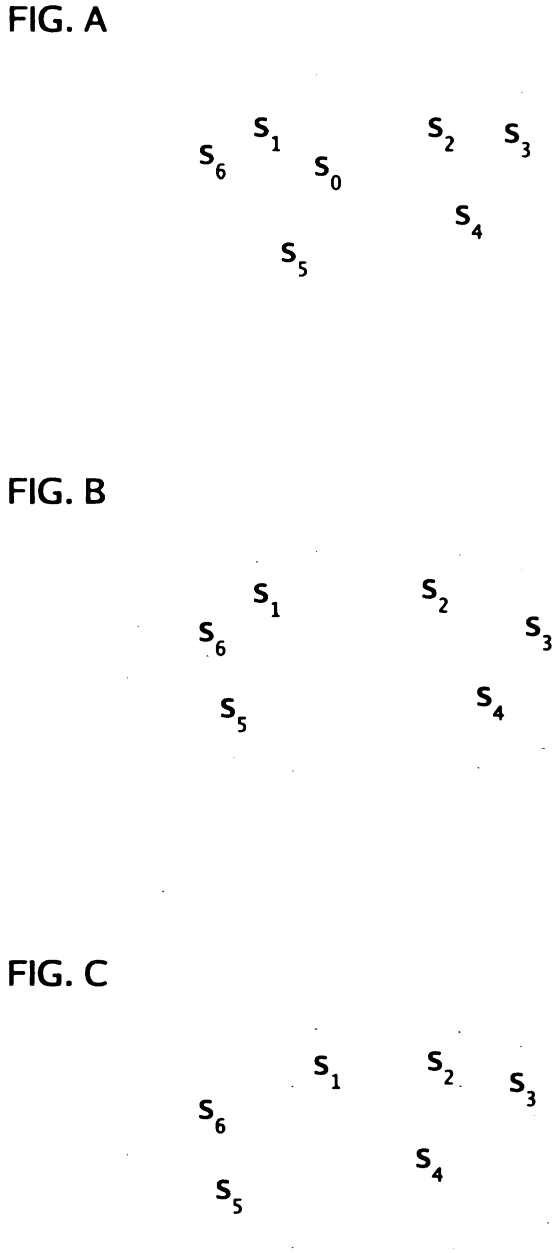


Figure 2.6: An example, for the case $k = 2$, of a $G_{N,6}$ cavity graph where $q = 6$ randomly chosen cavity spins have two neighbors only. Fig. A: All the other $N - 6$ spins outside the cavity are connected through a random graph şuch that every spin has $k + 1 = 3$ neighbors. Fig. B: Starting from $G_{N,6}$ cavity graph we can create a $G_{N+2,0}$ graph by adding two sites. Fig. C: Starting from $G_{N,6}$ cavity graph we can create a $G_{N,0}$ graph by adding three links. [82].

Using the fact that the total energy is asymptotically linear in N, the energy density of the ground state is,

$$U = \lim_{N\to\infty} \frac{E_N}{N} = \frac{E_{N+2} - E_N}{2} = \Delta E^{(1)} - \frac{k+1}{2}\Delta E^{(2)}. \qquad (2.39)$$

An intuitive interpretation of this result given by 2.39 is that in order to go from N to $N+1$ one should remove $(k+1)/2$ links (the energy for removing a link is minus the energy for adding a link) and then add a site.

When $q/N \ll 1$, generically, the distance on the lattice between two generic cavity spins is large (it diverges logarithmically in the thermodynamic limit). It is thus resonable to assume that various cavity spins become uncorrelated. This is the basic assumption of the replica symmetric solution and it is treated in a detailed way in the next chapter.

Chapter 3

Phase Transitions in Random Combinatorial Problems

In the last decade tools from statistical physics of frustrated systems are being used to gain a better understanding of complexity, by mapping the optimization problems onto the study of the ground state of disordered models [83], as we showed in Chapter 1. Concepts and methods from statistical physics have been applied to the analysis of the typical properties of optimization problems and also to different search algorithms in which temperature is an important control parameter [67]. In Section (3.1) we give an introduction to phase transitions in random combinatorial problems discussed in Chapter 1, Section (3.2) presents a more detailed discussion of K-SAT and Section (3.3) discusses replica symmetry breaking in spin glasses.

3.1 Phase Transitions

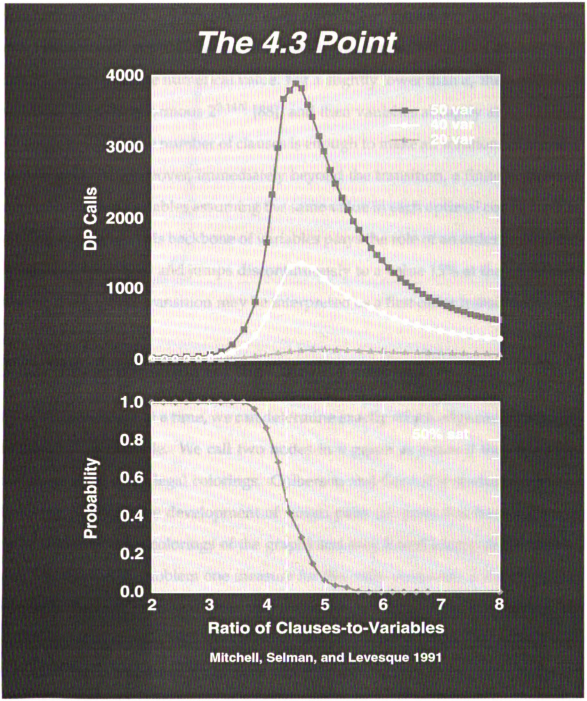
The observation of threshold phenomena in random mathematical and computer science problems has shown that not only ground state properties but also the critical behavior at glassy phase transitions occurring in disordered systems could be relevant for complexity theory, due to the so called intractability concentration phenomena [60].

For many NP-complete problems one or more order parameters can be defined, and hard instances occur around particular critical values of these order parameters. In addition, such critical values form a boundary that separates the space of problems into two regions. One region is under-constrained so the density of solutions is high, thus making it relatively easy to find a solution. The other region is over-constrained and very unlikely to contain a solution. If there are solutions in this over-constrained region, they have such a deep local minima that any reasonable algorithm is likely to find it. If there is no solution, then a backtrack search can usually establish this easily, since potential solution paths are usually cut off early in the search. Really hard problems occur on the boundary between these two regions, where the probability of finding a solution is low but non-negligible. At this point there are typically many local minima corresponding to almost solutions separated by high energy barriers. These almost solutions form deep local minima that may often trap search methods that rely on local information. Because it is possible to locate a region where hard problems occur, it is possible to predict whether a particular problem is likely to be easy to solve.

3.1.1 Phase transition for the SAT problem

The most widely used complete algorithms (*i.e.*, algorithms which are able to either find a satisfying assignment, or to prove that there is no such assignment) were developed by Davis and Putnam [27], improved later by Longemann and Loveland and are known in the literature as (DPLL) algorithms. DPLL operates by directed trial and error, using a search tree made of nodes connected through edges as follows: (1) A node corresponds to the choice of a variable. Depending on the value of the latter, DPLL takes one of the two possible edges; (2) Along an edge, all logical implications of the last choice made are extracted; (3) DPLL goes back to step (1) unless a solution is found or a contradiction arises; in the latter case, DPLL backtracks to the closest incomplete node, inverts the attached variable and goes to step (2). If all the nodes carry two descendent edges, unsatisfiability is proven.

The probability $P(\alpha)$ that an instance is satisfiable as a function of α, for different sizes N is shown in the lower part of Fig.(3.1). In addition to being a decreasing function of α as expected from above, a striking phenomenon happens as N grows. An abrupt decrease of the probability takes place at a critical value $\alpha_c \approx 4.3$. Instances with less than $\alpha_c N$ clauses are almost surely satisfiable, whereas the ones with more than $\alpha_c N$ clauses almost never have any solution. The upper part of Fig.(3.1) shows the median time necessary to solve a random $3-SAT$ instance, that is to find a solution (below the threshold), or check that there is none (above the threshold). The general pattern of the complexity as a function of α, namely easy, hard and less hard resolutions is a generic feature valid for commonly used algorithms for hard computational problems. Mathematicians have been able to establish rigorous bounds on the threshold, $3.14 \leq \alpha_c \leq 4.51$, but the exact value seems out of reach with the available probabilistic techniques [1], [2]. Approximation methods developed by physicists in the course of the study of phase transitions allow not only to estimate the value of the threshold but also to unveil the microscopic structure of the solutions and the mechanism leading to their disappearance. We already showed in Chapter 1 the mapping between the SAT problem and statistical physics: we translated the SAT problem with its ingredients (Boolean variables, clauses) to statistical physics (Ising spins, interactions). The main idea is to introduce an energy function, which is equal to the number of unsatisfied clauses for each variables-spins configuration, and study the ground-state properties. The satisfiability of the $3-SAT$ problem is therefore equivalent to the vanishing of ground-state energy. The physical scenario that was obtained for ran-


The 4.3 Point

Mitchell, Selman, and Levesque 1991

Figure 3.1: The 4.3 point for the 3-SAT problem

dom 3-SAT is the following [15], [14]. For $\alpha > \alpha_c$ the ground-state energy becomes positive with probability one, whereas it vanishes for $\alpha < \alpha_c$. Finding the exact solution of random $3 - SAT$ remains an open problem. However, combining some exact results with approximated techniques it was found [88] $\alpha_c \approx 4.48$, which is just 5% larger than the numerical value. For α slightly lower than α_c the number of solutions remains enormous $2^{0.14N}$ [88], and then vanishes abruptly at α_c. Hence, increasing slightly the number of clauses is enough to make all solutions disappear simultaneously. Moreover, immediately beyond the transition, a finite fraction of over-constrained variables assuming the same value in each optimal configuration abruptly emerges. This backbone of variables plays the role of an order parameter: it vanishes for $\alpha > \alpha_c$ and jumps discontinuously to a value 15% at the transition. Hence, the $3 - SAT$ transition may be interpreted as a first order transition.

3.1.2 Phase transition for the Coloring problem

By adding one edge at a time, we can determine exactly which edge causes a graph to become uncolorable. We call two nodes in a graph as *frozen* if they preserve the same color in all legal colorings. Culberson and Gent [25] studied for the 3-Coloring problem, the development of frozen pairs (so pairs that have the same color under all three colorings of the graph) and they found a jump at the threshold. For the K-SAT problem one measure for the order parameter is the backbone which is the number of variables that are frozen to a particular value under all satisfying assignments. For the Coloring problem it is more difficult to find and measure the order parameter because of the symmetry of solutions. If we add an edge to a colorable graph and then the graph becomes uncolorable, then in the graph without the edge the pair of vertices must have been colored with the same color in every coloring of the graph.

This measurement process is called *the frozen development process* and it was

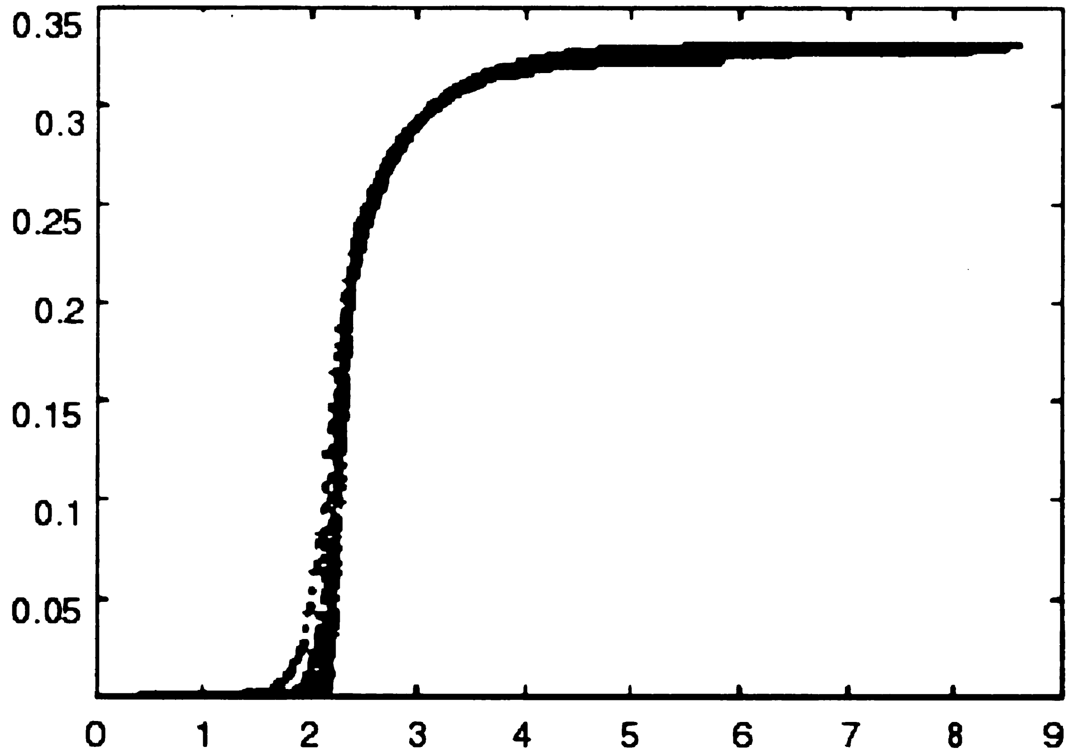


Figure 3.2: The ratio of frozen pairs to $\binom{n}{2}$ plotted against the ratio M/N, where M is the number of frozen pairs [24].

shown that this set of pairs shows an explosive jump at the threshold. By identifying pairs of vertices that are frozen, a collapsed graph was created which has the same set of 3-colorings. If the measure were to be taken with respect to minimization of violated edges, then this sudden drop would mostly dissapear. This is due to the fact that for large sets of edges removing any of the violated edges from the threshold would result in a distinct set of 3-colorings. This could explain the discrepancy between the jump that is observed and the lack in certain statistical mechanics analysis of the 3-coloring phase transition [106]. From Fig.(3.2) we see that the freezing process is not gradual; the addition of one edge can cause many pairs to become frozen simultaneously. Fig.(3.2) and Fig.(3.3) shows how the number of frozen and free pairs grows as the edge density increases. The full development process was run on 100 permutations at each value of $N = 50, 75, \ldots, 225$ in step of 25. This is typical for a phase transition, and the sharpness suggests that there

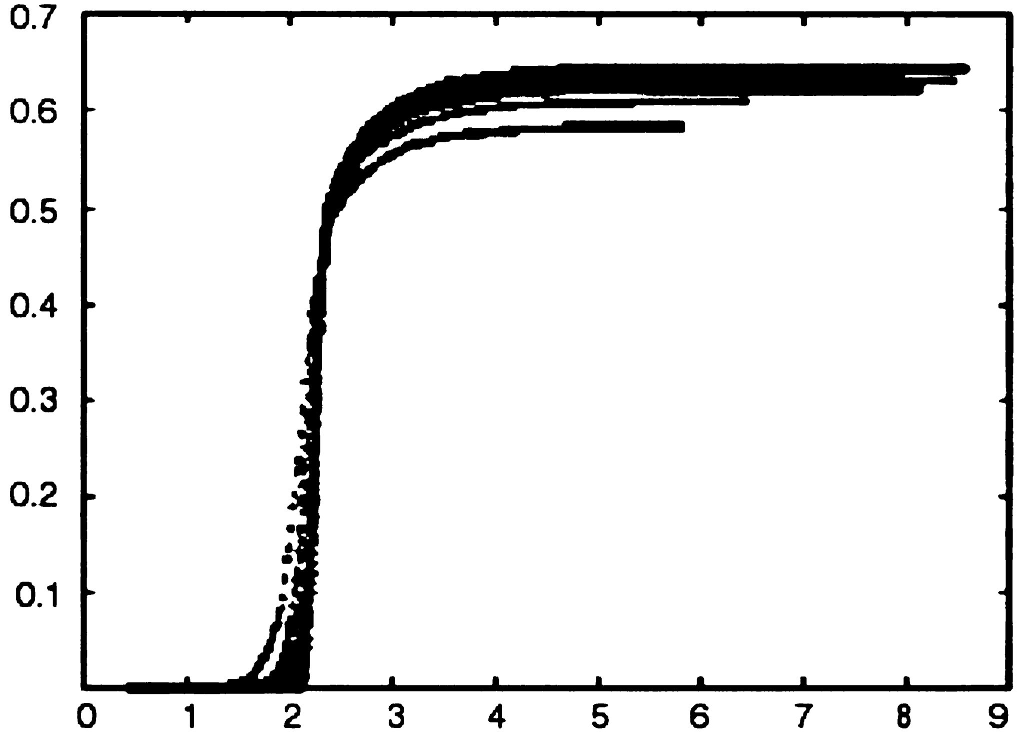


Figure 3.3: The ratio of free pairs to $\binom{n}{2}$ plotted against the ratio M/N, where M is the number of frozen pairs [24].

will be a discontinuity at $N \to \infty$.

3.1.3 Phase transition for the Vertex Cover problem

Let us consider a random graph with N vertices $V = 1, 2, \ldots, N$ where any pair of vertices is connected randomly and independently by an edge with probability p. The expected number of edges becomes $p\binom{N}{2} = pN^2 + O(N)$, and the average connectivity of a vertex is $p(N-1)$. As we defined in Chapter 1 the mathematical formulation of the VC problem is as follows: Consider an undirected graph $G = (V, E)$ with N vertices $i \in V = 1, 2, \cdots, N$ and edges $(i, j) \in E \subset V \times V$. A vertex cover is a subset $V_{vc} \subset V$ of vertices such that for all edges $(i, j) \in E$ there is at least one of its endpoints i or j in V_{vc}. We call the vertices that are in V_{vc} *covered,* whereas the vertices in its complement $V \setminus V_{vc}$ are called *uncovered.*

We map the random graph to a disordered spin systems and we seek to minimize the Hamiltonian:

$$H(\{S_i\}, \{J_{ij}\}) = \frac{1}{2} \sum_{i,j=1}^{N} J_{ij} \delta_{S_i,-1} \delta_{S_j,-1} \tag{3.1}$$

where $J_{ij} = 1$ whenever there is an edge between vertex i and vertex j and zero otherwise. The covering state of the vertices is mapped to a configuration of N Ising spins: $S_i = 1$ if $i \in V_{vc}$ so if vertex i is covered and $S_i = -1$ if $i \in V - V_{vc}$ so if vertex i is uncovered. The vertex-cover decision problem asks whether there are covers of fixed cardinality $xN = |V_{vc}|$ or in other words, we are interested if it is possible to cover all the edges of G by covering xN suitably chosen vertices. We have to minimize the Hamiltonian H under the constraint:

$$\frac{1}{N} \sum_{i=1}^{N} S_i = 2x - 1 \tag{3.2}$$

which fixes the cardinality of the cover set, or in physical terms, it fixes the magnetization of the Ising spin systems. If this make the energy equal to zero, then there are no uncovered edges and the decision problem can be positively answered. If a minimal energy greater than zero is found, there is no vertex cover of cardinality xN, case in which the minimum energy represents the minimum number of uncovered edges.

When the number xN of covering marks is lowered (the average connectivity c, which can be seen as edge concentration is kept constant), the model is expected to undergo a coverable-uncoverable transition. Using probabilistic tools, rigorous lower and upper bounds for this threshold [47] and the asymptotic behavior for large connectivities have been deduced [43]. In Fig.(3.4) the average ground-state energy density and the probability $P_{cov}(x)$ that a graph is coverable with xN vertices are shown for different problem sizes N = 25, 50, 100. The average connectivity considered is $c = 2$, but qualitatively equivalent results are found for other values of c too. The results were obtained using the branch-and-bound algorithm presented with data averaged over 10^3 $(N = 100)$ to 10^4 $(N = 25, 50)$ samples [56] [111]. As expected, the value of $P_{cov}(x)$ increases with the fraction of covered vertices. Growing the size of the graphs , the curves become steeper. This indicates that in the thermodynamic limit $N \to \infty$, a sharp threshold at $x_c \approx 0.39$ appears. Above x_c a graph is coverable with probability tending to one in the large N limit, while below x_c it is almost surely uncoverable. In the language of a physicist, a phase transition from a coverable phase to an uncoverable phase occurs and it is usually denoted as the coveruncover transition.

3.1.4 Phase transition for the Number Partition Problem

In Chapter 1 we defined the number partitioning problem as follows: given a sequence of positive real numbers $a_1, a_2, ..., a_N$ the NPP consists of partitioning them

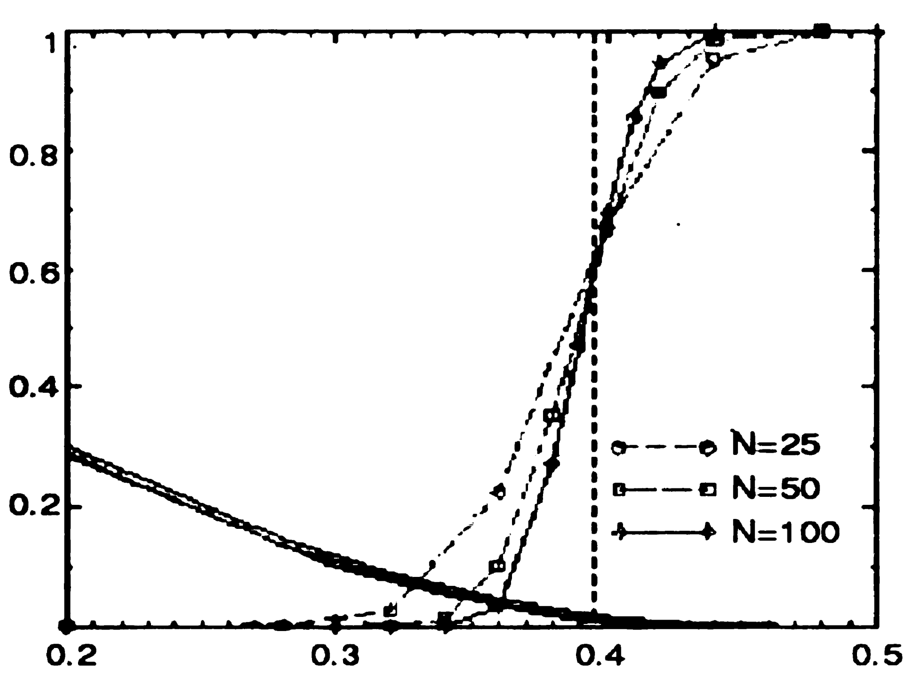


Figure 3.4: Probability $P_{cov}(x)$ that a cover exists for a random graph ($c = 2$) as a function of the fraction x of covered vertices. The result is shown for three different system sizes $N = 25, 50, 100$ (averaged for $10^3 - 10^4$ samples). Lines are guides for the eyes only [57].

into two disjoint sets A_1, A_2 such that the difference $|\sum_{a_j \in A_1} a_j - \sum_{a_j \in A_2} a_j|$ is minimized. When the difference between the two sums is less than or equal to one then the partition is called *perfect partition*, otherwise is called *imperfect partition*.

Fu [44] claims that in the random number partitioning problem no phase transition of any kind is found. If Fu were right, NPP would be a notable exception to the observation that many NP-complete problems do have a phase transition, parameterized by a control parameter that separates the easy from the hard to solve instances [21].

Mertens [79], [80], computed the expected number of perfect partition and for this they considered the binary representation of the n integers that are being partitioned. They considered the case when in the bags they have even sums where each bag must add to the same target sum. In order to develop an annealed theory, the probabilities are averaged independently over the different digit positions. On average we expect half of the possible partitions to add up to a number with the same parity as the least significant bit to the target sum. The expected number of perfect partitions, reads:

$$< Sol >= \frac{2^n}{l} \tag{3.3}$$

where it was considered that the numbers were drawn uniformly and at random from $(0.l]$. The expected number of solutions $< Sol >$ and the problem size n defines a *constrainedness* parameter:

$$\kappa = 1 - \frac{\log_2 < Sol >}{n} \tag{3.4}$$

If κ is small then the problem is under-constrained and there are a large number of solutions compared with the problem size. If κ is large, the problem is over-constrained and only few solutions exist. The transition gets sharper with increasing n. Finite-size scaling lead Gent and Walsh to find $\kappa_c = 0.96$ for $N \rightarrow \infty$. This result is in contradiction with Fu's claim but this type of phase transition can indeed be found in the statistical mechanics of the number partitioning problem. Substituting the annealed value of solutions $\langle Sol \rangle$ from Eq.(3.3) into Eq.(3.4) we get:

$$\kappa = \frac{\log_2 l}{n} \tag{3.5}$$

To estimate the critical value of κ, Gent and Walsh [50] found the probability that a bag with an even sum has a perfect partition as a function of κ for n varying from 6 to 30 and $\log_2 l$ from 0 to $2n$. They generated 1000 problems at each value of A and n. Similar results are seen using bags with an odd sum, and bags with both odd and even sums. As predicted, a phase transition occurs around $\kappa \approx 1$ with the transition sharpening as N increases. Finite size scaling methods were applied to determine how the probability scales with problem size [9]. Around some critical point, it was predicted that problems of all sizes will be indistinguishable except for a change of scale. This suggests:

$$Prob(perfect\ partition) = f((\frac{\kappa - \kappa_c}{\kappa_c})n^{1/\nu}) \tag{3.6}$$

where f is a fundamental function, κ_c is the critical point and $n^{1/\nu}$ provides the change of scale. The fraction $(\frac{\kappa - \kappa_c}{\kappa_c})$ plays the role of the reduced temperature $(T - T_c)/T_c$ in physical systems. Eq.(3.6) has a fixed point where $\kappa = \kappa_c$ and for all N, the probability is the constant value $f(0)$. In order to estimate κ_c the fixed point was taken where the spread in probabilities is the smallest, which gives $\kappa_c = 0.96 \pm 0.02$. To estimate ν the assumption that Eq. (3.6) holds at the point 50% was made, which gives $\nu = 1 \pm 0.3$. Then we can define a rescaled parameter:

$$\gamma = \frac{\kappa - 0.96}{0.96} n \tag{3.7}$$

Gent and Walh plotted the probability of a perfect partition as a function of γ. They found that the finite-size scaling provides both a simple and accurate model for the scaling of probability with problem size. A similar rescaling of the constrainedness parameter κ describe the phase transition in satisfiability [48], constrained satisfaction [67] and traveling salesman problem [49].

3.2 More details on K-SAT problem

3.2.1 Known results for the K-SAT problem

As noted in Section 3.1.1, when the number of clauses becomes of the same order as the number of variables ($M = \alpha N$) and in the large N limit (the *thermodynamic limit*), the K-SAT shows a striking threshold phenomena. Numerical experiments have shown that the probability of finding a solution (*i.e.* a correct Boolean assignment) falls abruptly from one down to zero when α crosses a critical value $\alpha_c(K)$ of the number of clauses per variable. Above $\alpha_c(K)$ all clauses cannot be satisfied any longer and we are interested in minimizing the number of unsatisfied clauses, which is the optimization version of K-SAT referred as MAX-K-SAT which we already introduced in Chapter 1. Near $\alpha_c(K)$, heuristic search algorithms get trapped in non-optimal solutions and a slow down effect is observed. We give a brief review of known results for the K-SAT problem that have been obtained in the framework of complexity theory.

- For $K = 2, 2 - SAT$ problem belongs to the class P of polynomial problems. For $\alpha > \alpha_c$, $MAX - 2 - SAT$ is NP-complete [46]. Mapping of $2 - SAT$ on graph theory allows to derive rigorously the threshold value $\alpha_c = 1$ and an explicit $2 - SAT$ polynomial algorithm has been developed [5].

- For $K \geq 3$, both K-SAT and MAX-K-SAT belong to the NP-complete class. From a rigorous point of view only upper and lower bounds of $\alpha_c(K)$ are known [29], [51]. Finite size scaling techniques, have been allowed one to determined precisely the numerical values of α_c for $K = 3, 4, 5, 6$ [67].

- For $K \gg 1$, clauses become decoupled and an asymptotic expression $\alpha_c \approx 2^K \ln 2$ can be found.

3.2.2 Statistical mechanics of the K-SAT problem

For $\alpha = M/N > 0$, K-SAT can be cast in the framework of statistical mechanics of random diluted systems by the identification of an energy-cost function $E(K, \alpha)$ equal to the number of violated clauses [66], [88], [?], . The study of its ground state allows one to address the optimization version of the K-SAT problem as well as to characterize the space of solutions by its typical entropy, *i.e.*, the degeneracy of the ground state. The vanishing condition on the ground state energy for a given K corresponds to the existence of a solution to the K-SAT problem and thus identifies a critical value $\alpha_c(K)$ below which random formulas are satisfiable with probability one. For $\alpha > \alpha_c(K)$, the ground state energy becomes nonzero and gives information on the maximum number of satisfiable clauses, *i.e.*, on the MAX-K-SAT problem. As already defined in Eq.(3.8), the energy cost function (the number of violated clauses) is given by:

$$E[C, S] = \sum_{l=1}^{M} \delta(\sum_{i=1}^{N} C_{l_i} S_i - K) \tag{3.8}$$

subject to the constraints $\sum_{l=1}^{M} C_{l_i} S_i = K$ and $\sum_{i=1}^{N} C_{l_i}^2 = K, \quad l = 1 \ldots M$. To compute the ground state energy, a fictitious temperature $1/\beta$ is introduced to regularize all mathematical expressions and $\beta \to \infty$ is taken at the end of the calculations. In this way we compute the free energy density at inverse temperature β, averaged over the clauses distribution:

$$F(\beta) = -\frac{1}{\beta N}\overline{\ln Z[C]} \tag{3.9}$$

with

$$Z[C] = \sum_{S_i} \exp(-\beta E[C,S]) \tag{3.10}$$

The energy given by Eq.(3.8) is self averaging and can be obtained from the free energy defined in Eq.(3.9). The overline denotes the average over the random clauses and is performed using the *replica trick*:

$$\overline{\ln Z} = \lim_{n\to 0}\frac{\overline{Z^n} - 1}{n} \tag{3.11}$$

One prepares n replicas of the original system, evaluates the configurational average of the product of their partition function Z^n, and then takes the limit $n \to 0$. This technique, the *replica method,* is useful because it is easier to evaluate $\overline{Z^n}$ than $\overline{\ln Z}$. The typical properties of the ground state will then be recovered in the $\beta \to \infty$ limit.

3.2.3 The simplest case, $K = 1$

The $K = 1$ case can be solved either by a direct combinatorial method or using the statistical physics approach. Moreover, the $K = 1$ case allows us to check the correctness of the statistical physics approach [89]. For this case, a sample of M clauses is completely described by giving the numbers t_i and f_i of clauses that a variable S_i must be true or false respectively. Then

$$Z[t,f] = \prod_{i=1}^{N}(e^{-\beta t_i} + e^{-\beta f_i}) \tag{3.12}$$

so for the average disorder we have

$$\frac{1}{N}\overline{\ln Z[t,f]} = \sum_{t_i,f_i} \frac{M!}{\prod_{i=1}^{N} t_i! f_i!} \ln Z[\{t,f\}] =$$

$$\ln 2 - \frac{\alpha\beta}{2} + \sum_{l=-\infty}^{\infty} e^{-\alpha} I_l(\alpha) \ln[\cosh(\frac{\beta l}{2})] \tag{3.13}$$

where I_l denotes the l^{th} modified Bessel function. The ground state energy (so the average number of violated clauses) at zero temperature is:

$$E_{GS} = \frac{\alpha}{2}[1 - e^{-\alpha} I_0(\alpha) - e^{-\alpha} I_1(\alpha)] \tag{3.14}$$

Using a well known formula from thermodynamics $E = U - TS$, the ground state entropy (defined as $\frac{1}{N}\ln(number\ of\ degenerate\ ground\ states)$) at zero temperature is then found to be:

$$S_{GS} = e^{-\alpha} I_0(\alpha) \ln 2 \tag{3.15}$$

These results are recovered using the RS ansatz for all α and β when $K = 1$. The finite value of the ground state entropy may be ascribed to the existence of unfrozen spins. The non-zero value of the ground state energy is due to the presence of completely unfrozen spins of magnetizations. Fig.(3.5) shows the plots of the above energy and entropy at zero temperature.

3.2.4 Replica symmetric solutions for all K

The fraction of violated clauses is given by

$$E = -\frac{1}{N}\frac{\partial}{\partial\beta}\overline{\ln Z[C]} \tag{3.16}$$

and the ground state energy depends only upon the magnetizations of order $\pm(1 - O(e^{-z\beta}))$ if any, and such contribution can be described by the introduction of the

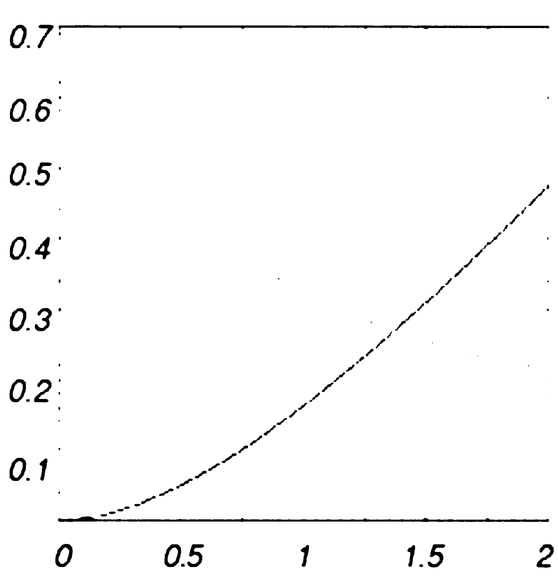


Figure 3.5: Typical fraction of violated clauses (bold line) and entropy (thin line) vs. α for $K = 1$ in the limit $N \longrightarrow \infty$ [89]

new rescaled function

$$R(z) = \lim_{\beta \to \infty} [P(\tanh(\frac{\beta z}{2}))\frac{\partial}{\partial z}\tanh(\frac{\beta z}{2})] \tag{3.17}$$

Also $R(z)$ defined in Eq.(3.17) obeys the saddle-point equation:

$$R(z) = \int_{-\infty}^{\infty} \frac{du}{2\pi}\cos(uz)\exp[-\frac{\alpha K}{2^{K-1}} + \alpha K \int_0^{\infty} \prod_{l=1}^{K-1} dz_l R(z_l)\cos(u\ min(1, z_1, \ldots z_{K-1}))] \tag{3.18}$$

Then, the corresponding ground-state energy reads

$$E_{GS}(\alpha) = \alpha(1-K)\int_0^{\infty}\prod_{l=1}^{K} dz_l R(z_l) min(1, z_1, \ldots z_K) + \\ \frac{\alpha K}{2}\int_0^{\infty}\prod_{l=1}^{K-1} dz_l R(z_l) min(1, z_1, \ldots, z_{K-1}) - \int_0^{\infty} dz R(z) z \tag{3.19}$$

The saddle-point Eq.(3.18) is a self-consistent identity for $R(z)$ in the range $z \in [0,1]$ only. Outside this interval, it is just a definition of the functional order parameter R:

$$R(z) = \sum_{l=-\infty}^{\infty} e^{-\gamma_1} I_l(\gamma_1)\delta(z-l) \qquad (3.20)$$

with γ_1 given by

$$\gamma_1 = \alpha K [\frac{1 - e^{-\gamma_1} I_0(\gamma_1)}{2}]^{K-1} \qquad (3.21)$$

Inserting Eq.(3.18) in Eq.(3.17) we obtain

$$E_{GS}(\alpha) = \frac{\gamma_1}{2K}(1 - e^{-\gamma_1} I_0(\gamma_1) - K e^{-\gamma_1} I_1(\gamma_1)) \qquad (3.22)$$

So, in the RS context, the SAT to UNSAT transition corresponds to the emergence of peaks centered at $x = \pm 1$ with finite weights, that is to a transition from $\gamma_1 = 0$ to $\gamma_1 > 0$. This result is in good agreement with our results using combinatorial methods presented in Chapter 4. In addition to Eq.(3.20), there exists other *RS* solutions to the saddle-point Eq.(3.16). For instance if we choose $z_0 = 1/2$, the insertion process ends up after two iterations and generates Dirac peaks centered in all integer and half-integer numbers. More generally, for any integer $p \geq 1$, we can define the solution of Eq.(3.18):

$$R(z) = \sum_{l=-\infty}^{\infty} r_l \delta(z - \frac{l}{p}) \qquad (3.23)$$

having exactly p peaks in the interval $[0,1]$, whose centers are $z_l = l/p, \; l = 0, \ldots, p-1$.

The self-consistency of Eq.(3.21) admits the solution $\gamma_1 = 0$ for any α. When $K = 2$, there is another solution $\gamma_1(\alpha) > 0$ above $\alpha = 1$ which maximizes E_{GS} [83]. Therefore, the RS theory predicts that $E_{GS} = 0$ for $\alpha \leq 1$ and increases continuously when $\alpha > 1$, giving back the rigorous result at $\alpha_c(2) = 1$. The transition taking place at α_c is of second order with respect to the order parameter. As far as $2 - SAT$ is concerned the value of the threshold is correctly predicted and the RS solution is exact for any value of α.

For the same case $K = 2$ but for $\alpha \geq \alpha_c$, the vanishing of the exponentially large number of solutions that were present below the threshold is surprisingly abrupt. It was shown [88] that this transition is due to the abrupt appearance of contradictory logical loops in all solutions at $\alpha = \alpha_c$ and not to the progressive decreasing of the number of solutions down to zero at the threshold.

It was found [89], [83] ,that for $K \geq 3$ and for sufficiently big values of α, the RS entropy is negative, whereas of course it has to be the logarithm of an integer number. This result is a consequence of replica symmetry breaking (RSB) effects. Despite the general complexity of such an approach in diluted models and the technical difficulty of the $K - SAT$ problem, recent attempts in this direction are successful [81]. In the next section we will describe the physical meaning of (RSB) for spin glasses without focusing on mathematical details.

3.3 Physical meaning of breaking the symmetry

Initially was suspected that the negative entropy might have been caused by the inappropriate exchange of limits $n \rightarrow 0$ and $N \rightarrow \infty$ in deriving the free energy. The correct order is $N \rightarrow \infty$ after $n \rightarrow 0$, but we take the limit $N \rightarrow \infty$ first so that the method of steepest descent is applicable. However it has been established that the assumption of replica symmetry $q_{\alpha\beta} = q, \ \forall\ (\alpha, \beta) : \ \alpha \neq \beta$ (the parameter q is introduced below) is the real source of the trouble.

3.3.1 Replica symmetric solution for the Sherington-Kirkpatrick model

In order to understand the physical meaning of replica symmetry and replica symmetry breaking we introduce a well studied infinite-range spin glass model, the

Sherington-Kirkpatrick (SK) model. For the SK model the Hamiltonian is given by:

$$H_J[s] = -\sum_{1\leq i<j\leq N} J_{ij}s_is_j - h\sum_i s_i \tag{3.24}$$

where h is the magnetic field and the J's are independent random variables with zero mean and variance $1/N$:

$$\overline{J_{ij}} = 0, \quad \overline{J_{ij}^2} = 1/N, \quad J_{ij} = J_{ji}$$

The factor $1/N$ has been chosen such that at fixed $\beta = 1/k_BT$ the total energy is proportional to N and therefore the energy density is N-independent.

For fixed J's we expect that in the high temperature phase the local magnetization $m_i \equiv \langle s_i\rangle$ is different from zero only if a magnetic field is present and it vanishes when the magnetic field goes to zero; on the other hand we would expect that in the low temperature region there should be some freezing of the spins (like in ferromagnets) in the position which is mostly favored energetically, so m_i should be different from zero also at $h = 0$. But the local magnetization m_i depends on J's and will sometimes be positive and sometimes negative (actually the global magnetization density $(1/N)\sum_i m_i = 0$ at $h = 0$) so it is convenient to characterize the system in terms of the quantity

$$q_{EA} = \frac{1}{N}\sum_{i=1}^{N} m_i^2 \tag{3.25}$$

called the Edwards-Anderson parameter [32]. Here we will refer briefly only on the results when we start from the partition function [68]:

$$Z_n = \sum_J P(J)\sum_s \exp\sum_{a=1}^{N}\left(\beta\sum_{i<k} J_{ij}s_i^as_j^a + \beta h\sum_i s_i^a\right) \tag{3.26}$$

After calculations (which use properties of Gaussian integrals) it was found that [32]

$$Z_n = \int \prod_{a<b} (dQ_{a,b}(N\beta^2/2\pi)^{1/2}) \exp(-NA[Q]) \tag{3.27}$$

with

$$A[Q] = -\frac{n\beta^2}{4} + \frac{\beta^2}{2} \sum_{1\leq a<b\leq n} (Q_{a,b})^2 - \ln Z[Q] \tag{3.28}$$

$$Z[Q] = \sum_{S} \exp(-\beta H[Q,S]) \tag{3.29}$$

$$H[Q,S] = -\beta \sum_{1\leq a<b\leq n} Q_{a,b} S_a S_b - h \sum_{a=1}^{n} S_a \tag{3.30}$$

where the matrix Q is an $n \times n$ symmetric matrix, zero on the diagonal, and the sum over S goes over the 2^n configurations of the variables S_a, $a = 1, \cdots, n$ with $S_a = \pm 1$. Eq.(3.30) suggests that Z_n can be computed through a saddle-point method which gives:

$$f_n = -\frac{1}{\beta N n} \log(Z_n) = \frac{1}{\beta n}\, min\ A[Q] \tag{3.31}$$

So we must find the solution of the $n(n-1)$ saddle-point equations $\partial A/\partial Q_{a,b} = 0$ and after we find the saddle-point matrix Q_{sp}, the free energy is obtained using Eq.(3.31):

$$f = \lim_{n\to 0} (1/\beta n) A[Q_{sp}] \tag{3.32}$$

The equation $\partial A/\partial Q_{a,b} = 0$ can be written under the form of self-consistency equations

$$Q_{a,b} = \langle S_a S_b \rangle_Q = \overline{\langle S_a S_b \rangle}, \quad a < b \tag{3.33}$$

where the expectation value $\langle\rangle_Q$ is taken with respect to the single site Hamiltonian $H[Q,S]$.

The function $A[Q]$ is left invariant when we exchange some of the lines (or rows) of the matrix Q and therefore the group of permutations of n elements ($P_n \quad or \quad S_n$) is a symmetry of the problem (all replicas are equivalent) and we call this group the replica group. For positive integer n the minimum A can be found if the matrix Q has the following form [59]

$$\forall a,\ b\ \ a \neq b:\ Q_{a,b} = q; \quad \forall a : Q_{a,a} = 0 \tag{3.34}$$

This form of the matrix Q is the only one which is left invariant by the action of the replica group and is therefore the natural solution, usually named the replica symmetric solution. Its properties are studied in detail in [105] and here we will show only the results.

If we analitycally continue the solution of the equation $dA/dq = 0$, up to $n = 0$ we find for q

$$q = \int_{-\infty}^{\infty} dz/(2\pi)^{1/2} \exp(-z^2/2) \tanh^2(\beta z q^{1/2} + \beta h) \tag{3.35}$$

and for the free energy

$$f = -(\beta/4)(1-q)^2 - \int_{-\infty}^{\infty} dz/(2\pi)^{1/2} \exp(-z^2/2) \ln(2\cosh(\beta z q^{1/2} + \beta h)) \tag{3.36}$$

At zero magnetic field Eq.(3.36) has only the solution $q = 0$ for $1/\beta \equiv T > T_c = 1$, and for $T < T_c$ there is another solution (the physical one) where q is different from zero. So, there is a phase transition at $T = 1, \ \ h = 0$ while there is no transition at $h \neq 0$.

Although everything looks okay, detailed computation made by Sherington and Kirkpatrick [105] shows that the entropy becomes negative at small temperature. At zero temperature $S(0) = -1/2\pi \approx -1.7$ which is unphysical so in this case the replica symmetric method leads to a disaster. On the other hand the valu

met

tem

tem

wh

wh

3.3

Re

pa

pre

As

wh

value of many thermodynamic functions computed within the above replica symmetric solution do not disagree too much with the numerical data, even at low temperature where it is incorrect; for example the internal energy per spin at zero temperature (the ground state energy) gives

$$U(0) = -(2/\pi)^{1/2} \approx -0.798 \tag{3.37}$$

while numerical simulations [68], [59] give:

$$U(0) = -0.76 \pm 0.01 \tag{3.38}$$

which are not too different from each other.

3.3.2 Experimental evidence of replica symmetry breaking

Replica symmetry breaking affects the equilibrium properties of the system and in particular the magnetic susceptibility. For example let us consider a system in the presence of an external constant magnetic field, with the Hamiltonian given by:

$$H[s] = H_0[s] + \sum_i hs_i \tag{3.39}$$

As soon as replica symmetry is broken we can define two magnetic susceptibilities which are different:

- The magnetic susceptibility that we obtain when the system is constrained to remain in a valley. In the limit of zero magnetic field this susceptibility is given by $\chi_{oq} = \beta(1 - q_{EA})$.

- The total magnetic susceptibility (the system is allowed to change state as an effect of the magnetic field). In the limit of zero magnetic field this suscepti-

Fig
ter
pro

Bo
we
ter
fec
ins
we
sec
en
wh
so
an

the

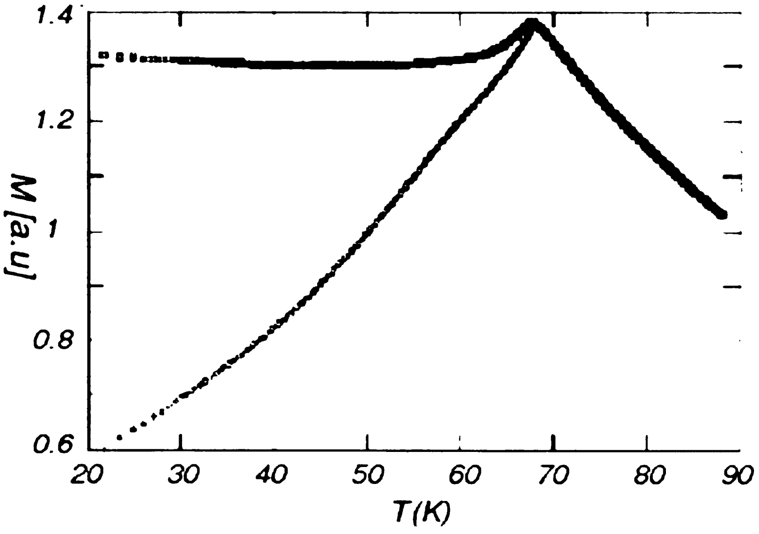


Figure 3.6: FC- and ZFC-magnetization (higher and lower curve respectively) vs. temperature of $Cu(Mn13.5\%)$, $H = 1$ Oe. For such a low field the magnetization is proportional to susceptibility [28].

bility is given by $\chi_{eq} = \beta \int dqP(q)(1-q) \leq \beta(1-q_{EA})$.

Both susceptibilities are experimentally observable. The first susceptibility is when we measure if we cool in zero field and then add a very small magnetic field at low temperature. The field should be small enough in order to neglect non-linear effects. In this situation, when we change the magnetic field, the system remains inside a given state and it is not forced to jump from a state to another state and we measure the zero field cooled (ZFC) susceptibility, that corresponds to χ_{oq}. The second susceptibility can be approximately measured doing a cooling in the presence of a small field: in this case the system has the ability to choose the state which is most appropriate in presence of the applied field. This susceptibility, the so called field cooled (FC) susceptibility is nearly independent of the temperature and corresponds to χ_{eq} .

To conclude, we can identify χ_{oq} and χ_{eq} with the ZFC susceptibility and with the FC susceptibility respectively. The experimental plot of the two susceptibilities is shown in Fig.(3.6). They are clearly equal in the high temperature phase while they differ in the low temperature phase. The difference among the two susceptibilities is a signature of replica symmetry breaking and it can be explained in this framework. This phenomenon is due to the fact that a small change in the magnetic field pushes the system into a slightly metastable state, which may decay only with a very long time scale. This happens only if there are many states which differ one from the other by a very small amount in free energy.

Chapter 4

Geometric Approach

As is evident from the first three chapters, there is intense interest in the relations between statistical physics and computational complexity, from both the computer science and physics communities [30, 83, 85]. The physics approach to hard problems is based on replica methods which is quite complex. In this chapter we develop methods to study hard problems analytically using methods similar to the message passing procedures that were discussed in Chapter 2. In the first part we introduce the network algorithms that are going to be used in applications from biology. In the second part we cover networks algorithms that have evolved from applications of statistical physics to hard computational problems.

4.1 Introduction

The $k-$connectivity and $k-$core problems [99] have attracted interest, for example in designing redundant networks [69]. $k-$connectivity is the generalization of the conventional connectivity percolation problem to the requirement of $k-$fold connectivity. That is, a graph is k-connected if for each pair of vertices in the graph there exist at least k mutually independent paths connecting them. The $k-$core of a graph is the largest subgraph with minimum vertex degree k. The Bethe lattice equations for the k-core were actually first derived in the context of k-bootstrap percolation [20]. k-bootstrap percolation is the percolation process found by recursively deleting all nodes, which have connectivity less than k. More recently the Bethe lattice $k-$core equations have been used to develop theories for rigidity percolation [31,91,94]. We give a brief introduction to the connectivity percolation and $g-$ rigidity equations on Bethe lattices and then describe similar percolation processes which are important in the Viana-Bray spin-glass model, the coloring problem and the K-SAT problem. For these problems we develop equations for the probability that an infinite frozen cluster emerges. We then show that in the simplest approximation, this formalism reproduces the replica symmetric equations in a surprisingly straightforward manner.

Frozen order is a unifying concept in the analysis of glasses and geometrically frustrated systems in physics [13] and also in NP-complete problems in computer science, such as coloring [25] and K-SAT [30]. Frozen long-range order is most easily understood at zero temperature. At zero temperature the paradigm geometry is to fix the variables on a surface of the system and then to test whether these frozen degrees of freedom cause the propagation of frozen order into the bulk of a sample. A spin is frozen only if the spin is fixed or constrained by the spin configurations of its neighbors, as we shall demonstrate explicitly below using the Viana-Bray model. Frozen order may occur even though the variables (e.g. the spins in a spin glass) at each vertex of a graph look random. Furthermore, not all of the variables in the system need to be frozen. However for the system to be in the frozen ordered ground state, the frozen component must percolate.

As discussed in Chapter 1 the vertex q-coloring problem is equivalent to finding the ground state of the q-state Potts anti-ferromagnet [114]. Each node of a complex graph may have any one of q colors. The objective is to find the color configuration which minimizes the number of edges which have the same color at each end. The propagation of frozen color has many conceptual similarities with the propagation of rigidity in central force networks [61,92]. However there is a key difference, which makes the coloring problem NP-complete whereas the rigidity problem is polynomial. The key difference is that the constraints in coloring are distinguishable while the constraints in rigidity percolation are not.

Spin glasses and many frustrated antiferromagnets map exactly to problems in the NP-complete class [83]. NP-complete problems are of central interest in computer science (CSE) [46] and have motivated many attempts to design quantum algorithms for their efficient solution. The phase transitions which physicists study often correspond to a change in the computational complexity of the corresponding CSE problem. Since these problems are of enormous interest in physics, CSE and also in practical applications it is not surprising that there is a burgeoning of efforts to better understand the phase transition which occurs in NP-complete problems.

The physics community has applied the replica method to NP combinatorial problems with remarkable success [45,56,63,89,90,95,108]. In addition new algorithms have been developed based on a combination of replica symmetry breaking ideas from physics and belief propagation ideas from the artificial intelligence community [18,85,86]. Though the replica method is an excellent tool, it is quite difficult both technically and intuitively. In this chapter we show that a simple combinatorial procedure based on percolation ideas can reproduce many of the successes of the replica method. The percolation process occurring at the phase transition can be thought of as either percolation of constraint or percolation of frozen order. We derive the replica symmetric theories for K-SAT, the Viana-Bray model and coloring using percolation concepts. Elsewhere we show how these procedures lead to new algorithms for hard problems.

In the next section of we give a brief review of the analysis of connectivity percolation on Bethe lattices and random graphs, and also describe its extension to k-connectivity percolation. Also we describe the analysis of the glass transition, at $T = 0$, in the Viana-Bray model. Then we focus on the coloring problem, and in the last section we present an analysis of K-SAT.

4.2 Connectivity and Rigidity percolation

Percolation on diluted Bethe lattices was analyzed by Fisher and Essam [41], who defined the probability that a node is part of the infinite cluster, T. They found that the probability that a node is not on the infinite cluster, $Q = 1 - T$, only requires that all of its connected neighbors also not be part of the infinite cluster, so that

$$Q = (1 - p(1 - Q))^{\alpha} \tag{4.1}$$

where p is the probability that an edge is present in the Bethe lattice, and $\alpha = z - 1$, where z is the co-ordination number of the Bethe lattice. Note that this expression may be written as,

$$T = \sum_{l=1}^{\alpha} \binom{\alpha}{l} (pT)^{l} (1 - pT)^{\alpha - l} \tag{4.2}$$

which is more convenient for the generalization to rigidity percolation. From Eq.(4.2), it is easy to show that there is a phase transition at $p_c = \frac{1}{\alpha}$ and that $T \sim (p - p_c)$ near the critical threshold. The phase transition is thus continuous with order parameter exponent one. Somewhat earlier, this transition was also studied in the graph theory community by Erdös and Rényi [33]. They concentrated on random graphs, which consist of highly diluted complete graphs. A complete graph is a graph where every node is connected to every other node. In fact they defined $p = c/N$, where c is finite and showed that a giant (extensive) connected cluster emerges at $c = 1$. They derived an equation for the probability that a node is on the giant cluster, γ. Their equation is found from Eq.(4.2), by taking the limit $p = c/N$, $N = z \to \infty$, to find $\gamma = 1 - e^{-c\gamma}$. Near the critical point $\gamma \sim 2(c-1)/c^2$ so, as expected based on the universality hypothesis, γ also has an order parameter exponent of one. Rigidity percolation on Bethe lattices is described by a simple generalization of Eq.(4.2). In this generalization, each node has g degrees of freedom. For example if we wish to model rigidity percolation on central force networks, then $g = d$, where d is the lattice dimension. In order to make a giant g-rigid cluster, we need to constrain the g degrees of freedom at each node with at least g bonds, so we generalize Eq.(4.2) to:

$$T_g = \sum_{l=g}^{\alpha} \binom{\alpha}{l} (pT_g)^l (1 - pT_g)^{\alpha - l} \tag{4.3}$$

which is the simple generalization of Eq.(4.2) to the requirement of at least $g-1$ neighbor connections.

Eq.(4.3) was first discovered in the context of a Bethe lattice theory for Bootstrap percolation [20] and has been used more recently to develop a Bethe lattice theory for rigidity percolation [31,91,94]. In the random graph limit, Eq.(4.3) reduces to,

$$\gamma_g = 1 - e^{-c\gamma_g} \sum_{l=0}^{g-1} \frac{(c\gamma_g)^l}{l!} \tag{4.4}$$

When $g = 1$ this gives the Erdös and Rényi result [33] for the emergence of a giant cluster in random graphs, while for $g > 1$, there is a discontinuous onset of a finite solution at a sharp threshold c_g [94]. Numerical solution of Eq.(4.4) indicates that for $g = 2$, $c_2 = 3.3510(1)$. This value has also been found in a recent mathematical analysis [99] of the threshold for the emergence of the giant 3-core on random graphs. The k-core problem is equivalent to the k-bootstrap percolation problem. However the $k+1$-core is in general different than the k-rigidity problem, and even on Bethe lattices and random graphs there are some important differences.The most important difference is that for g-rigidity, the finite solution T_g is metastable for a range of $c > c_g$ [31,94]. The true rigidity transition actually sets in at $c_r > c_g$ and is identified using constraint counting arguments [31,91]. Nevertheless the probability of being on the infinite rigid cluster is correctly found from Eq.(4.4), provided $c > c_r$, where c_r is the rigidity threshold [31,91].

As we shall see below the analogous theories for glassy combinatorial problems, in particular the Viana-Bray model, $K - SAT$ and q-coloring, provide solutions at the level of the replica symmetric theory. Moreover, the methodology we introduce here can be used to develop simple and accurate recursive algorithms for these glassy problems on general graphs [40]. In the case of first order transitions, as occurs in q-coloring (with $q \geq 3$) and for K-SAT ($K \geq 3$), the transition point we find below marks the onset of metastability. In order to find the true threshold we need the ground state energy.

4.3 Viana-Bray model

We first analyze the onset of frozen order in the Viana-Bray(VB) spin-glass model [110], which provides a basic model for disordered and frustrated magnets, such as $Eu_xSr_{1-x}S$ [77]. The Hamiltonian for the VB model is,

$$H = \sum_{ij} J_{ij} S_i S_j \tag{4.5}$$

where $S_i = \pm 1$. The exchange constants J_{ij} are randomly drawn from the distribution,

$$D_p(J_{ij}) = p[\frac{1}{2}\delta(J_{ij} + J) + \frac{1}{2}\delta(J_{ij} - J)] + (1 - p)\delta(J_{ij}) \tag{4.6}$$

We focus on the random graph limit $p = c/N$ and we introduce the following probabilities:

P = probability a site is frozen in the up state

M = probability a site is frozen in the down state

D = probability a site is degenerate

In the absence of an applied field and within a symmetric assumption, $P = M$ and $D = 1 - 2M$. We then need to consider only one of these probabilities. However for clarity and for ease of generalization, we continue to include M and P separately. In terms of these order parameters, the magnetization is given by $m = P$-M and the spin glass order parameter is $q = P + M$. The recurrence formula for P, using $p = c/N$ is,

$$P = \sum_{k=0}^{\alpha} \sum_{l=k+1}^{\alpha} \frac{\alpha!}{k!l!(\alpha - k - l)!}$$

$$(\frac{cP}{2N} + \frac{cM}{2N})^k (\frac{cM}{2N} + \frac{cP}{2N})^l (1 - \frac{c}{N}(M + P))^{\alpha-k-l} \tag{4.7}$$

This is understood as follows. If a bond connects a site at the lower level to a site at the upper level then the site at the upper level will be frozen up: if the connecting bond is ferromagnetic and the lower level spin is frozen up; *or* if the connecting bond is anti-ferromagnetic and the lower level spin is frozen down. This event has probability $cP/2N + cM/2N$. Similarly, the probability that a spin at the upper level of the bond will be frozen down (negative) is given by, $cM/2N + cP/2N$. The newly added spin at the upper level is frozen up if there are a larger number of connections from the upper to the lower level which prefer the frozen up state. The sum in Eq.(4.7) is thus restricted to events of this sort. The event $(1 - c(P + M)/N)$ is the probability that a site at the lower level in the tree is either degenerate or disconnected from the newly added site. In the large N limit, Eq.(4.7) reduces to:

$$q = 2e^{-cq} \sum_{k=0}^{\infty} \sum_{l=k+1}^{\infty} \frac{(\frac{cq}{2})^{k+l}}{k!l!} = 1 - e^{-cq} I_0(cq) \tag{4.8}$$

where we have used the fact that we are considering a case where the magnetization $m = 0$. In that case, $M = P = q/2$, where q is the spin glass order parameter. I_0 is the spherical Bessel function of zeroth order. The result given by Eq.(4.8) has been found before within the replica symmetric solution to the Viana-Bray (VB) model [63]. Thus symmetric constraint percolation (CP) in the VB model is equivalent to the ground state spin glass transition as found within the replica symmetric approach. The CP approach is attractive because is it is simple, it avoids the mathematical difficulties of the replica method and it is physically transparent. The construction we have used makes it clear that simple connectivity is sufficient to ensure propagation of spin glass order in the VB model. Constraint percolation occurs at $c = 1$ and the order parameter approaches zero as $q \sim \frac{4}{3c^2}(c-1)$, so the CP transition in this case is continuous, with the same exponent as the Erdös-Rényi transition.

4.4 K-SAT

As discussed in previous chapters, the satisfiability problems we consider ask the following question: given a set of binary variables, $z_i = 0, 1$ or equivalently $z_i = True\ orFalse$, is it possible to satisfy a specified set of constraints on these variables? In the $K - SAT$ case, a typical constraint is of the form,

$$(z_i \wedge \bar{z}_j \wedge z_k) \tag{4.9}$$

where $\wedge$ is the logical AND operation and the overline indicates a negated variable. This logical clause is satisfied (SAT) if any one of the variables in the clause is SAT. The variables z_i and z_k are SAT if they are true (T), which we take to be $z_i = z_k = 1$, while the variable $\bar{z}_j$ is SAT when z_j is false (F), which corresponds to $z_j = 0$. We shall also fix the number of variables in each clause to be

K, which is the K-SAT problem. In these SAT problems we shall randomly choose a set of M clauses and try to find the assignment of the binary variables which minimizes the number of violated clauses. Each variable appearing in a clause is negated with probability $1/2$ and the number of variables is N. The key ratio is $\alpha = M/N$. We would like to find the threshold for constraint percolation. That is, what is the threshold for the appearance of a giant cluster of clauses where each clause is completely specified or *frozen*. These completely specified clauses cannot be altered without increasing the total number of violated clauses, so that they are non-degenerate. There are three types of clauses in an optimal assignment of a formula:

(i) Clauses that are SAT but are degenerate;

(ii) Clauses that are SAT but are frozen;

(iii) Clauses that are UNSAT but are degenerate.

Only type (ii) clauses propagate constraint.

We make a tree construction of the factor graph for the $K - SATproblem$ as it is shown in Fig.(4.1). The probability that a *variable* is frozen and part of the giant frozen cluster is V and the probability that a *clause* is frozen and part of the giant frozen cluster is F. The branching of the variable nodes has maximum coordination M, but the probability that a link actually exists between a node and clause is $p = K/N$. We start by assuming that a variable is frozen at level 1 (as it is shown in Fig.4.1) and then determine the consequences of this assumption at levels 2 and 3.

The probability F that level 2 clause is frozen, given the probability V, that a variable is frozen at level 1 is given by:

$$F = (\frac{V}{2})^{K-1}. \tag{4.10}$$

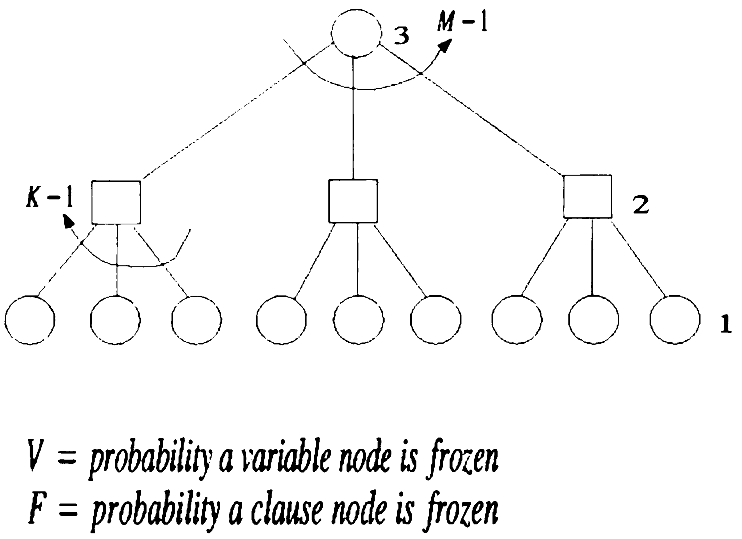

3
M - 1
K - 1
2
1
V = probability a variable node is frozen
F = probability a clause node is frozen

Figure 4.1: The factor graph used to construct the recurrence relations. The circles denote variable nodes, while the square nodes are the clause nodes. V is the probability that a variable node is frozen, while F is the probability that a clause node is frozen (see the text). We assume that a variable at level 1 is frozen and find the probability that a variable at level 3 is frozen. The clause nodes have co-ordination K, while the variable nodes have co-ordination M

This equation is understood as follows. In order for a clause at level 2 to be frozen by the variables at level 1, all of the level 1 variables to which it is connected must be frozen and in conflict with the assignment in the clause. This imposes a fixed assignment on the variable 3. This is the only configuration of variables at level 1 which propagates constraint through a clause to level 3. Now we must consider the cumulative effect of all of the clauses which are connected to the variable at level 3. There are $M - 1$ such clauses of which a fraction F propagate constraint (are frozen) according to the mechanism of the previous paragraph. Some of these frozen clauses propagate the requirement x and others propagate the requirement $\overline{x}$. The variable at level 3 then has three possible states: P = positive, N = negative and D=degenerate. The state of the level 3 variable is degenerate if the number of constrained connections which favor the positive state (x) is the same as the number of connections which favor the negative state ($\overline{x}$). The probability this variable is frozen (*i.e.* either negated or not) is $V = P + N = 1 - D$ as we are considering the case where the probability that a variable is negated is $1/2$. It is straightforward to generalize to the case of unequal probabilities. The probability that the node at level 3 is degenerate is then:

$$D = \sum_{k=0}^{M} \frac{M!}{(k!)^2(M-2k)!}(\frac{pF}{2})^{2k}(1-pF)^{M-2k} \tag{4.11}$$

Where we have used the fact that the probability that a connection occurs between a variable node and a clause node is $p = K/N$. Eq.(4.11) is understood as follows. The probability that a clause at level 2 is frozen and connected(*i.e.* it propagates constraint), and it requires the variable at level 3 to be x is $pF/2$. The probability that a clause propagates constraint and it requires the variable at level 3 to be $\overline{x}$ is also $pF/2$. The variable at level 3 is degenerate if these two events occur an equal number of times, hence the term $(pF/2)^{2k}$. The combinatorial factor gives all ways of arranging these events, taking into account that the x and $\overline{x}$ events are distinct. In the thermodynamic limit, using $pM = \alpha K$, we find,

$$D = e^{-\alpha KF} \sum_{k=0}^{\infty} \frac{1}{(k!)^2}(\frac{\alpha KF}{2})^{2k} \tag{4.12}$$

Then we can write:

$$V = 1 - D = 1 - e^{-\alpha KF} I_0(\alpha KF) \tag{4.13}$$

where I_0 is the spherical Bessel function of zero order. Note that $I_0(0) = 1$. For completeness, we note that the probability that the new variable is frozen in the positive (not negated) state is:

$$P = \sum_{k=0}^{M} \sum_{l=k+1}^{M} \frac{M!}{(k!)^2(M-k-l)!}(\frac{pF}{2})^{k+l}(1-pF)^{M-k-l} \tag{4.14}$$

The probability that the variable is frozen in the N state is the same as P for the case

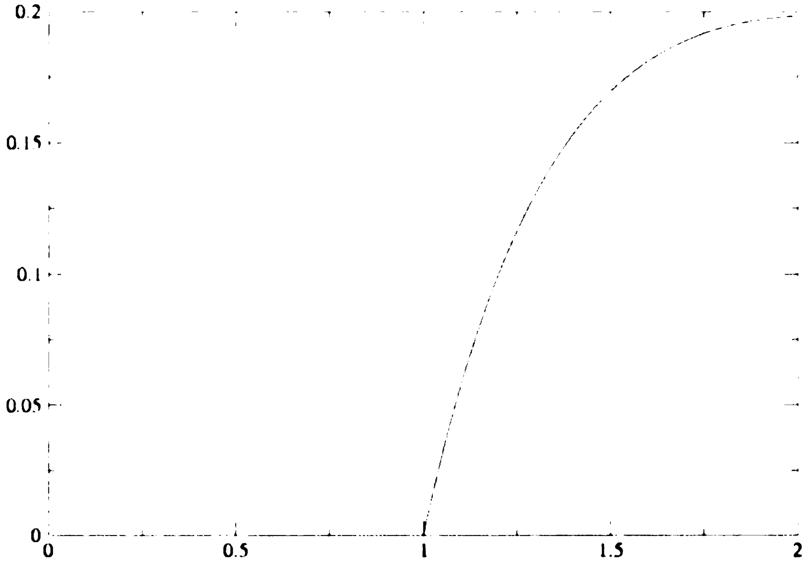


Figure 4.2: The probability that a clause is frozen, F, as a function of α, for 2-SAT.

we are considering, where the variables have equal probability of being negated and not negated.

Equations (4.14) and (4.10) provide the self consistent theory for the onset of a giant constrained cluster in K-SAT. We now analyze this theory for the two typical cases.

The 2-SAT case ($K = 2$) In this case Eq.(4.10) is $F = V/2$. Expanding Eq.(4.13) in powers of F, we then have,

$$F = \frac{1}{2}[1 - (1 - 2\alpha F + 2\alpha^2 F^2 + ..)(1 + \alpha^2 F^2))] \tag{4.15}$$

This has the trivial solution $\alpha = 1$. It also has the non-trivial solution

$$F \approx \frac{2}{3\alpha^2}(\alpha - 1) \quad with \;\; (\alpha - 1) \ll 1 \tag{4.16}$$

Thus the random $2 - SAT$ giant cluster emerges smoothly at $\alpha = 1$. Numerical calculation of F from equations (4.10) and (4.13) is presented in Fig.4.2.

The $K \geq 3$-SAT case. In this case, Eq.(4.14) and Eq.(4.10) do not have a solution

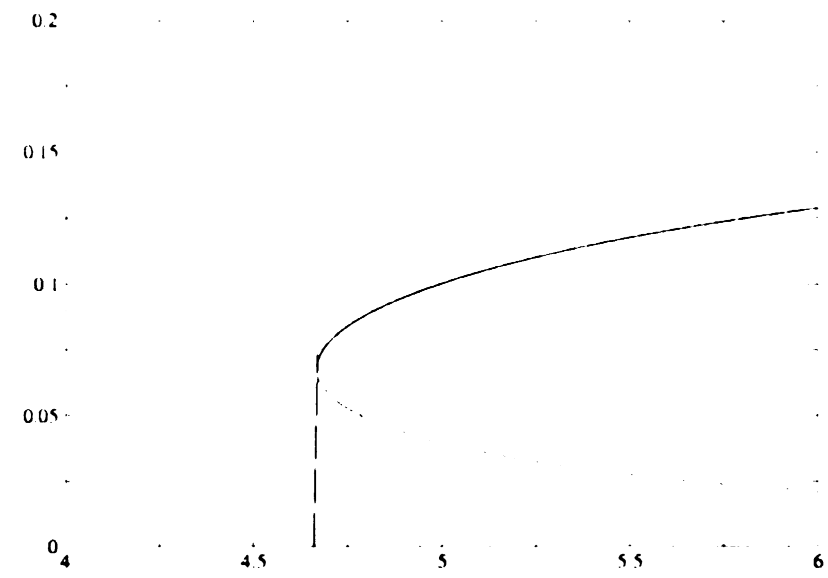


Figure 4.3: The probability that a clause is frozen, F, as a function of α, for 3-SAT.

with a smooth behavior near the critical point. However they do have a non-trivial solution which has a discontinuous onset at a threshold value, $\alpha_c(K)$. This solution is found by iteration of Eq.(4.10) and Eq.(4.13) and the result is presented in Fig.(4.3). We find that although the emergence of the giant cluster is discontinuous, for any $K > 2$ the size of the first order jump decreases quite rapidly with increasing K. This indicates that the K-SAT transition is weakly first order and that an analytic analysis at large K is possible. The 3-SAT critical value which we find, $\alpha_c(3) \approx 4.6673(3)$, is consistent with the replica symmetric solution [89] for the metastability point, and significantly higher than the numerical values for the K-SAT transition which lie around 4.3 [85]. The numerical results we have found (using Eq.(4.14)) for the metastable point and the jump in F at that point: $\alpha_c(3) = 4.6673(3)$, $\delta F_c = 0.0680(1)$; $\alpha_c(4) = 11.833(1)$, $\delta F_c = 0.0341(3)$; $\alpha_c(5) = 29.91(1)$, $\delta F_c = 0.016(1)$; $\alpha_c(6) = 64.1(1)$, $\delta F_c = 0.0071(1)$. The lower branches (the unstable solutions for equations (4.10) and (4.13)) were found using the half-search interval method.

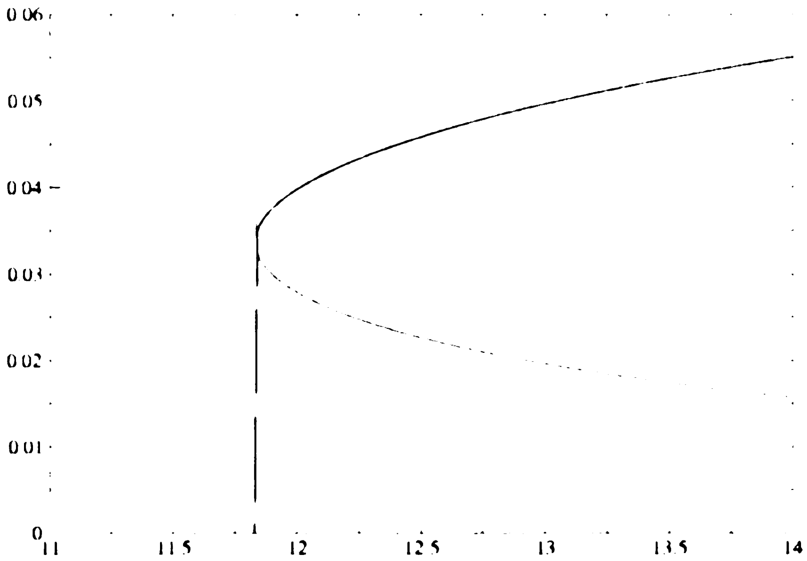


Figure 4.4: The probability that a clause is frozen, F, as a function of α, for 4-SAT.

4.5 Energy per variable

For the $K-SAT$ problem the energy gain of removing a variable node and all clauses connected to it reads:

$$\epsilon_v = E(N,M) - E(N-1, M-pM) = \frac{\partial E}{\partial N} + K\alpha\frac{\partial E}{\partial M} \tag{4.17}$$

The energy gain of removing a clause is:

$$\epsilon_c = E(N,M) - E(N,M-1) = \frac{\partial E}{\partial M} \tag{4.18}$$

Also we can write:

$$\frac{\partial E}{\partial N} = \frac{\partial (N\epsilon)}{\partial N} = \epsilon + N\frac{\partial \epsilon}{\partial \alpha}\frac{\partial \alpha}{\partial N} = \epsilon - \alpha\frac{\partial \epsilon}{\partial \alpha}$$

and

$$\frac{\partial E}{\partial M} = \frac{\partial (N\epsilon)}{\partial M} = N\frac{\partial \epsilon}{\partial \alpha}\frac{\partial \alpha}{\partial M} = \frac{\partial \epsilon}{\partial \alpha}$$

In this way, Eq.(4.17) and Eq.(4.18) become:

$$\epsilon_v = \epsilon - \alpha\frac{\partial\epsilon}{\partial\alpha} + K\alpha\frac{\partial\epsilon}{\partial\alpha} \tag{4.19}$$

and

$$\epsilon_c = E(N,M) - E(N,M-1) = \frac{\partial E}{\partial M} = \frac{\partial\epsilon}{\partial\alpha} \tag{4.20}$$

Plugging Eq.(4.20) into Eq.(4.19) we obtain the *true* energy per variable for the $K-SAT$ problem

$$\epsilon = \epsilon_v - (K-1)\alpha\epsilon_c \tag{4.21}$$

For the $3-SAT$ problem Eq.(4.21) is the same as the one obtained by Mezard [86] using the cavity approach. The energy density for removing a variable node reads:

$$\begin{gathered}\epsilon_v = \sum_{K=0}^{\infty} e^{-x}\frac{K}{(K!)^2}\left(\frac{x}{2}\right)^{2K} + \\ 2\sum_{K=0}^{\infty}\sum_{l=K+1}^{\infty} K\frac{1}{K!l!}\left(\frac{x}{2}\right)^{K+l} e^{-x} = S_1 + 2S_2\end{gathered} \tag{4.22}$$

with S_1 and S_2 given by:

$$S_1 = \sum_{K=0}^{\infty} e^{-x}\frac{K}{(K!)^2}\left(\frac{x}{2}\right)^{2K} \tag{4.23}$$

$$\begin{aligned}S_2 &= \sum_{K=0}^{\infty}\sum_{l=K+1}^{\infty} K\frac{1}{K!l!}\left(\frac{x}{2}\right)^{K+l} e^{-x} \\ &= \sum_{K=0}^{\infty}\sum_{n=1}^{\infty}\frac{K}{K!(K+n)!}\left(\frac{x}{2}\right)^{2K+n}\end{aligned} \tag{4.24}$$

In our notation $x = \alpha KF$ with F given by Eq.(4.10). Using the identities:

$$\sum_{K=0}^{\infty} \frac{K}{K!(K+n)!}\left(\frac{x}{2}\right)^{2K+n} = \frac{x}{2} I_{n+1}(x)$$

and

$$e^x = I_0(x) + 2\sum_{n=1}^{\infty} I_n(x)$$

we have for S_1 and S_2:

$$S_1 = \frac{x}{2} e^{-x} I_1(x) \tag{4.25}$$

and

$$S_2 = \frac{x}{4}\Big(1 - e^{-x}(I_0(x) + 2I_1(x))\Big) \tag{4.26}$$

Using Eq. (4.25) Eq.(4.26) and Eq.(4.22), the final expression for energy density when we remove a variable node reads:

$$\epsilon_v = \frac{x}{2}\Big(1 - e^{-x}(I_0(x) + I_1(x))\Big) \tag{4.27}$$

The energy density when we remove a clause node is given by:

$$\epsilon_c = \left(\frac{V}{2}\right)^K = \frac{x}{\alpha K}\frac{1 - e^{-x} I_0(x)}{2} \tag{4.28}$$

where $V = 1 - e^{-\alpha KF} I_0(\alpha KF)$ as we already found (Eq.(4.13). In this way the true energy density (4.21) reads:

$$\epsilon = \epsilon_v - (K-1)\alpha\epsilon_c = \frac{x}{2K}\Big(1 - e^{-x}(I_0(x) + KI_1(x))\Big) \tag{4.29}$$

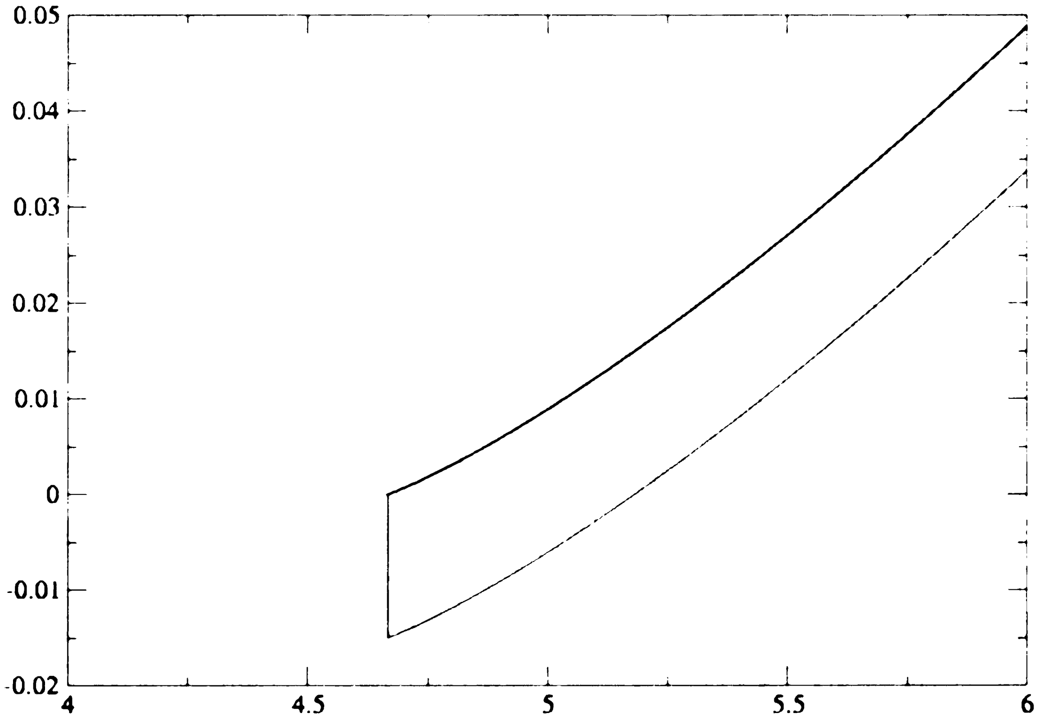


Figure 4.5: Ground state energy for $K = 3$ using Eq.4.30 (upper curve) and using the *cavity approach* (lower curve line) as a function of α

For the $K = 3$ case, the energy given by Eq.(4.29) is represented in Fig.(4.5). In the same figure for the $K = 3$ case is represented the energy given by equation:

$$\epsilon = \int_0^{\alpha} F^{\frac{K}{K-1}} \, d\alpha \tag{4.30}$$

From Fig.(4.5) we see that the metastable solution for $E_{GS} = 0$ is reached at $\alpha_{metastable} \approx 4.667$ while the stable solution is reached at $\alpha_{stable} \approx 5.181$.

Fig.(4.6) is for the $K = 4$ case, and we see that the ground state energy is reached at $\alpha_{metastable} \approx 11.832$ and $\alpha_{stable} \approx 14.369$.

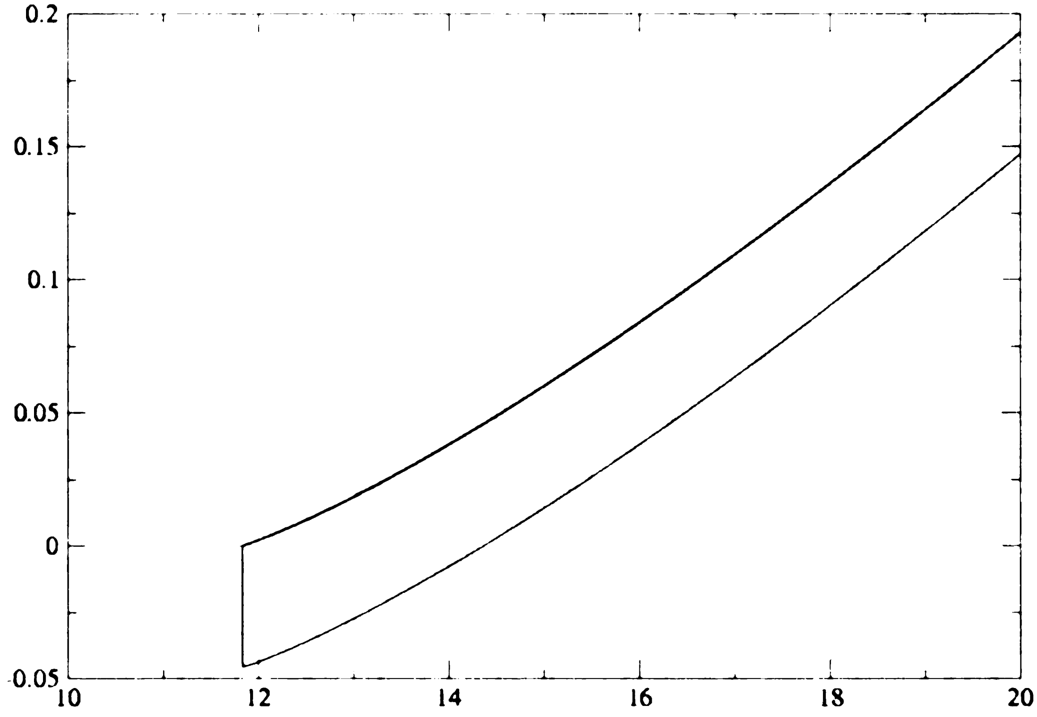


Figure 4.6: Ground state energy for $K = 4$ using Eq.4.30 (upper curve) and using **the** *cavity approach* (lower curve) as a function of $\alpha = M/N$.

4.6 Coloring

The Graph Coloring problem problem is simply stated but is very difficult to solve either analytically or numerically. Given a graph, or a lattice, and given a number q of available colors, the problem consists in finding a coloring of vertices such that no edge has its two end vertices of the same color. The possibility of finding such a solution depends on the way the graph is constructed and also on the number of colors. The minimally number of colors is the *chromatic number* of the graph. For large random graphs, there exists a critical average connectivity beyond which the graphs become uncolorable with probability going to one as the graph size (the number of vertices) goes to infinity.

As discussed in Chapter 3 graphs generated close to their critical connectivity are hard to color and therefore the study of critical instances is an algorithmic challenge for understanding the onset of computational complexity. For many NP problems, the complexity shows an easy-hard-easy pattern [25] [21] [70], [109].

Recently, methods from statistical physics have been applied to computational complexity problems one of them being the Coloring problem. The statistical physics approach is based on the introduction of a cost function (or energy), calculate the free energy in the large system limit (thermodynamic limit) and from here calculate the macroscopic quantities of interest. From the free energy we can calculate the ground state energy which allows us to make predictions of the graph colorability: a non-zero ground state energy indicates that random graphs are typically not colorable. Apart from determining the colorability of the graph, from the ground state energy we can determine the typical minimal fraction of unsatisfied edges when the graph is not colorable. Also, the ground state entropy gives us information about the different coloring schemes that share the minimum number of unsatisfied edges (*i.e.* the degeneracy).

As shown in Chapter 1, the q-coloring problem can be mapped to a Potts model:

$$E = \sum_{i,j} b_{ij} \delta_{x_i x_j} \tag{4.31}$$

where $b_{ij} = 1$ (0) if an edge is present (absent) between sites i and j. This is the same as the energy of the Potts anti-ferromagnetic model which is an example of physical system with geometrical frustration [114]. Optimizing the color configuration of a graph is equivalent to finding the ground state of the Potts anti-ferromagnetic. A proper coloring of a graph corresponds to a zero energy ground state configuration of a Potts anti-ferromagnetic with q-state variables.

In the next section we discuss the order parameter we use to describe frozen order, over-constrained variables and colorable variables. This is followed by a discussion of the coloring problem on complete graphs where the problem is trivially solvable. The next two sections contain the analysis of the equations for percolation of frozen order on diluted Bethe lattices, with particular emphasis on the large coordination number limit. Then we present some results for the special case $q = 2$ which exhibits a threshold behavior similar to percolation, and also the more interesting case $q = 3$. The last section contains a brief summary and some concluding remarks.

4.7 The model and Limiting results

As described above, each site has a variable which may have one of q colors, that is $x_i = 1, \ldots, q$. If a site has no neighbors, then the energy is the same for any of these q possible colors. However if the site has many neighbors we need to find the color configuration which minimizes the number of mono-color bonds, *i.e.* bonds which have the same color at each end. Of particular interest is the possibility that a site

fixed

whos

G

H

A sit

ored

the r

froz

bor

F =

fro

lat

reg

ot

ex

ov

co

U

t

a site has a unique color which minimizes the energy. That is, the color of a site is fixed by its neighbor configuration. We introduce two probabilities related to sites whose color is fixed, *i.e.* , *frozen*, namely:

$G =$ Probability that a site is frozen and colorable,

$H =$ Probability that a site is frozen and not colorable.

A site is frozen and colorable if the site has a unique color, but the site can be colored without causing an increase in the energy cost of the system. That is, none of the neighbors of the site has the same color as the frozen site. In contrast, a site is frozen and uncolorable if its color is unique, but it has at least one connected neighbor which has the same color in the ground state. The sum of these probabilities, $F = G + H$, plays the most important role in the analysis. In order to be a globally frozen ground state, frozen order must percolate. If frozen order does not percolate, then a surface exists which separate one region of frozen order from another region of frozen order. These two regions may be deformed with respect to each other and hence the ground state is unstable to a long wave length zero-energy excitation. We shall also find the probability, O, that a site is not colorable and over-constrained (this probability includes the probability H of being frozen and colorable). Finally, we will find the probability U that a site is under-constrained. Under-constrained sites are colorable but are not frozen. The following conservation law holds for these probabilities:

$$O + G + U = 1 \tag{4.32}$$

The energy is found from the order parameter by noticing that if we add one edge to the system, we have probability $F^2/2$ of making a monocolor bond, so the energy of the system in increased by one unit. That is, if both of the sites at the ends of the edge have frozen colors and their colors are the same, then the edge is frus-

trat
wh

wl
di
ar

In

v
i
e
v
t

trated and costs unit energy. The probability that the two sites are frozen is F^2, while the probability that the two sites have the same color is $1/q$. We thus have:

$$E(B+1) - E(B) \approx \frac{\partial E}{\partial B} = \frac{F^2}{q} \tag{4.33}$$

where B is the number of bonds in the system before the edge is added. In a diluted complete graph, $B = pN(N-1)/2$. In the random graph limit, $p = c/N$ and $B = cN/2$. Defining the energy density $\epsilon = E/N$, we find:

$$\frac{\partial \epsilon}{\partial c} = \frac{F^2}{2q} dc \tag{4.34}$$

Integrating this expression we have,

$$\epsilon(c) = \int_{c^*}^{c} \frac{F^2}{2q} \tag{4.35}$$

where c_* is the coloring threshold. A simple example which can be solved explicitly is a complete graph, where every site is connected to every other site (this is equivalent to $c \to N/2$ in the above). We define n_l to be the number of vertices which are assigned color l. Since every vertex is connected to every other vertex, the energy is simply:

$$E(n_1, n_2, \ldots, n_q) = \frac{1}{2} \sum_{l=1}^{q} n_l(n_l - 1) \delta(\sum_{l=1}^{q} n_l - N) \tag{4.36}$$

If we define the fractions $x_l = n_l/N$ and take the limit $N \to \infty$, we find,

$$E(x_1, x_2, \ldots, x_q) = \frac{N^2}{2} \sum_{l=1}^{q} x_l^2 \delta(\sum_{l=1}^{q} x_l - 1) \tag{4.37}$$

This energy is a convex function of the fractions x_l, so its minimum is either at a boundary of the domain, or it is a unique minimum in the interior of the domain.

A straightforward analysis demonstrates that the minimum is a symmetric solution at $x_l = 1/q$ and hence the energy is $E = N^2/2q$. This state has degeneracy $N!/((N/q)!)^q$ which is all the ways of arranging q sets of n/q items each amongst the total of N items. In order to make contact with the energy that we defined in (4.35), we set $c = N$, and we find,

$$\epsilon(c \to \infty) = \frac{c}{2q} \tag{4.38}$$

4.8 Coloring on Bethe Lattice

A Bethe lattice introduced by Hans Bethe in 1935 is a connected cycle-free graph where each node is connected to z neighbors with z called the coordination number. Bethe lattice can be seen as a tree-like structure growing from a central node called root, with all the nodes arranged in shells around the central one.

We solve the problem of coloring random lattices in the limit of large coordination number. In this limit the Bethe lattice solution approaches that of the solution of random graphs. It is important to point out at the outset that trees are bipartite, so that coloring on simple trees is trivial. We only need two colors to color a bipartite graph, so for any $q > 2$ there is no transition on trees with free boundaries. However there is a symmetric fixed point (*i.e.* all colors are equally probable) even on trees. This symmetric solution is produced either by choosing equal probability boundary conditions on the outer leaves on the trees or by explicitly equating the probabilities of all of the available colors.

Consider a Bethe lattice of coordination number z. The limiting behavior of such lattices is calculated on one branch of the tree which has coordination number $\alpha = N - 1$. We consider randomly removing bonds on the Bethe lattice so that a bond is present with probability p. In the random graph limit $N \to \infty$ and

$p \longrightarrow c/N$ where c is finite, there is typically a finite number of connections between neighboring sites. We solve the problem of finding the lowest cost coloring of diluted Bethe lattices. As is usual on trees, this calculation can be done recursively due to the independence of the probabilities on a given level of the tree. In this problem we have q colors and we seek to assign colors to the vertices of the graph so that the minimum number of connected neighbor sites have the same color.

Our analysis centers on the probability F_l $(l = 1, 2, ..q)$, which is the probability that a site is frozen in color l. We also introduce the probability G_l, which is the probability that a site is frozen with color l and is colorable. The probability G_1 is given by the recursion relation:

$$G_1 = \sum_{k_2=1}^{\infty} \sum_{k_3=1}^{\infty} .. \sum_{k_q=1}^{\infty} \sum_{k_{q+1}=0}^{\infty} \frac{\alpha!}{k_2!k_3!...k_q!k_{q+1}!}$$

$$(pF_2)^{k_2}(pF_3)^{k_3}...(pF_q)^{k_q}(1 - p\sum F_l)^{k_{q+1}}\delta(s + \sum_{l=2}^{q+1} k_l - \alpha) \qquad (4.39)$$

This formula is understood as follows. In order for a site to be frozen in the color 1, all the other $q - 1$ colors must appear and be frozen on one of the connected neighbor sites. The probability that a neighbor site is connected and frozen in color l is pF_l. This event may occur one or more times, so we must sum over $k_l = 1, ...\infty$. We thus have a term $(pF_l)^{k_l}$ for each of the required $q - 1$ frozen neighbor colors. However, we must also allow for the possibility of events which are not of the type pF_l, which leads to the term $(1 - p\sum F_l)^{k_q+1}$. This probability is summed from 0 to ∞ as it does not have to exist in a configuration in order to ensure that G_1 be finite. Note however that $(1 - p\sum F_l)$ is by far the most likely event in the random graph limit, where $p \longrightarrow c/N$. All of these probabilities are exclusive and independent. We must also allow for all ways of arranging this set of $q + 1$ exclusive events amongst the α possible connections between our newly added site and the sites at the lower level in the tree. This leads to the multinomial factor. Eq. (4.39) occurs for each of the q colors that are allowed. Now we make the symmetric assumption in which each color is assumed to occur on frozen sites with equal probability. We thus can write $G_i = G/q$ and $F_i = F/q$. Equation 4.39 reduces to:

$$G = q \sum_{k_2=1}^{\infty} \sum_{k_3=1}^{\infty} .. \sum_{k_q=1}^{\infty} \sum_{k_{q+1}=0}^{\infty} \frac{\alpha!}{k_2!k_3!...k_q!k_{q+1}!}$$

$$(pF/q)^{\sum_{l=2}^{q} k_l}(1 - pF)^{k_{q+1}}\delta(s + \sum_{l=2}^{q+1} k_l - \alpha) \tag{4.40}$$

Using a convenient form for the δ function we have:

$$\begin{aligned} G = q\frac{\alpha!}{2\pi i} \int \frac{dz}{z^{\alpha+1}} \left(\sum_{k=1}^{\infty} \frac{1}{k!} (\frac{pFz}{q})^k \right)^{q-1} \sum_{k=0}^{\infty} \frac{1}{k!} [(1 - pF)z]^k \\ = q\frac{\alpha!}{2\pi i} \int \frac{dz}{z^{\alpha+1}} (e^{\frac{pFz}{q}} - 1)^{q-1} e^{(1-pF)z} \\ = q \sum_{r=0}^{\infty} (-1)^{q-1-r} \binom{q-1}{r} \frac{\alpha!}{2\pi i} \int \frac{dz}{z^{\alpha+1}} \exp[(1 - pF)z + \frac{pFrz}{q}] \\ = q \sum_{r=0}^{\infty} (-1)^{q-1-r} \binom{q-1}{r} \left(1 - pF + \frac{pFr}{q}\right)^{\alpha} \end{aligned} \tag{4.41}$$

Now we take the random graph limit, $p = c/N$, $\alpha \to N$ to find:

$$\begin{aligned} G = s \sum_{r=0}^{q-1} (-1)^{q-r-1} \binom{q-1}{r} e^{-cf+\frac{cFr}{2}} \\ = qe_{-cF}[e^{cF/q} - 1]^{q-1} \end{aligned} \tag{4.42}$$

The calculation of H, the probability that a site is frozen but not colorable (*i.e.* it has neighbors which have the same color) is similar though more complex. For a site to be *uncolorable* but frozen, it may have a color which occurs on a neighboring frozen site. However for color 1 to be frozen and colorable, the number of times a frozen neighbor has color 1 must be strictly smaller then the number of times any of the other $q - 1$ colors occur on frozen neighbors. There is an infinite series of terms of this sort, consisting of cases where the color 1 has $1, 2, 3, ...$ neighbors of the same color while all the other colors occur a strictly larger number of times for each case. The probability H_1 is given by,

$$H_1 = \sum_{s=1}^{\alpha} \sum_{k_2=s+1}^{\infty} .. \sum_{k_q=s+1}^{\infty} \sum_{k_{q+1}=0}^{\infty} \frac{\alpha!}{s!k_2!k_3!...k_q!k_{q+1}!}$$

$$(pF_1)^s (pF_2)^{k_2} ... (pF_q)^{k_q} (1 - p\sum F_l)^{k_{q+1}} \delta(s + \sum_{l=2}^{q+1} k_l - \alpha) \tag{4.43}$$

Similar equations occur for H_i. The case $s = 0$ correspond to the probability G_1 of Eq.(4.39). Making the symmetric assumption $H_1 = H/q$, $F_l = F/q$, and using the integral form of the delta function we find,

$$H = q\frac{\alpha!}{2\pi i} \int \frac{dz}{z^{\alpha+1}} \sum_{s=1}^{\infty} \frac{1}{s!} (\frac{pFz}{q})^s$$

$$\left(\sum_{k=s+1}^{\infty} \frac{1}{k!} (\frac{pFz}{q})^k \right)^{q-1} \sum_{t=0}^{\infty} \frac{1}{t!} \Big((1 - pF)z \Big)^t \tag{4.44}$$

Using $F = G + H$ together with Eq. (4.42) and Eq.(4.44) we have the key equations for the symmetric theory of constraint percolation. The solutions of these self-consistent equations lead to predictions for the behavior of F from which we may calculate other quantities of interest. For example, the probability that a site is not colorable or over-constrained is given by,

$$O = \sum_{k=1}^{\infty} \sum_{k_2=1}^{\infty} .. \sum_{k_q=1}^{\infty} \sum_{k_{q+1}=0}^{\infty} \frac{\alpha!}{k_1!k_2!k_3!...k_q!k_{q+1}!}$$

$$(pF_1)^{k_1} (pF_2)^{k_2} ... (pF_q)^{k_q} (1 - p\sum F_l)^{k_{q+1}} \delta(\sum_{l=1}^{q+1} k_l - \alpha) \tag{4.45}$$

That is a site is over-constrained if all of the colors occur and are frozen on neighboring sites. Using the symmetric assumption and the integral form for the delta function we have,

$$O = \frac{\alpha!}{2\pi i} \int \frac{dz}{z^{\alpha+1}} \left(\sum_{k=1}^{\infty} \frac{1}{k!} (\frac{pFz}{q})^k \right)^{q-1} \sum_{t=0}^{\infty} \frac{1}{t!} [(1-pF)z]^t$$
$$= (1 - e^{-cF/q})^q \tag{4.46}$$

From this we find the simple the result:

$$O = (1 - e^{\frac{cF}{q}})^q \tag{4.47}$$

Finally, the probability that a site is under-constrained is given by the probability that there are $q-2$ or fewer connected neighbors which have frozen colors. For the symmetric case, this leads to,

$$U = e^{-cF} \sum_{s=0}^{q-2} C_s^q \left(\sum_{k=1}^{\infty} (\frac{cF}{q})^k \frac{1}{k!} \right)^s \tag{4.48}$$

which reduces to

$$U = 1 - (1 - e^{-cF/q})^q - qe^{-cF}(e^{cF/q} - 1)^{q-1} \tag{4.49}$$

Comparing Eqs. (4.42), (4.47) and (4.49), we have the expected normalization condition $G + O + U = 1$.

4.9 Results

For $q = 2$, we assume that F is continuous near the percolation threshold and expand this expression in powers of F which yields,

$$F \approx cF - \frac{3}{4}(cF)^2 + O((cF)^3) \tag{4.50}$$

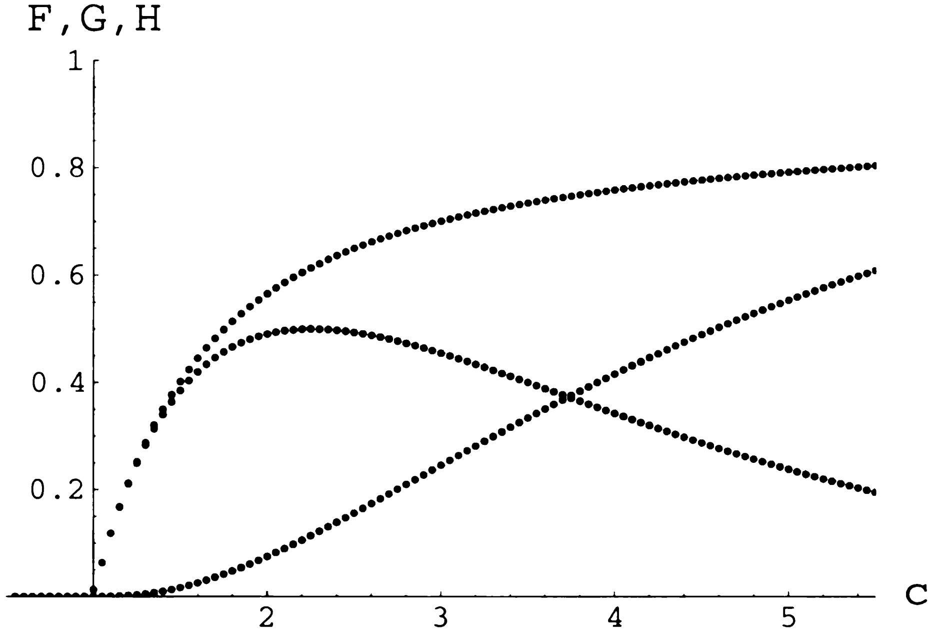


Figure 4.7: The coloring order parameters for $q = 2$. The lower two curves are the probability that a site is frozen and colorable, G (the $s = 0$ term in eq.(4.40)), and the probability that a site is frozen and frustrated, H (the $s \geq 1$ term in eq.(4.40)). The top curve is the probability that a site has a frozen color $F = G + H$, which is found by solving eq.(4.40) with $q = 2$.

This has the solution:

$$F \approx \frac{4}{3c^2}(c-1) \;\; c \geq 1 \tag{4.51}$$

So, other than a prefactor of $4/3c^2$ instead of $2/c^2$, the critical behavior of the giant cluster probability is the same as is in random graphs [33,41]. For c well away from the transition, we solve Eq.(4.50) numerically. The s and k sums converge rapidly and for the c range near critical threshold, only a few terms are required for high accuracy results.

From the solution for F we obtain all of the results of interest and they are presented in Fig.(4.7). The continuous behavior of 2-coloring near threshold is evident from these data. Using eq.(4.35), the energy density for $q = 3$ is represented in Fig.(4.9). For $q = 3$, an attempt to find a continuous transition by expanding in powers of F fails.

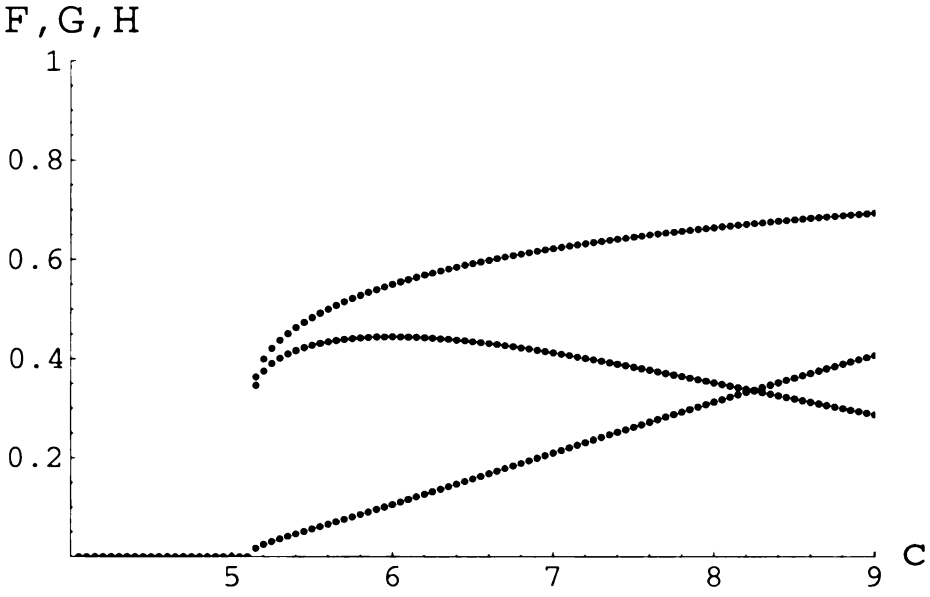


Figure 4.8: The coloring order parameters for $q = 3$. The lower two curves are the probability that a site is frozen and colorable, G (the $s = 0$ term in eq.(4.40)), and the probability that a site is frozen and frustrated, H (the $s \geq 1$ term in Eq.(4.40)). The top curve is the probability that a site has a frozen color $F = G + H$, which is found by solving Eq.(4.40) with $q = 3$.

Numerical solution of Eq.(4.40) is presented in Fig.(4.8) where it was seen that there is a jump discontinuity in the infinite frozen cluster probability at a sharp threshold. We find that $c_* = 5.14(1)$ and that the jump in the order parameter is $\Delta F_c = 0.365(1)$. We thus find that the coloring transition for $q = 3$ is first order as has been found in numerical simulations on random graphs [25].

Our coloring threshold is consistent with a recent replica symmetric numerical calculation, which yielded $c_* \approx 5.1$ [108], but is significantly higher than that found in the simulation work of Culberson and Gent [25] where $c_* \approx 4.5 - 4.7$ or in the numerical work on survey propagation [95], which yields $c_* \approx 4.42$. Nevertheless the nature of the transition is correctly captured by the simple CP theory. It is also important to note that the solution found here may also be metastable for a range of c, as was found in the rigidity case [31]. The onset of metastability is an important threshold from the point of algorithmic efficiency, as it marks the onset

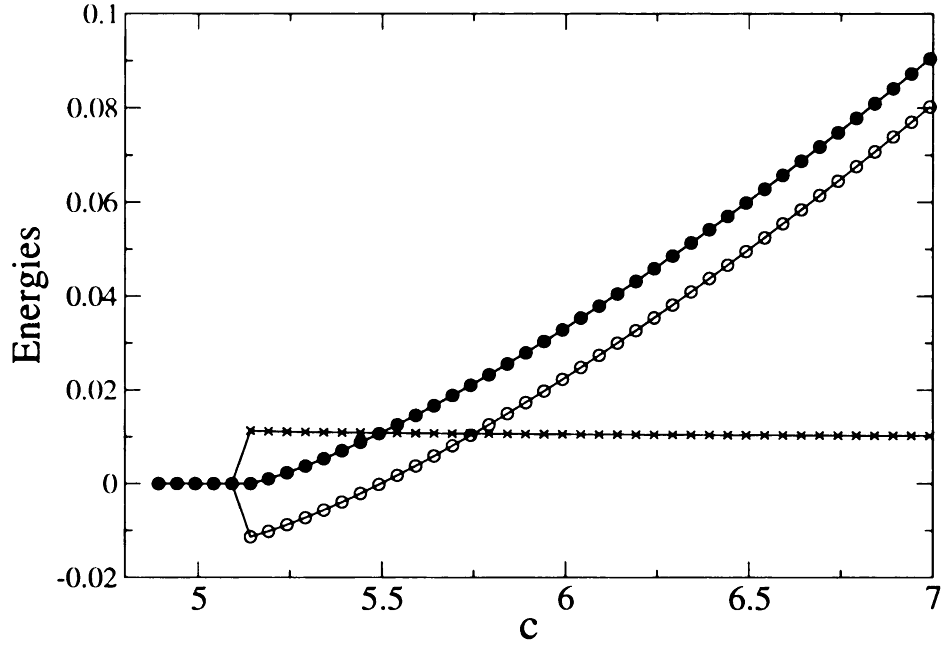


Figure 4.9: Energy density for the $q = 3$ case

of glassy relaxation dynamics.

The coloring theory developed above can be formulated in a very similar way to the formulation of the propagation of the k-core. However, there is a critical difference. The constraints in the coloring theory have to be treated as distinguishable, while the constraints in the k-core calculation are indistinguishable.

Although we concentrated on the symmetric theory, cavity methods [85] hold promise for generalizing this approach to the unsymmetric case, as will be presented elsewhere. The coloring transition is continuous for $q = 2$ and discontinuous for $q \geq 3$, similarly $K - SAT$ is continuous for $K = 2$ and discontinuous for $K \geq 3$. In contrast the VB model of glasses has a continuous phase transition. As found in the rigidity percolation problem [94], processes which require more than 2-connectivity in order to propagate constraint have a tendency toward first order transitions. However, a counter example is rigidity percolation on triangular lattices, where the rigidity transition is continuous [93]. It thus seems a difficult task to determine the conditions which produce continuous as opposed to discontinuous percolation transitions in complex combinatorial problems.

Chapter 5

Applications to System Biology

Research at the intersection of the biological and computational sciences holds the potential to enable a number of important advances for both communities. Computational models of cell regulatory networks and biochemical signaling cascades are being constructed to elucidate the inner working of living cells. Biologists are utilizing these models to explore the logical implication of alternative competing hypothesis, to design drugs that are highly selective for specific targets, and to control the behaviors of cells in response to external inputs. Similarly, advances in the biological sciences are being used to drive innovation in the design of new computing architectures based on biomolecules. The inherent ability of DNA and RNA nucleotides to perform very big computations is being exploited to solve NP hard computational problems.

5.1 Combinatorial Optimization Methods for Dissecting Gene Regulatory Networks During Neuronal Differentiation

Combinatorial and coordinated actions of transcription factors and signaling proteins dictate the determination of cell fate during retinal development. *NRL* (neural retina leucine zipper) is the key transcriptional regulatory protein, which initiates a cascade of molecular events leading to functional rod photo-receptors from committed post-mitotic precursors. It controls the expression of most, if not all, rod-specific genes including the visual pigment rhodopsin. When *NRL* is absent, a rod precursor changes its course and becomes a cone, implying the existence of pools of progenitor cells that can acquire either a rod or a cone cell fate. To understand the gene regulatory networks (*GRNs*) during photoreceptor differentiation, Swaroop's group [36] generated a transgenic mouse line (*NRL* :: *GFP*) that expresses enhanced green fluorescent protein (*EGFP*), under the control of *NRL* promoter, specifically in rod photo-receptors. Using *FACS*-purified photoreceptors, we have produced genome-wide expression profiles at five different developmental time points corresponding to distinct stages of differentiation and from multiple mouse mutants. These studies have provided huge lists of differentially expressed genes. However, manually sifting through the datasets to postulate downstream targets and networks of co-regulated genes have been highly cumbersome and error-prone. We therefore develop computational methods for constructing key functional paths in the gene regulatory network for rod/cone differentiation. Technologies developed in these studies should find wider applications in other microarray-based investigations. A comprehensive understanding of *GRNs* might lead to better design of drug targets for macular degeneration and retinal dystrophies.

5.1.1 Computational Background

To identify functional motifs in gene regulatory networks, we seek develop new methods to cluster genes in a systematic manner, beginning with the most highly correlated gene pairs. At each level in our clustering procedure, we identify the gene pair in each cluster with the strongest correlation. The most highly correlated gene pairs occur in small clusters and are interpreted as functional motifs in the gene regulatory network. Motifs extracted from different knockout experiments and at different developmental time points provide key insight into functional pathways in the network.

We developed the following clustering method using co-expression data: starting with small correlated clusters, which are candidates for motifs in the regulatory network. Clustering is carried out in a systematic way using Kruskal's algorithm for minimum spanning tree, a well known combinatorial optimization procedure. The most highly correlated genes are clustered first and at later times less highly correlated gene clusters are merged. Each cluster-cluster merging event is characterized by the strength of its co-expression correlation.

Using co-expression data at different time points and for different knockouts, gene expression changes and co-expressed motifs are expected to be modified continuously during development (temporal changes). Using our clustering procedures, the hope is to characterize the way in which development modifies motifs and define how motifs are altered by different knockouts.

5.1.2 Biological Bakground

Retinal development

The vertebrate retina is an intriguingly complex yet a relatively simple model to investigate molecular details underlying complex cellular processes and higher order functions of the central nervous system. It consists of six major neuron types and one glia. During retinal differentiation, these different cell types are generated in a predictable sequence from a common pool of multi-potent progenitors [19], [75].

The competence model of cell fate determination proposes that a heterogeneous pool of multi-potent progenitors passes through states of competence, in that it can produce a specific set of cell types. At the molecular level, this competence is acquired under the influence of extrinsic as well as cell-intrinsic mechanisms, which include transcriptional regulatory proteins [19], [75]. Functional maintenance and survival of mature retina also requires quantitatively precise gene expression; over- or under-expression of certain cell-type specific genes results in retinopathies. Hence, elucidation of transcriptional regulatory networks in developing and mature retina is fundamental in understanding the neuronal differentiation as well as disease pathogenesis.

Photoreceptor differentiation and development

In the mammalian retina, rod and cone photo-receptors account for over 70% of all cells; in most mammals, rods are almost 20-fold higher in number compared to cones. Cones are born earlier than rods during retinal development and rods develop over a much broader temporal window than cones Fig.(5.1).

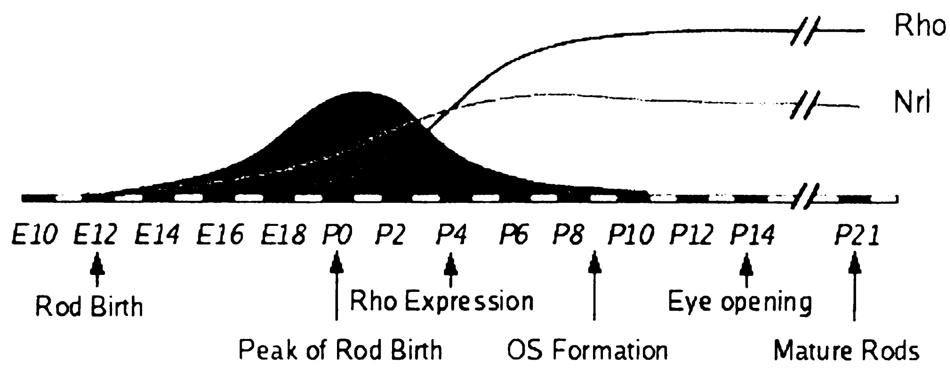


Figure 5.1: Time-line of rod photoreceptor birth, major developmental events, and the kinetics of *NRL* and rhodopsin (*Rho*) gene expression. Rod birth peaks at *P*1 – 2. At *P*6, expression of rhodopsin and several other rod-specific genes is observed. Outer segments begin to form at *P*10 and by *P*28 mature rods are formed. Adapted from Akimoto *et al.* [36]

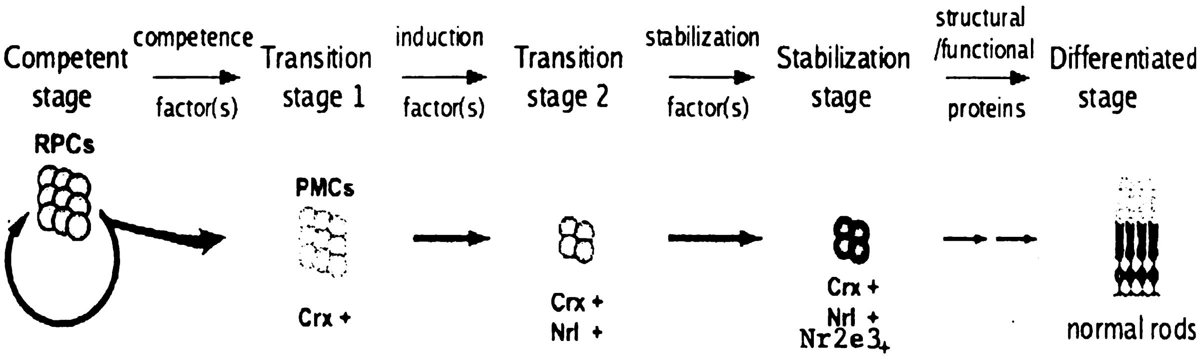


Figure 5.2: A proposed model of photoreceptor differentiation, integrating the transcriptional regulatory functions of *NRL* and *NR2E3* [22].

A majority of rods are born postnatally. Post-mitotic rod precursors exhibit variable delays, depending upon their time of birth (*early* or *late*), before expressing the photo-pigment rhodopsin, a definitive marker of mature rods [36]. Though several transcription factors are shown to regulate rod gene expression, the rod photoreceptor-specific transcription factor NRL is required for rod differentiation [78]. It interacts with cone rod homeobox (CRX), photoreceptor-specific orphan nuclear receptor ($NR2E3$), and other proteins to regulate the expression of most, if not all, rod-specific genes. Ablation of NRL in mice ($NRL^{-/-}$) results in a rodless retina with $S-$cones [26]. A similar phenotype is observed in the retinal degeneration 7 for mice and in patients with enhanced S-cone syndrome, caused by mutations in a rod-specific orphan nuclear receptor $NR2E3$. Significantly, $NR2E3$ expression is undetectable in the $NRL - /-$ retina, suggesting that both NRL and $NR2E3$ share similar regulatory functions. It was shown that $NR2E3$ is a direct transcriptional target of NRL in the retinal developmental hierarchy. Mutations in NRL and $NR2E3$ are associated with distinct retinal disease phenotypes ([112], [62], [104], [87], [55]).

As elaborated in Fig.(5.1) and Fig.(5.2), retinal progenitor cells ($RPCs$) undergo terminal mitosis at specific times during development. The post-mitotic cells ($PMCs$) are directed towards photoreceptor lineage and pass through distinct transition stages. $PMCs$ that actively express NRL are instructed to rod cell fate, whereas those not actively expressing NRL produce cones. Expression of NRL induces rod differentiation, but only subsequent $NR2E3$ expression stabilizes the cell fate by completely repressing cone genes. In the $RD7$ retina, the absence of $NR2E3$ transforms some of the early-born rod precursors to S-cones, while others may acquire rod-cone hybrid phenotype (private communications from Professor Swaroop). In the $NRL - /-$ retina, since NRL and $NR2E3$ are not present, these potential rod precursors adopt the *default* S-cone fate. Ectopic expression of $NR2E3$

in these cells can partially rescue the rod morphology and gene expression but not functionality [22].

Experimental design and data

In order to understand the gene regulatory pathways during rod photoreceptor differentiation, it was generated a transgenic mouse line ($NRL :: GFP$) in wild-type background that expresses enhanced green fluorescent protein ($EGFP$) under the control of NRL promoter, specifically in rod photo-receptors [36]. To directly evaluate the origin of enhanced S-cones in the $NRL - /-$ retina, the wild-type transgenic mice were inter-bred with $NRL - /-$ mice. The GFP-tagged rod precursors in this line take the identity of S-cones in the absence of NRL and $NR2E3$. Since a similar phenotype was observed by the loss of $NR2E3$ function in $RD7$ mouse retina, $NRL :: GFP$ crossed with $RD7$ mice comprised the third mouse line.

To explore additional possible downstream targets of $NR2E3$, it was ectopically expressed in the $NRL - /-$ retina in the fourth mouse line using CRX promoter. Expression of $NR2E3$ in the $NRL - /-$ retina completely suppressed cone differentiation and resulted in morphologically rod-like photo-receptors, which were however not functional, probably due to the absence of NRL [22]. Mating of different transgenic lines with the $NRL :: GFP$ mice allows tagging of rod precursors and mature rods with $EGFP$, facilitating their enrichment by fluorescence-activated cell sorting ($FACS$) [36].

The advantage of using the transgenic mouse lines is that purified photo-receptors from mutant mice can be obtained by $FACS$. Hence, one can generate gene profiles of a specific cell type during development (instead of using the whole retina). The four mutant transgenic mouse lines used have the following phenotype: (i) wild-type where rods are tagged with GFP (both NRL and $NR2E3$ are expressed); (ii) $NRL - /-$ where the GFP-tagged rod precursors take the identity of S-cones (both NRL and $NR2E3$ are absent); (iii) RD7 where rod-cone hybrid cells are tagged with GFP (NRL is expressed, $NR2E3$ is absent); and (iv) $NR2E3$ transgenic in $NRL-/-$ background where non-functional rods are tagged with GFP ($NR2E3$ is expressed, but NRL is absent). Thus, these mouse lines represent different combinations of the two key transcription factors, NRL and $NR2E3$, which are required for normal rod photoreceptor differentiation. Swaroop's generated gene profile data from the $GFP+$ cells purified from the retinas of these transgenic mouse lines at five different developmental time points: $E16$, $P2$, $P6$, $P10$ and $P28$ as is shown in Fig.(5.1). Four independent samples were used at each time point. Affymetrix GeneChips 430 were used for gene profiling.

5.1.3 Construction of gene regulatory networks (GRN) from experimental data

Much of the initial work has focused on the development of techniques for accurate identification of differentially expressed genes and their statistical significance. However, the main difficulty in the analysis lies not in the identification of differentially expressed genes but in their interpretation. Hero's group developed an approach [116] that clusters genes based on their functionality rather than on the level of expression. The group implemented successfully [34] an algorithm that first constructs the relevance network (determined by three parameters: FDR, MAS and p) based on the estimation of pairwise gene profile correlation, and then discovers the biological significance of the network.

The microarray data set available from these studies of mutant mice may be summarized in Fig.(5.3). Data currently exists for four mutant phenotypes at five different time points during photoreceptor development in mammalian retina. However, for a more accurate control of statistical significance of gene pair correlation an adaptive FDR controlling procedure [65] can be used; depending on

$$\begin{array}{c|llll} & & & \text{TIME} \rightarrow & \\ \hline \text{P} & \text{WT} & \{E^{WT}_{i}(t_1)\} & \{E^{WT}_{i}(t_2)\} & \dots \\ \text{H} & & & & \\ \text{E} & & & & \\ \text{N} & & & & \\ \text{O} & \text{rd7} & \{E^{rd7}_{i}(t_1)\} & \{E^{rd7}_{i}(t_2)\} & \dots \\ \text{T} & & & & \\ \text{Y} & \text{NRL}^{-/-} & \{E^{NRL-/-}_{i}(t_1)\} & \{E^{NRL-/-}_{i}(t_2)\} & \dots \\ \text{P} & & & & \\ \text{E} \downarrow & & & & \end{array}$$

Figure 5.3: The phenotype matrix. $E_i^l(t)$ represents the expression level of a set of genes, with each gene labeled by i, corresponding to the l-th experiment (*i.e.* knockout phenotype) at time t.

the number of effects, one can choose which standard error estimator works best in conjunction with the original FDR controlling method.

5.1.4 Discovery of functions/pathways from constructed retinal GRN

Biologists are interested in developing new methods for extracting functional pathways from data like that illustrated in Fig.(5.3). One approach is to extend the relevance network approach [116] and to consider gene expression levels at different time points enabling the identification of the time course of functionally-related genes. The simplest multivariate statistic is the pair correlation between genes, which may be at different time points and/or in different phenotypes. A study of pair correlations at different time points would reveal time delays in the expression of downstream genes. Also, a study of correlations between different mutant phenotypes could reveal the effect of a specific transcription factor (NRL or $NR2E3$) on levels of gene expression. If there is a time delay in the effect of mutation on the expression of downstream genes, this will be revealed by multivariate analysis between different phenotypes at different times. An issue of considerable concern is that experimental uncertainty may obscure the trends in individual gene pairs,

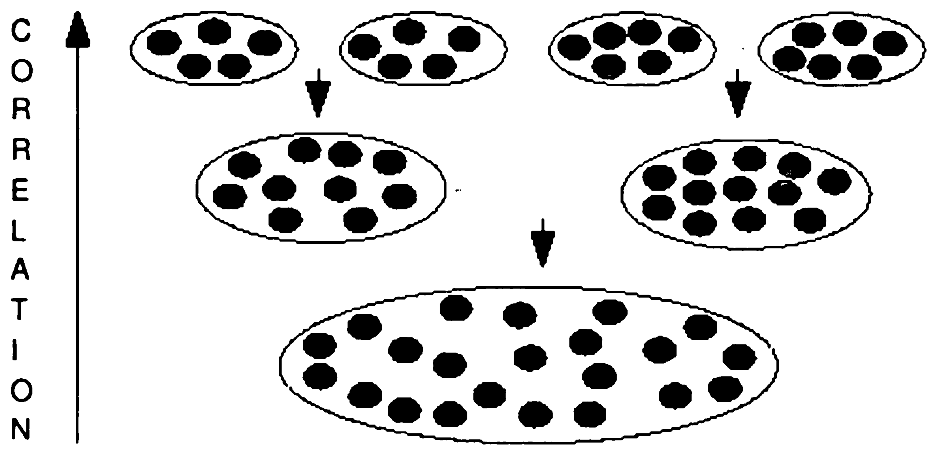


Figure 5.4: Clustering of genes based on correlation level. Small clusters of high correlation appear first. When clusters join, they are linked by an edge (thick edges in the figure) of lower correlation which gives a measure of the confidence associated with the larger cluster. The two different degree of grayness are for the two types of co-expressions: up-, respectively down-regulated genes.

so that identification of gene subsets which behave similarly provides a means to reduce statistical error. To alleviate this, we searched subsets to find gene clusters, which exhibit maximum average correlations.

Clustering of genes. Starting with a set of genes which are not connected, we form clusters of genes that are most correlated by adding the most correlated gene pairs first (first level, from Fig.(5.4)). When clusters merge during this process, by construction, the correlation of last gene pair added before emerging is lower than the correlation of any other gene pair already in the clusters. We will call herein this last link, the weakest link. Larger clusters appear but only through relatively low correlation edges - *the weakest links.* As illustrated, the most strongly correlated clusters are small and are the most important motifs in the gene regulatory network. Using this approach we construct the entire network using the available data for different time points and for one knockout.

5.2 Preliminary results

Using unpublished data from Swaroop group we present some results based on minimum spanning tree algorithm. For sorting the data we are using the false discovery rate (FDR) combined with the confidence interval method (CI) (also called the FDRCI method) of wild type compared to $NRL - /-$ at 5 different time points run with a minimum fold change of 2. The wild type was used as the control. Wild type , is the typical form of an organism, strain, gene, or characteristic as it occurs in nature and it refers to the most common phenotype in the natural population.

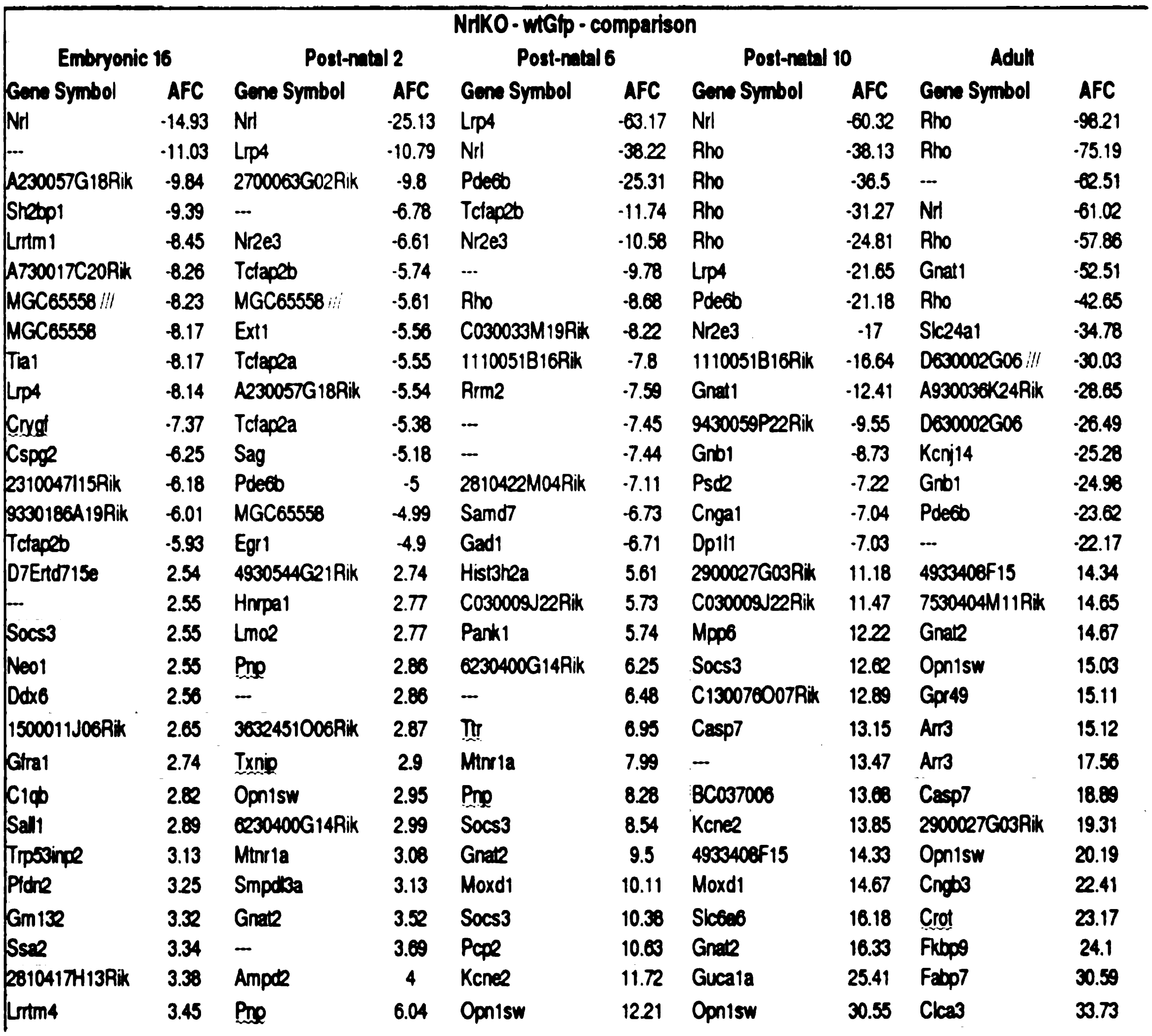
As we already showed in Fig.(5.4) the clustering of genes is based on the level of correlation. We performed two different ways of clustering. The first method is when small clusters of high correlation appears first. When clusters join, they are linked by an edge of lower correlation (which we called *the weakest link*) which gives a measure of the confidence associated with the large cluster. This is actually the minimum spanning tree. The second method of clustering is when small clusters of low correlation appear first. In this way we actually clusters genes which are anti-correlated.

To construct the network we need to determine the weights for the edges that connect the genes. For this we use the following scaling equation:

$$w_{ij} = Abs\left(\frac{l_i \pm l_j}{\mid l_i \mid + \mid l_j \mid}\right)^{\alpha} \tag{5.1}$$

In Eq.(5.1) w_{ij} represents the weight of the link that connects gene i to gene j; l_i, l_j represent the co-expression level for the two genes and α is a scaling parameter. The minus sign in Eq.(5.1) corresponds to the case where the clustering is made with genes that are correlated whereas the plus sign corresponds to the case when the genes are anti-correlated.

Initially we constructed a relatively small network with only 30 genes where



NrlKO - wtGfp - comparison									
Embryonic 16		Post-natal 2		Post-natal 6		Post-natal 10		Adult	
Gene Symbol	AFC	Gene Symbol	AFC	Gene Symbol	AFC	Gene Symbol	AFC	Gene Symbol	AFC
Nrl	-14.93	Nrl	-25.13	Lrp4	-63.17	Nrl	-60.32	Rho	-98.21
---	-11.03	Lrp4	-10.79	Nrl	-38.22	Rho	-38.13	Rho	-75.19
A230057G18Rik	-9.84	2700063G02Rik	-9.8	Pde6b	-25.31	Rho	-36.5	---	-62.51
Sh2bp1	-9.39	---	-6.78	Tcfap2b	-11.74	Rho	-31.27	Nrl	-61.02
Lrrtm1	-8.45	Nr2e3	-6.61	Nr2e3	-10.58	Rho	-24.81	Rho	-57.86
A730017C20Rik	-8.26	Tcfap2b	-5.74	---	-9.78	Lrp4	-21.65	Gnat1	-52.51
MGC65558 ///	-8.23	MGC65558 ///	-5.61	Rho	-8.68	Pde6b	-21.18	Rho	-42.65
MGC65558	-8.17	Ext1	-5.56	C030033M19Rik	-8.22	Nr2e3	-17	Slc24a1	-34.78
Tia1	-8.17	Tcfap2a	-5.55	1110051B16Rik	-7.8	1110051B16Rik	-16.64	D630002G06 ///	-30.03
Lrp4	-8.14	A230057G18Rik	-5.54	Rrm2	-7.59	Gnat1	-12.41	A930036K24Rik	-28.65
Crygf	-7.37	Tcfap2a	-5.38	---	-7.45	9430059P22Rik	-9.55	D630002G06	-26.49
Cspg2	-6.25	Sag	-5.18	---	-7.44	Gnb1	-8.73	Kcnj14	-25.28
2310047I15Rik	-6.18	Pde6b	-5	2810422M04Rik	-7.11	Psd2	-7.22	Gnb1	-24.98
9330186A19Rik	-6.01	MGC65558	-4.99	Samd7	-6.73	Cnga1	-7.04	Pde6b	-23.62
Tcfap2b	-5.93	Egr1	-4.9	Gad1	-6.71	Dp1l1	-7.03	---	-22.17
D7Ertd715e	2.54	4930544G21Rik	2.74	Hist3h2a	5.61	2900027G03Rik	11.18	4933408F15	14.34
---	2.55	Hnrpa1	2.77	C030009J22Rik	5.73	C030009J22Rik	11.47	7530404M11Rik	14.65
Socs3	2.55	Lmo2	2.77	Pank1	5.74	Mpp6	12.22	Gnat2	14.67
Neo1	2.55	Pnp	2.86	6230400G14Rik	6.25	Socs3	12.62	Opn1sw	15.03
Ddx6	2.56	---	2.86	---	6.48	C130076O07Rik	12.89	Gpr49	15.11
1500011J06Rik	2.65	3632451O06Rik	2.87	Ttr	6.95	Casp7	13.15	Arr3	15.12
Gfra1	2.74	Txnip	2.9	Mtnr1a	7.99	---	13.47	Arr3	17.56
C1qb	2.82	Opn1sw	2.95	Pnp	8.28	BC037006	13.68	Casp7	18.89
Sall1	2.89	6230400G14Rik	2.99	Socs3	8.54	Kcne2	13.85	2900027G03Rik	19.31
Trp53inp2	3.13	Mtnr1a	3.08	Gnat2	9.5	4933408F15	14.33	Opn1sw	20.19
Pfdn2	3.25	Smpdl3a	3.13	Moxd1	10.11	Moxd1	14.67	Cngb3	22.41
Gm132	3.32	Gnat2	3.52	Socs3	10.38	Slc6a6	16.18	Crot	23.17
Ssa2	3.34	---	3.69	Pcp2	10.63	Gnat2	16.33	Fkbp9	24.1
2810417H13Rik	3.38	Ampd2	4	Kcne2	11.72	Guca1a	25.41	Fabp7	30.59
Lrrtm4	3.45	Pnp	6.04	Opn1sw	12.21	Opn1sw	30.55	Clca3	33.73

Figure 5.5: The FDRCI output of wild type compared to $NRL - / -$ at 5 different time points run with a minimum fold change of 2. Only the most down- and -up 30 regulated genes are shown.

we choose the most 15 up- and the most 15 down- co-expressed anti-correlated genes as is shown in Fig.(5.5).

Fig.(5.6) shows the anti-correlation network for three different time points: embryonic 16, post-natal 6 and two months (adult). As we can see from this figure, for the embryonic 16 time point, the first link added is between the gene *leucine rich repeat transmembrane neuronal* 4 (*Lrtm*4) which is a rat tetraspan protein and gene *transcription factor* $AP-2$ *beta* ($Tcfap2b$). As time evolves the network becomes less bushy as is shown in Fig.(5.6) for post-natal 6 and adult. In the same way

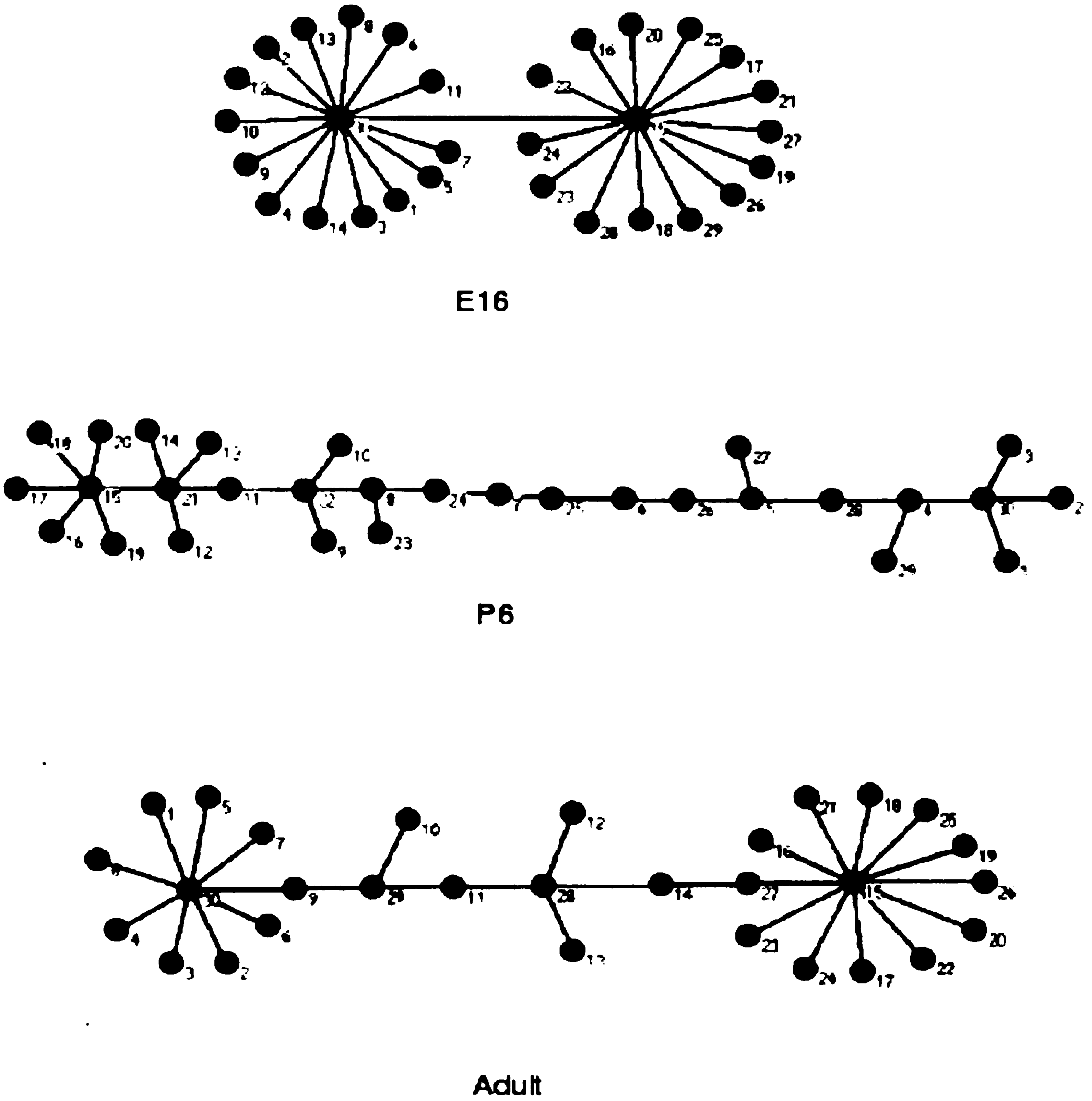


Figure 5.6: Minimum Spanning Tree corresponding to the most highly anti-correlated 30 genes at embryonic 16, post-natal 6 and 2 months (adult) for NrlKO-wtGfp.

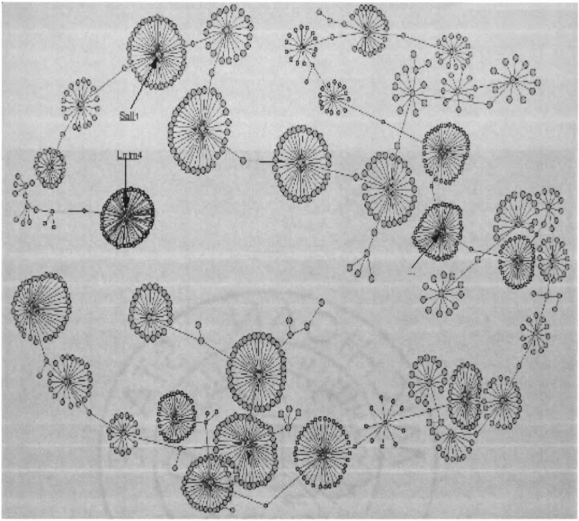


Figure 5.7: Minimum Spanning Tree corresponding to the highly anti-correlated genes at embryonic 16 for NrlKO-wtGfp.

that we constructed the network from Fig.(5.6), using the data provided from Swaroop's lab we constructed the whole network corresponding to the $E16$ time point. Fig.(5.7) shows this network where 961 genes were used. From this network we can identify the most connected genes which are also called *hubs*. The presence of hubs is very important in keeping together the network. As time evolves, the size of the hubs is reduced (Fig.(5.8),Fig.(5.9),Fig.(5.10),and Fig.(5.11)). For the adult time point, the structure of the network shows again a bushy structure as is shown in Fig.(5.11). As we can see from this figure, more hubs are present but with lower

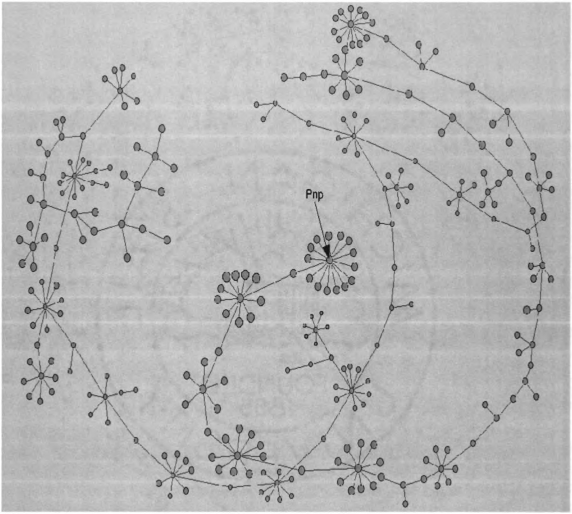


Figure 5.8: Minimum Spanning Tree corresponding to the highly anti-correlated genes at post-natal 2 for NrlKO-wtGfp.

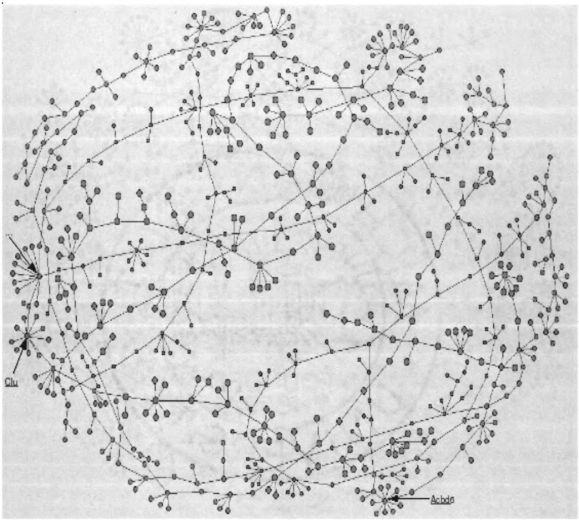


Figure 5.9: Minimum Spanning Tree corresponding to the highly anti-correlated genes at post-natal 6 for NrlKO-wtGfp.

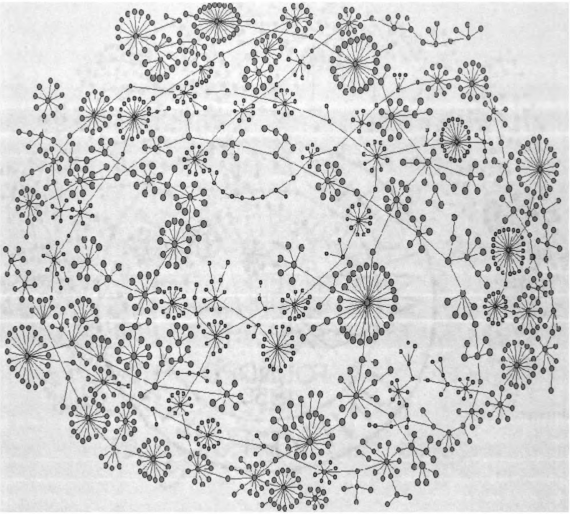


Figure 5.10: Minimum Spanning Tree corresponding to the highly anti-correlated genes at post-natal 10 for NrlKO-wtGfp.

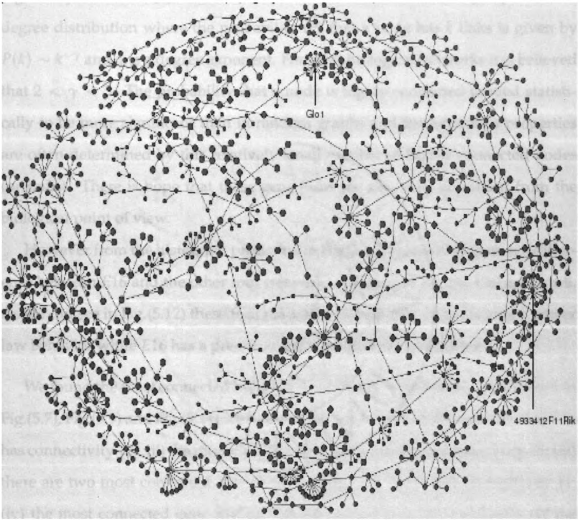


Figure 5.11: Minimum Spanning Tree corresponding to the highly anti-correlated genes at 2 months (adult) for NrlKO-wtGfp.

connectivity. By looking at the network we can see that the network corresponding to time points $P2$, $P6$, $P10$ and adult resemble networks. Scale-free networks proved of crucial importance in understanding many inter-disciplinary areas spanning from Internet [97] and world wide web [16] to social networks [76] and gene regulatory networks [6]. Scale-free networks are characterized by a power-law degree distribution where the probability (P) that a node has k links is given by $P(k) \sim k^{-\gamma}$ and γ is a degree exponent. For most biological networks it is believed that $2 < \gamma < 3$. The probability that a node is highly connected proved statistically to be more significant than in random graphs and the network's properties are often determined by this relatively small number of highly connected nodes (or hubs). There is hope that these gene-hubs are also very important from the biological point of view.

However from the histogram presented in Fig.(5.12) we see that there is a difference between $E16$ and the other four networks obtained at $P2$, $P6$, $P10$ and adult. As we can see in Fig.(5.12) these four networks display the typical case of a power law behavior while $E16$ has a predominance of highly connected hubs.

We found the most connected hubs for five different time points. As is shown in Fig.(5.7), Fig.(5.9) and Fig.(5.11) we have: (i) the most connected gene $Lrrtm4$ at $E16$ has connectivity 101;(ii) the most connected gene Pnp at $P2$ has connectivity 15; (iii) there are two most connected, one of them being Clu at $P6$ with connectivity 11; (iv) the most connected gene $Rtel$ at $P10$ has connectivity 28, and finally (v) the most connected gene $4933412F11Rik$ at 2 months has connectivity 15.

Next, we constructed the correlated network (the minus sign case in Eq.(5.1)) for three different time-points: $e16$, post-natal 2 and post-natal 6. The results are presented in Fig.(5.13), Fig.(5.14) and Fig.(5.15). These three networks resemble scale free networks. For the correlated case the network at $E16$ looks like a scale-free network, while for the anti-correlated case and for the same time point this is

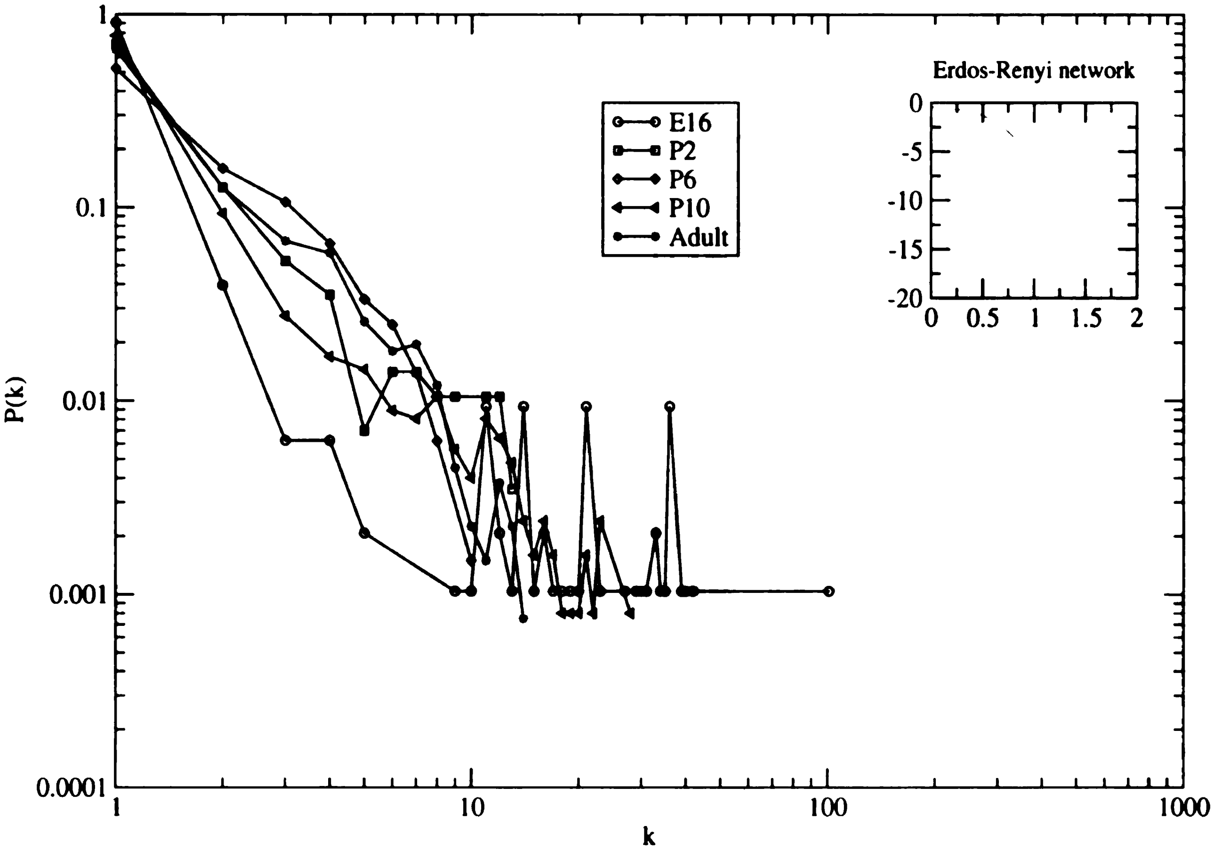


Figure 5.12: k vs. N(k) for the anti-correlated case corresponding to the 5 networks corresponding to the anti-correlated case.

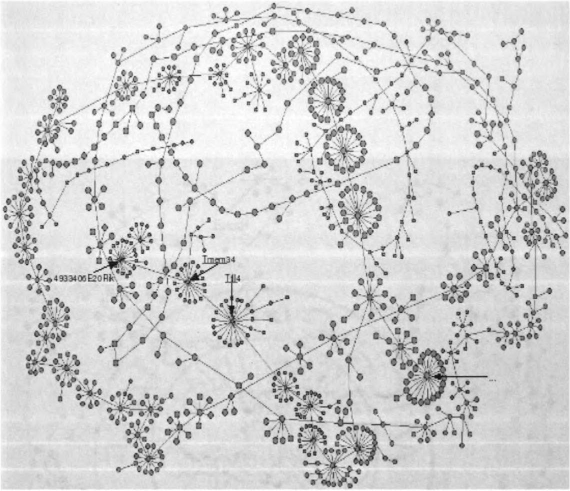


Figure 5.13: Minimum Spanning Tree corresponding to the highly correlated genes at embryonic 16 for NrlKO-wtGfp.

not the case.

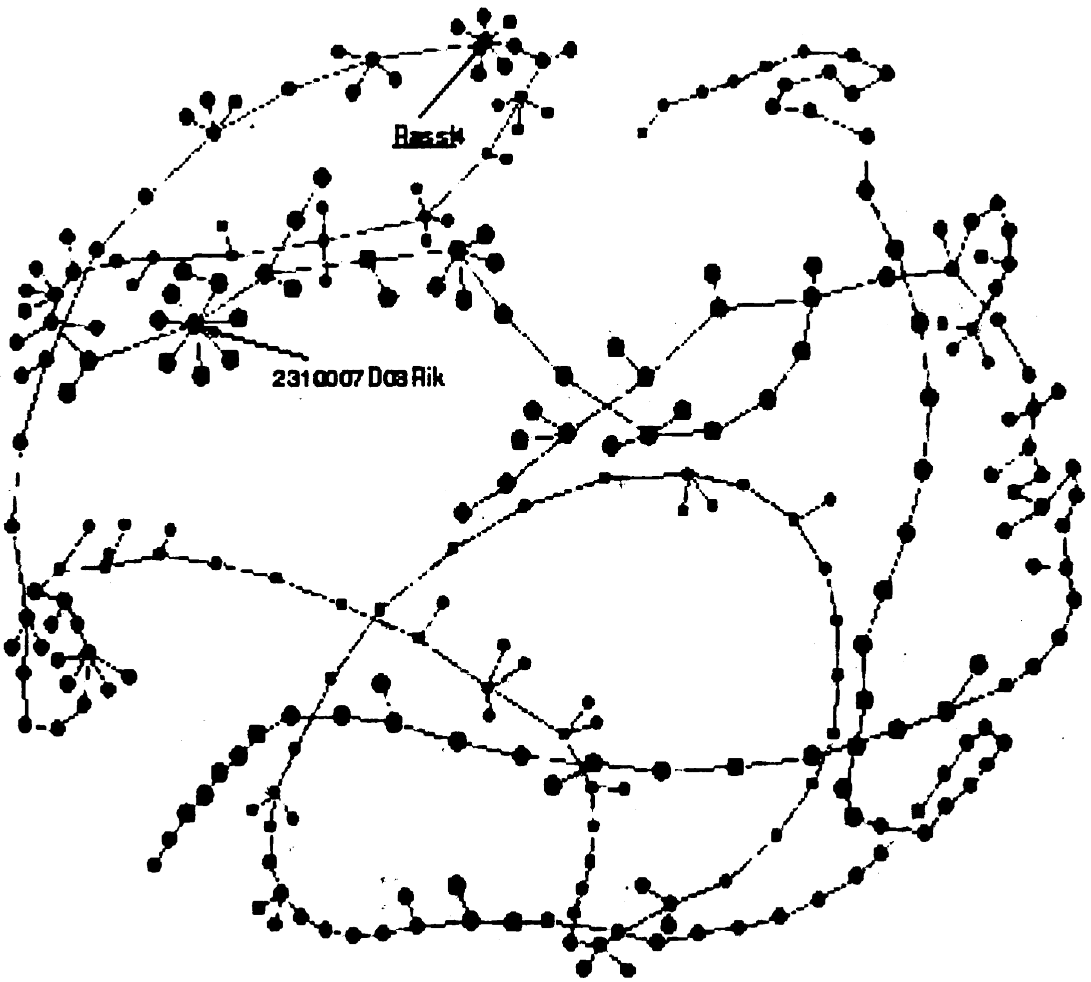


Figure 5.14: Minimum Spanning Tree corresponding to the highly correlated genes at post-natal 2 for NrlKO-wtGfp.

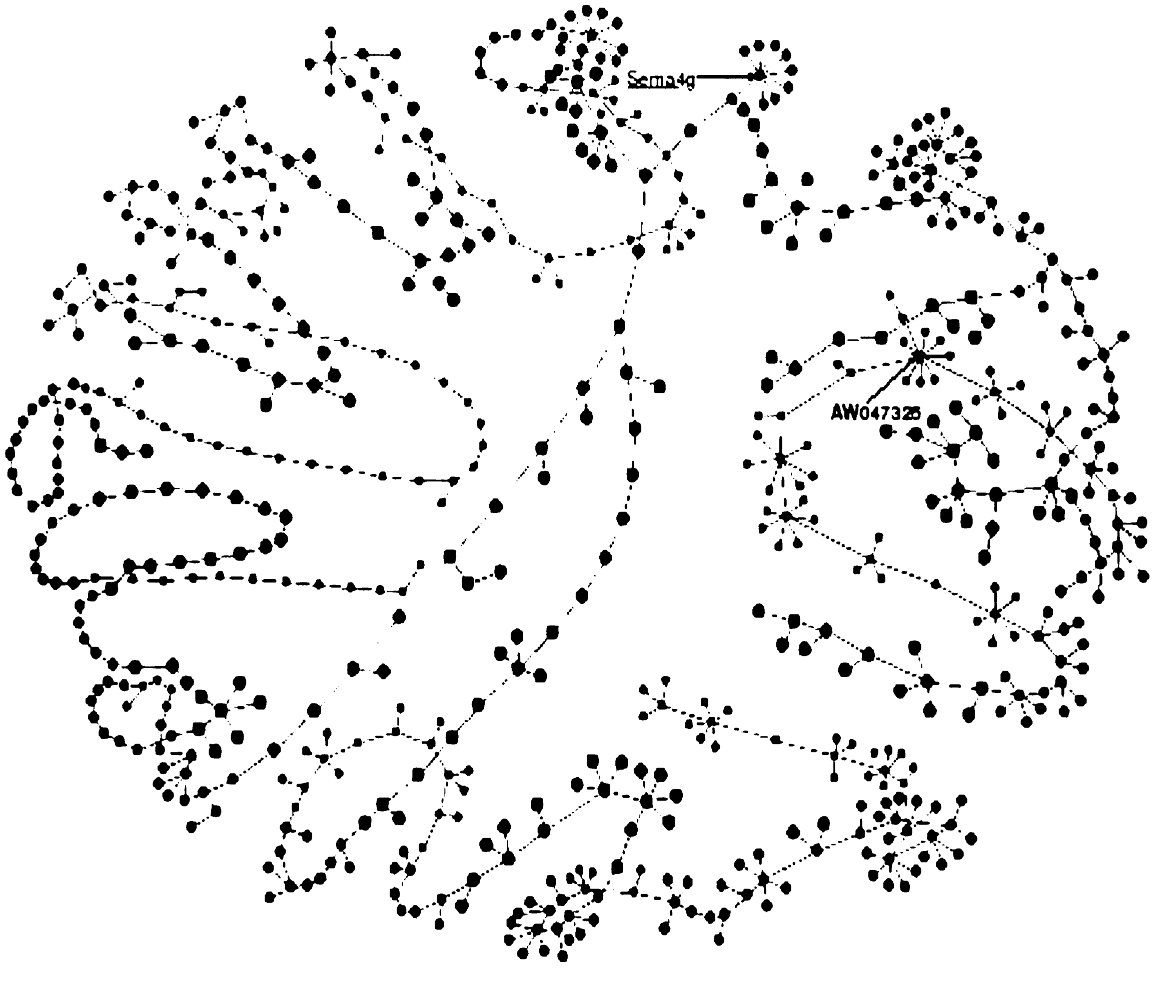


Figure 5.15: Minimum Spanning Tree corresponding to the highly correlated genes at post-natal 6 for NrlKO-wtGfp.

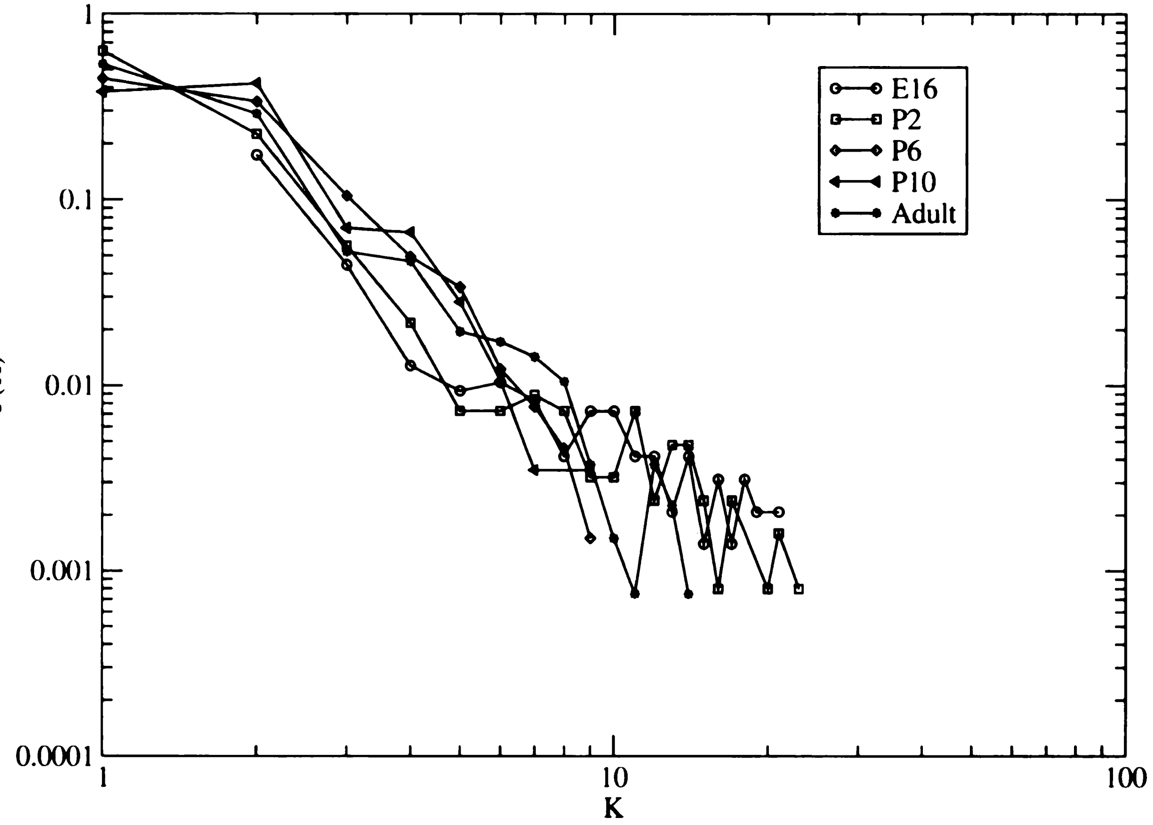


Figure 5.16: k vs. N(k) for the anti-correlated case corresponding to the 5 networks corresponding to the correlated case.

We found the hubs for the five different time points that we considered throughout this analysis. As it is shown in Fig.(5.13), Fig.(5.14) and Fig.(5.15) we found: (i) the most connected gene *Ttll44* at *E*16 has connectivity 21;(ii) the most connected gene 2310007*D*03*Rik* at *P*2 has connectivity 9; (iii) the most connected gene *AW*047325 at *P*6 has connectivity 10; (iv) the most connected gene *Mak*3 at *P*10 has connectivity 23, and finally (v) the most connected gene *Kifap*3 at 2 months has connectivity 15. In Fig.(5.16) we draw the histograms for the five networks corresponding to the correlated case; $N(k)$ represents the number of genes with connectivity k. Although this work is preliminary we have identified several key candidates as targets for treatment of different retinal diseases. In our networks, genes are vertices and pairwise co-expressions are network edges. Traditional work on gene networks focuses on the two aspects: discovery of networks which are statistical significant (False Discovery Rate) and discovery of networks which are biological significant (Minimum Acceptable Strength).

We used unpublished data from mice retina provided from Swaroop's lab from University of Michigan, but our method of clustering and finding networks can be applied to any other biological system not only to retina. We plan to take this work to the next level and identify the topology of cellular pathways involved in photoreceptor differentiation and to to identify possible targets for treatement of different eye diseases.

Chapter 6

Conclusion

This thesis covers two areas of application of statistical physics. Firstly it covers application of statistical physics to computer science problems, namely to complex network systems. Seccondly it covers application of statistical physics to biology.

Application to complex networks from computer science is based on one of the most important properties specific to NP problems namely that many apparently different problems can be mapped to each other so that solutions are preserved under the mapping. An important observation is that problems whose order parameter is at the critical boundary are typically hard. We considered problems from combinatorial optimization as models in statistical physics. The cost function was renamed as the Hamiltonian, a term more familiar to physicists and the ground states correspond to the solutions of optimization problems.

We outlined a few application of polynomial combinatorial optimization algorithms useful in extracting the universal zero-temperature properties for different disordered systems. Dijkstra's algorithm can be used for models of non-directed elastic lines on graphs with isotropically correlated random potentials. The preflow-push algorithm for minimum-cut/max-flow problems can be used to investigate transitions that occur in a model for elastic manifolds in a periodic potential in the presence of point disorder.

The existence of many degenerate states and/or metastable states is due to frustration. It is well known among computational physicists that for investigating disordered systems, finding the equilibrium states (or the ground states) is nearly impossible for large systems. Usually the size of the systems used is at most of the order of few hundred variables. A break-through technique that we discussed came from statistical physics where the cavity method was used successfully for systems with few millions of variables.

Problems that are NP-complete are unlikely to be solved exactly. In statistical physics, we are interested in results in the thermodynamic limit $N \to \infty$ rather than for finite systems. So after we mapped problems from computer science to problems from statistical physics, we focused on classifying problems from statistical physics according to their computational complexity. For many NP-complete problems one more order parameter can be defined, and usually hard instances occur around particular critical values of these order parameters. These critical values form a boundary that separates the space of problems into three regions. One under-constrained region where the density of solutions is large so we can easily find solutions, one over-constrained region where is very unlikely to find a solution and a third region which is between these two under- and over-constrained regions. Usually in this third region hard problems occur. We foccused on phase transitions from statistical physics and from computer science point of view but phase transitions are also interesting from the mathematical point of view because they are singular points.

Mainly, in this thesis the novel contribution is presented in Chapter 4 and in Chapter 5.

In Chapter 4 we introduced a more elegant and simple method for NP-complete problems, starting with Viana-Bray model and then continuing with the K-SAT and Coloring problems. We studied analytically and numerically these three hard problems using a method similar to the message passing procedure. A local order parameter which is important in the analysis of phase transition in frustrated combinatorial systems is the probability that a node is frozen in a particular state. There is a percolative transition when an infinite connected cluster of these frozen nodes emerges. The emergence of frozen order can be considered to be a form of constraint percolation which proved to be useful in making analogies with rigidity percolation and its associated matching problem. We showed that a giant frozen cluster emerges at the phase transition and the emergence is continuous for 2-SAT problem and is discontinuous for the $K \geq 2$ case. For the Coloring problem on random Bethe lattice, when $q = 2$ (so the two colors case), the onset of constraint percolation is of second order, similar to emergence of giant component in random graphs. However, using numerical methods it was shown that the onset of constraint percolation for $q = 3$ (the three colors case) is of first order.

Advances in molecular biology, analytical and computational techniques are proving to be powerful tools to systematically investigate the complex molecular processes underlying biological systems. In particular, using high-throughput gene expression arrays, we are able to measure the output of the gene regulatory networks. To identify functional motifs in gene regulatory networks, in Chapter 5 we developed novel methods to cluster genes in a systematic manner, beginning with the most highly correlated gene pairs. At each level in our clustering procedure, we identified the gene pair in each cluster with the strongest correlation. The most highly correlated gene pairs occur in small clusters and are interpreted as functional motifs in the gene regulatory network. As a future work we will focus on discovering functional pathways in these networks and on developing methods to control the activity of key functional pathways in these networks, particularly signaling pathways.

BIBLIOGRAPHY

Bibliography

[1] D. Achlioptas, L. M. Kirousis, E. Kranakis, and D. Krizans. *Rigorous Results for Random (2+p)SAT.*, 14:63, 1997.

[2] D. Achlioptas, A. Naor, and Y. Peres. Rigorous Location of Phase Transitions in Hard Optimization Problems.

[3] L. M. Adleman. Molecular Computation of Solutions to Combinatorial Problems. *Science*, 266(5187):1021–1024, 1994.

[4] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms and Applications.* Prentice-Hall, NJ, 1993.

[5] B. Aspvall, M. F. Plass, and R. E. Tarjan. A Linear-Time Algorithm for Testing the Truth of Certain Quantified Boolean Formulas. *Inform. Process. Letter*, 8:121–123.

[6] A. L. Barabasi and R. Albert. Emergence of Scaling in Complex Networks. *Science*, 286:509–512, 1999.

[7] F. Barahona. On the Computational Complexity of Ising Spin Glass Models . *J. Phys. A.*, 15:3241–3253, 1982.

[8] F. Barahona, M. Grotschel, M. Junger, and G. Reinelt. An Application of Combinatorial Optimization to Statistical Physics and Circuit Layout Design. *Oper. Res.*, 36:493–513, 1988.

[9] M. N. Barber. Finite-Size Scaling. *Phase Transitions and Critical Phenomena*, 8:145–266, 1983.

[10] S. Bastea. Degeneracy Algorithm for Random Magnets. *Phys. Rev. E*, 58:7978, 1998.

[11] E. B. Baum. Building an Associative Memory Vastly Larger Than the Brain. *Science*, 268:542–545, 1995.

[12] R. J. Baxter. Colorings of a Hexagonal Lattice. *J. of Mathematical Physics*, 11:784–789, 1970.

[13] K. Binder and A. P. Young. Spin Glasses: Experimental Facts, Theoretical Concepts and Open Questions. *Rev. Mod. Phys.*, 58:801, 1986.

[14] G. Biroli, S. Coco, and R. Monasson. Phase Transitions and Complexity in Computer Science: An Overview of the Statistical Physics Approach to the Random Satisfiability Problem. *Physica A*, 306.

[15] G. Biroli, R. Monasson, and M. Weigt. A Variational Description of the Ground-State Structure in Random Satisfibility Problems. *Eur. Phys. J. B*, 14:551, 2000.

[16] James M. Bower and Hamid Bolouri (editors). *Computational Modeling of Genetic and Biochemical Networks.* Computational Molecular Biology Series, Addison-Wesley, Reading, Mass., 2001.

[17] A. Braunstein, M. Mezard, and R. Zecchina. Survey Propagation: an Algorithm for Satisfiability. *Random Structures and Algorithms*, 27:201–226, 2005.

[18] A. Braunstein, R. Mulet, A. Pagnani, M. Weigt, and R. Zecchina. Polynomial iterative algorithms for coloring and analyzing random graphs. *Cond-Mat*, 0304558, 2003.

[19] C. L. Cepko, C. P. Austin, X. Yang, M. Alexiades, and D. Ezzeddine. Cell Fate Determination in the Vertebrate Retina. *PNAS*, 93:589–595, 1996.

[20] J. Chalupa, P.L. Leath, and G. R. Reich. Bootstrap percolation on a Bethe lattice. *J. Phys. C.*, 12:L31, 1979.

[21] P. Cheeseman, B. Kafensky, and W. M. Taylor. Where the Really Hard Problems Are. *Proceedings of the 12th IJCAI*, pages 331–337, 1991.

[22] H. Cheng, T. S. Aleman, A. V. Cideciyan, R. Khanna, S. G. Jacobson, and A. Swaroop. In Vivo Function of the Orphan Nuclear Receptor NR2E3 in Establishing Photoreceptor Identity During Mammalian Retinal Development. *In Press.*

[23] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms.* MIT press, 1990.

[24] J. Culberson and I. Gent. Empirical Evidence for an Asymptotic Discontinuity in the Backbone of the 3-Coloring Phase Transition. *APES-16-1999*, 16, 1999.

[25] J. Culberson and I. Gent. Frozen Development in Graph Coloring. *Theor. Comp. Sci.*, 265:227, 2001.

[26] L. L. Daniele, C. Lillo, A. L. Lyubarsky, S. S. Nikonov, N. Philp, A. J. Mears, A. Swaroop, D. S. Williams, and E. Pugh Jr. Cone-Like Morphological, Molecular, and Electrophysiological Features of the Photoreceptors of the NRL Knockout Mouse. *IOVS*, 46:2156–2167, 2005.

[27] M. Davis and H. Putnam. A Computing Procedure for Quantification Theory. *Assoc. Comput. Mach.*, 7:201–215, 1960.

[28] C. Djurberg, K. Jonason, and P. Nordblad. Magnetic Relaxation Phenomena in a CuMn Spin Glass. *arXiv:cond-mat/9810314*, 1998.

[29] O. Dubois. Counting the Number of Solutions for Instance of Satisfiability. *Theor. Comp. Science*, 81:49–64, 1991.

[30] O. Dubois, R. Monasson, B. Selman, and R. Zecchina. Phase Transitions in Random Combinatorial Problems. In O. Dubois, R. Monasson, B. Selman, and R. Zecchina, editors, *Theoretical Computer Science, volume 265*. 2001.

[31] P. M. Duxbury, D. J. Jacobs, M. F. Thorpe, and C. Moukarzel. Floppy Modes and the Free Energy: Rigidity and Connectivity Percolation on Bethe Lattice. *Phys. Rev. E*, 59:2084, 1999.

[32] S. F. Edwards and P. W. Anderson. Theory of Spin Glasses. *J. Phys F: Metal Phys.*, 5:965, 1975.

[33] P. Erdös and A. Rényi. On the Evolution of Random Graphs. *Publ. Math. Inst. Hung. Acad. Sci.*, 5:17, 1960.

[34] D. Zhu *et al.* Network Constrained Clustering for Gene Microarray Data. *Bioinformatics*, 21(21):4014–4020, 2005.

[35] K. Sakamoto *et. al.* Molecular Computation by DNA Hairpin Formation. *Science*, 288:1223–1226, 2000.

[36] M. Akimoto *et al.* Targeting GFP to Newborn Rods by NRL Promoter and Temporal Expression Profiling of Flow-Sorted Photoreceptors. *PNAS*, 10(103):3890–3895, 2006.

[37] Q. Liu *et. al.* DNA Computing on Surfaces. *Nature*, 403:175–179, 2000.

[38] Y. Benenson *et. al.* An Autonomous Molecular Computer for Logical Control of Gene Expression. *Nature*, (429):423–429, 2004.

[39] D. Faulhammer, A. R. Cukras, R. J. Lipton, and L. F. Landweber. Molecular Computation: RNA Solutions to Cheese Problem. *Proc. Natl. Acad. Sci. USA*, 97(4):1385–1389, 2000.

[40] C. W. Fay, J. W. Liu, and P. M. Duxbury. Maximum Independent Set on Diluted Triangular Lattices. *Phys. Rev. E*, 73:056112, 2006.

[41] M. E. Fisher and J. W. Essam. Some Cluster Size and Percolation Problems. *J. Math. Phys.*, 2:609, 1961.

[42] B.J. Frey and D.J.C. Mackay. A Revolution: Belief Propagation in Graphs with Cycles. *Adv. in Neural Information Porcessing System*, 10, 1998.

[43] A. M. Frieze. On the Independence Number of Random Graphs. *Discrete Mathematics*, 81:3171–175, 1990.

[44] Y. Fu. *The Uses and Abuses of Statistical Mechanics in Computational Complexity.*, volume 1. Lectures in Sciences and Complexity. Addison-Wesley, Reading, Mass., 1989.

[45] Y. Fu and P.W. Anderson. Application of Statistical Mechanics to NP Complete Problems in Combinatorial Optimization. *J. Phys. A: Math. Gen.*, 19:1605, 1986.

[46] M. R. Garey and D. S. Johnson. *Computers and Intractability.* Academic Press., 1979.

[47] P. G. Gazmuri. Independent Sets in Random Sparse Graphs. *Networks*, 14:367–377, 1984.

[48] I. P. Gent, E. MacIntryre, P. Prosser, and T. Walsh. Scaling Effects in the CSP Phase Transition. *Principles and Practice of Constraint Programming*, pages 70–87, 1995.

[49] I. P. Gent and T. Walsh. The TSP Phase Transition. *Artificial Intelligence*, 88(Issue 1-2):349–358, 1996.

[50] I.P. Gent and T. Walsh. Phase Transitions and Annealed Theories: Number Partitioning as a Case Study. *12th European Conference on Artificial Intelligence*, pages 170–174, 1996.

[51] A. Goerdt. A Threshold for Unsatisfiability. *Journal of Computer and System Sciences*, 53:469–486.

[52] F. Guarnieri, M. Fliss, and C. Bancroft. Making DNA Add. *Science*, 273:220–223, 1996.

[53] A. Suyama H. Yoshida. Solution to 3-SAT by Breadth First Search. *DNA Based Computers: DIMACS Series in Discrete Mathematics and Theoretical Computer Science.*, 54(5):9–20, 1999.

[54] F. Hadlock. Finding a Maximum Cut of a Planar Graph in Polynomial Time. *SIAM. J. Comput*, 4:221–225, 1975.

[55] N. B. Haider, S. G. Jacobson, A. V. Cideciyan, R. Swiderski, L. M. Streb, C. Searby, G. Beck, R. Hockey, D. B. Hanna, and S. Gorman. Mutation of a Nuclear Receptor Gene, NR2E3, Causes Enhanced S Cone Syndrome, a Disorder of Retinal Cell Fate. *Nat Genet*, 24:127–131.

[56] A. K. Hartmann and M. Weigt. Statistical Mechanics Perspective on the Phase Transition in Vertex Covering of Finite-Connectivity Random-Graphs. *Theor. Comp. Sci.*, 265:199, 2001.

[57] A. K. Hartmann and M. Weigt. Statistical Mechanics of the Vertex-Cover Problem. *Journal of Physics A: Mathematical and General*, 36:11069–11093, 2003.

[58] B. Hayes. Can't Get No Satisfaction. *American Scientist*, March - April 1997.

[59] G. L. Van Hemmen and R. G. Palmer. The Replica Method and Solvable Spin Glass Model. *J. Phys. A*, 12:563, 1979.

[60] T. Hogg, B. A. Huberman, and C. Williams. Phase Transitions and the Search Problems. *Artifical Intelligence*, 81:1–15, 1996.

[61] D. J. Jacobs and M. F. Thorpe. Generic Rigidity Percolation in Two Dimensions. *Phys. Rev. E*, 53:3683, 1996.

[62] S. G. Jacobson, A. Sumaroka, T. S. Aleman, A. V. Cideciyan, S. B. Schwartz, A. J. Roman, R. R. McInnes, V. C. Sheffield, E. M. Stone, and A. Swaroop. Nuclear Receptor NR2E3 Gene Mutations Distort Human Retinal Laminar Architecture and Cause an Unusual Degeneration. *HMG*, 13:1893–1902, 2004.

[63] I. Kanter and H. Sompolinsky. Mean-Field Theory of Spin-Glasses with Finite Coordination Number. *Phys. Rev. Lett.*, 58:164, 1987.

[64] R. M. Karp. Reducibility Among Combinatorial Optimization. *Complexity of Computer Computations*, R. Miller and J. Teacher, eds. Plenum Press, NY:85–103, 1972.

[65] M. K. Kimel and Y. Benjamini. The False Discovery Rate for Multiple Testing in Factorial Experiments. *Techical Report*, 2006.

[66] S. Kirkpatrick, G. Gyorgyi, N. Tishby, and L Troyansky. The Statistical Mechanics of K-Satisfaction. *Adv. Neur. Inform. Proc. Syst.*, 6:439–446, 1994.

[67] S. Kirkpatrick and B. Selman. Critical Behaviour in the Satisfiability of Random Boolean Expressions. *Science*, 264:1297, 1994.

[68] S. Kirkpatrick and D. Sherrington. Infinite-Ranged Models of Spin-Glasses. *Physical Review B*, 17(11):4384, 1978.

[69] S. Kirkpatrick, W.W. Wilcke, R.B. Garner, and H. Huels. Percolation in Dense Storage Arrays. *Physica A*, 314:220–229, 2002.

[70] A. D. Korshunov. The Main Properties of Random Graphs With a Large Number of Vertices and Edges. *Russian Math. Surveys*, pages 121–198, 1985.

[71] F.R. Kschischang, B.J. Frey, and H. A. Loeliger. Factor Graphs and the Sum Product Algorithm. *IEEE Trans. Info. Theory*, 47:498–519, 2001.

[72] S. L. Lauritzen and D. J. Spiegelhalter. Local Computations with Probabilities on Graphical Structures and their Application to Expert Systems. *Journal of the Royal Statistical Society*, B(50):157–224, 1988.

[73] D. Lidar and O. Biham. Simulating Ising Spin Glasses on a Quantum Computer. *Phys. Rev. E*, 56(3):3661–3681, 1997.

[74] R. Lipton. DNA Solution of Hard Computational Problems. *Science*, 268(5210):542–545, 1995.

[75] F. J. Livesey and C. L. Cepko. Vertebrate Neural Cell-Fate Determination: Lessons from the Retina. *Nat. Rev. Neuroscience*, 2:109–118, 2001.

[76] Juyong Park M. E. J. Newman. Why Social Networks Are Different From Other Types of Networks. *Phys. Rev. E*, 68:036122, 2003.

[77] H. Maletta and W. Felsch. Insulating Spin-Glass System $E_xSr_{1-x}S$. *Phys. Rev. B*, 20:1245–1260, 1979.

[78] A. J. Mears, M. Kondo, P. K. Swain, Y. Takada, R. A. Bush, T. L. Saunders, P. A. Sieving, and A. Swaroop. NRL Is Required For Rod Photoreceptor Development. *Nat. Genet.*, 29:447–452, 2001.

[79] S. Mertens. Phase Transition in the Number Partitioning Problem. *prl*, 81(20):4281–4284, 1998.

[80] S. Mertens. A Physicist's Approach to Number Partitioning. *Theoretical Computer Science*, 265:79–108, 2001.

[81] S. Mertens, M. Mézard, and R. Zecchina. Threshold Values of Random K-SAT From the Cavity Method. *Random Structure and Algorithms*, 28(3):340–373, 2006.

[82] M. Mézard and G. Parisi. The Cavity Method at Zero Temperature. *Journal of Statistical Physics*, 111, 2003.

[83] M. Mezard, G. Parisi, and M. Virasoro. *Spin Glass Theory and Beyond: An Introduction to the Replica Method and its Applications.* World Scientific Lectrures Notes in Physics, 1987.

[84] M. Mezard, G. Parisi, and M.A. Virasoro. *Spin Glass Theory and Beyond*, volume 9 of *World Scientific Lecture Notes in Physics*. 1988.

[85] M. Mézard, G. Parisi, and R. Zecchina. Analytic and Algorithmic Solultion of Random Satisfiabilty Problem . *Science*, 297:812–815, 2002.

[86] M. Mézard and R. Zecchina. Random K-Satisfiability: from an Algorithmic Solution to a New Efficient Algorithm. *Phys. Rev. E*, 66:56126, 2002.

[87] A. H. Milam, L. Rose, A. V. Cideciyan, M. R. Barakat, W. X. Tang, N. Gupta, T. S. Aleman, A. F. Wright, E. M. Stone, and V. C. Sheffield. The Nuclear Receptor NR2E3 Plays a Role in Human Retinal Photoreceptor Differentiation and Degeneration. *PNAS*, 99:473–478, 2002.

[88] R. Monasson and R. Zecchina. Entropy of the K-Satisfiability Problem. *Physical Review Letters*, 76(21):3881, 1996.

[89] R. Monasson and R. Zecchina. Statistical Mechanics of the Random K-Satisfiability Model. *Phys. Rev. E*, 56:1357–1370, 1997.

[90] R. Monasson, R. Zecchina, S. Kirkpatrick, and L. Troyansky B. Selman. Determining Computational Complexity From Characteristic Phase Transitions. *Nature*, 400:133–137, 1999.

[91] C. Moukarzel. A Fast Algorithm for Backbones. *Int. J. Mod. Phys. C*, 9:887, 1998.

[92] C. Moukarzel and P. M. Duxbury. Stressed Backbone and Elasticity of Random Centra-Force Systems. *Phys. Rev. Lett.*, 75:4055–4058, 1995.

[93] C. Moukarzel and P. M. Duxbury. Comparison of Rigidity and Connectivity Percolation in Two Dimensions. *Phys. Rev. E*, 59:2614–2622, 1999.

[94] C. Moukarzel, P. M. Duxbury, and P. L. Leath. First Order Rigidity on Cayley Trees. *Phys. Rev. E*, 55:5800–5811, 1997.

[95] R. Mulet, A. Pagnani, M. Weight, and R. Zecchina. Coloring Random Graphs. *Phys. Rev. Lett.*, 89:268701, 2002.

[96] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice Hall, 1982.

[97] Juyong Park and M. E. J. Newman. The Origin of Degree Correlations in the Internet and Other Networks. *Phys. Rev. E*, 68:026112, 2003.

[98] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Academic Press, 1988.

[99] B. Pittel, J. Spencer, and N. Wormald. Sudden Emerges of a Giant K-Core in a Random Graph. *J. Comb. Theory B*, 67:111, 1996.

[100] Q. Quyang, P. D. Kaplan, S. Liu, and A. Libchaber. DNA Solution of the Maximal Clique Problem. *Science*, 278(446-449):446–449, 1997.

[101] S. Ravinderjit, N. Chelyapov, C. Johnson, P. W. K. Rothemund, and L Adleman. Solution of a 20-Variable 3-SAT Problem on a DNA Computer. *Science*, 296:499–502, 2002.

[102] P. W. K. Rothemund. *DNA Computers: DIMACS Series in Discrete Mathematics and Theoretical Computer Science.*, 27:75–119, 1996.

[103] S. Roweis, E. Winfree, R. Burgoyne, N.V. Chelyapov, M.F. Goodman, P.W.K. Rothemund, and L.M. Adleman. A Sticker-Based Model for DNA Computation. *Journal of computational biology*, 5:615–629, 1998.

[104] D. Sharon, M.A. Sandberg, R.C. Caruso, E.L. Berson, and T.P. Dryja. Shared Mutations in NR2E3 in Enhanced S-Cone Syndrome, Goldmann-Favre Syndrome, and Many Cases of Clumped Pigmentary Retinal Degeneration. *Arch. Ophthalmology*, 121:1316–1323, 2003.

[105] D. Sherrington and S. Kirkpatrick. Solvable Model of a Spin-Glass. *Physical Review Letters*, 35(26):1792, December 1975.

[106] P. Svenson and M. G. Nordahl. Relaxation in Graph Coloring and Satisfiability Problems. *Phys. Rev. E*, 59:3983, 1999.

[107] G. Tolouse. Theory of the Frustration Effect in Spin Glasses. *Commun. Phys*, 2(4):115 - 119, 1977.

[108] J. van Mourik and D. Saad. Random Graph Coloring: Statistical Physics Approach. *Phys. Rev. E*, 66:56120, 2002.

[109] B. Vandegriend and J. Culberson. The $g_{n,m}$ Phase Transition Is Not Hard for the Hamiltonian Cycle Problem. *Journal of Artificial Intelligence Research*, 9:219–245, 1998.

[110] L. Viana and A.J. Bray. Phase Diagrams for Dilute Spin Glasses. *J. of Phys. C*, 18:3037, 1985.

[111] M. Weigt and A. K. Hartmann. Minimal Vertex Covers on Finite-Connectivity Random Araphs: A Hard-Sphere Lattice-Gas Picture. *Phys. Rev. E*, 63:056127, 2001.

[112] A. F. Wright, A. C. Reddick, S. B. Schwartz, J. S. Ferguson, T. S. Aleman, U. Kellner, B. Jurklies, A. Schuster, E. Zrenner, and B. Wissinger. Mutation Analysis of NR2E3 and NRL Genes in Enhanced S Cone Syndrome. *Hum Mutat*, 24:439, 2004.

[113] W.T.Freeman, J.S. Yedidia, and Y. Weiss. Constructing Free Energy: Approximations and Generalized Belief Propagations Algorithms. *Technical Report*, 2002.

[114] F. Y. Wu. The Potts Model. *Review Modern Physics*, 54(235), 1982.

[115] J.S. Yedidia, W.T. Freeman, and Y. Weiss. Bethe Free Energies: Kikuchi Approximations and Belief Propagation Algorithms. *Technical Report*, 2001.

[116] D. Zhu and A. O. Hero. Gene Co-Expression Network Discovery with Controlled Statistical and Biological Significance. *ICASSP*, pages 369–372, 2005.