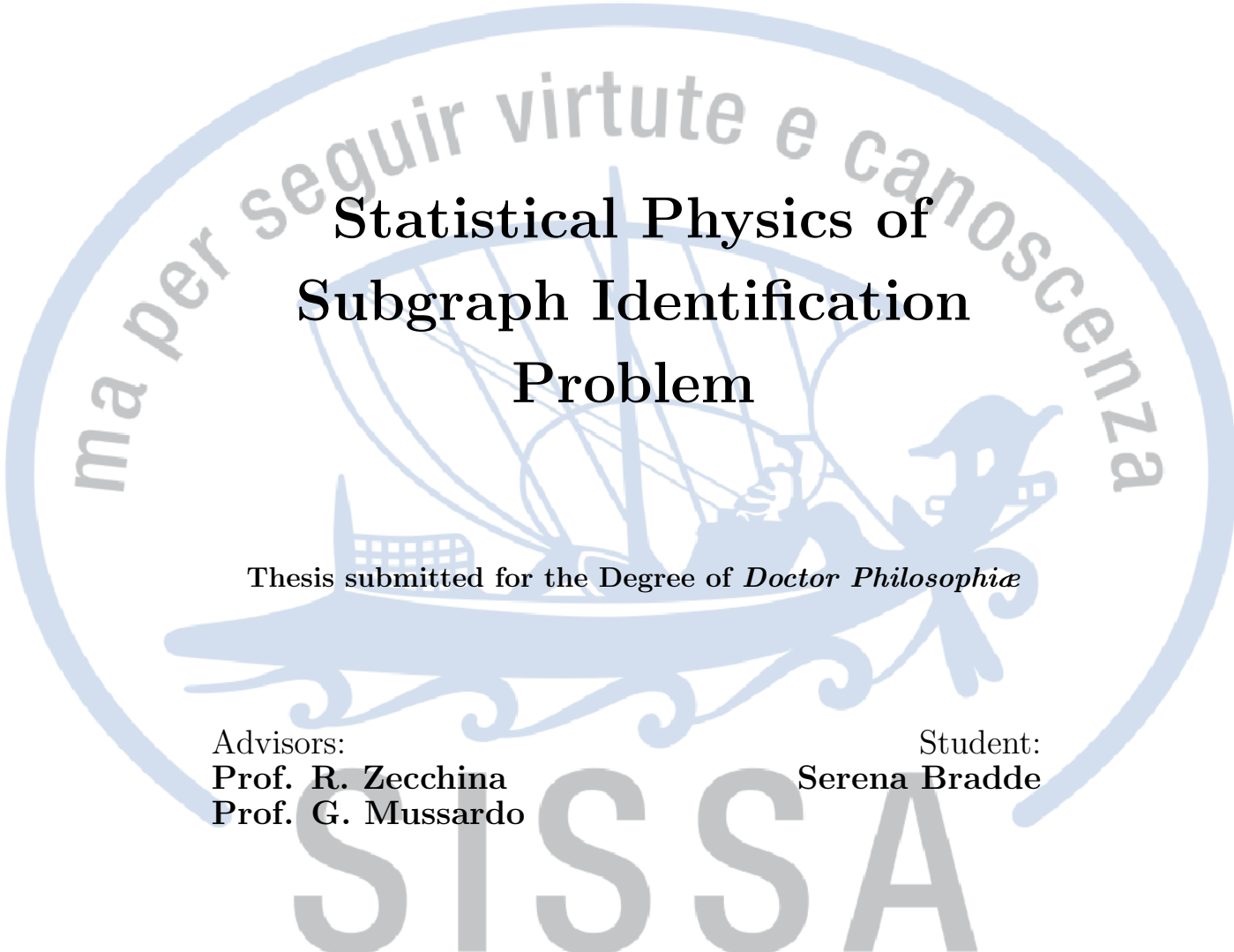


INTERNATIONAL SCHOOL for ADVANCED STUDIES

---

HIGH ENERGY SECTOR  
STATISTICAL PHYSICS CURRICULUM



**Statistical Physics of  
Subgraph Identification  
Problem**

Thesis submitted for the Degree of *Doctor Philosophiæ*

Advisors:  
Prof. R. Zecchina  
Prof. G. Mussardo

Student:  
Serena Bradde

**SISSA**

---

Academic Year 2009/2010



# Contents

<b>Introduction</b>	<b>v</b>
<b>1 The Cavity Method</b>	<b>1</b>
1.1 Optimization and statistical physics . . . . .	1
1.1.1 Graphical Representation . . . . .	2
1.2 Cavity Method and Belief Propagation . . . . .	4
1.2.1 The symmetric solution on a single graph . . . . .	7
1.2.2 Average over the graph ensemble . . . . .	11
1.2.3 Belief propagation equations over the ensemble . . . . .	12
1.3 Extracting the single instance solution . . . . .	13
1.4 Exact variational method . . . . .	15
1.5 Traveling Salesman Problem . . . . .	17
<b>2 Finding trees in networks</b>	<b>25</b>
2.1 Global vs Local constraints . . . . .	25
2.2 Clustering . . . . .	26
2.3 A Common Framework . . . . .	27
2.4 Single Linkage limit . . . . .	32
2.5 Affinity propagation limit . . . . .	33
2.6 Applications to biological data . . . . .	34
2.6.1 Clustering of protein datasets . . . . .	35
2.6.2 Clustering of verbal autopsy data . . . . .	39
<b>3 Identification of Subgraphs</b>	<b>43</b>
3.1 Graph Alignment Problem . . . . .	43
3.2 The Maximum Clique Problem . . . . .	46
<b>4 Biological Applications</b>	<b>53</b>
4.1 Finding interacting partners . . . . .	54
4.2 Motifs in biological network . . . . .	57
4.2.1 Algorithm . . . . .	60

4.2.2 Discussion . . . . .	64
<b>Conclusions and Perspective</b>	<b>69</b>
<b>List of Publications</b>	<b>71</b>
<b>Bibliography</b>	<b>83</b>

*The ability to reduce everything to simple fundamental laws does not imply the ability to start from those laws and reconstruct the universe. In fact, the more the elementary particle physicists tell us about the nature of the fundamental laws, the less relevance they seem to have to the very real problems of the rest of science, much less to those of society. The constructionist hypothesis breaks down when confronted with the twin difficulties of scale and complexity. The behavior of large and complex aggregates of elementary particles, it turns out, is not to be understood in terms of a simple extrapolation of the properties of a few particles. Instead, at each level of complexity entirely new properties appear and the understanding of the new behaviors requires research which I think is as fundamental in its nature as any other.*

Science, **177**: 393. P.W. Anderson



# Introduction

Biological systems are special classes of systems that encode and process information virtually without errors even if subject to strong thermal noise. One question that can be raised is how this can be explained within the laws of physics. The difficulty to find the answer is deeply related to the complexity of those systems, where several different biomolecules are involved in thousands of biochemical and physical interactions. The usual approach to model these systems, taking only the key ingredients in such complex scenario, still results to be a challenging and astonishingly difficult task. In the thesis we just give a flavor of how statistical physics could be useful to answer question coming from biological problems. We start giving a description of the most interesting experiments that allow to address new quantitative questions in the realm of biology.

## Motivations

Network analysis, inference and optimization represent methodological challenges which play a central role in large scale data analysis. Their practical relevance arises from the huge quantity of empirical noisy data that is being made available in many fields of science, biology and economics in first place.

For example, the recent abundance of genome sequence data has brought an urgent need for systematic analysis to decipher the protein networks that dictate cellular functions.

While stylized dynamical models of gene regulation were formulated as early as in the 1960s [47], the integration of dynamical models with the experimental information about transcription interaction is one of the big modern challenges [21, 87]. Nowadays, by advances in technologies such as mass spectrometry [34], genome-wide chromatin immunoprecipitation [37], yeast two-hybrid assays [94], combinatorial reverse genetic screens [93] and rapid literature mining techniques [81], data on thousands of interactions in humans and most model species have become available. Now the challenge

is to develop new strategies and theoretical frameworks to filter, interpret and organize interaction data into models of cellular functions. If thought from a theoretical physics perspective, this an extremely difficulty task since nor the relevant degrees of freedom neither the microscopic Hamiltonian are generally known for these systems. In this framework theoretical research in biology aims at identifying basic design principles governing the cellular behavior giving the missing ingredients, i.e. the laws of interactions.

It is largely known that apparently different realizations of networks can be identical if looked from a more general perspective namely from a *coarse grained* description. This seems to be true also in biology where interaction networks are the results of random historical choices, subjects to strong selection rules. Despite the difficult task to understand the mechanisms behind the functionality of these networks, what is more interesting, ultimately, is to investigate the principles at the base of historical evolution. In the following we provide a brief description of some of the most popular experiments that allow to probe such tiny systems.

## Experiments

Hereafter, I describe shortly the leading experimental setups which allow to identify protein-protein and protein-DNA interactions within the cell. This resume does not want to be an exhaustive description of the modern experimental state-of-the-art but to give only a flavor of what can be measured nowadays.

Chromatin immunoprecipitation (ChIP) is a powerful tool for identifying transcription factor proteins (TF), associated with specific genomic regions namely specific portion of DNA [88]. The success of the procedure relies on the ability of the antibody to bind to target protein after linked to the DNA. The first step is to wait for the formation of *cross-link* between TF and DNA using some chemical precursor, after which the cell is lysed namely destroyed, freeing the cell extract (the complex of DNA and all the binding proteins). At this point the DNA is shorn in small pieces through a procedure called *sonication* and only after that the DNA/TF complex is precipitated using the target antibody attached to a fixed substrate. The precipitation is the process through which a solid is formed in a solution thanks to a chemical reaction. After treating the complex, finally the DNA portion is purified and identified with some standard techniques. The *in vivo* nature of this method is in contrast with other approaches employed to answer the same questions and, moreover, due to the specificity of the antibody interaction, it permits to select specific TF proteins. The technique can be be used



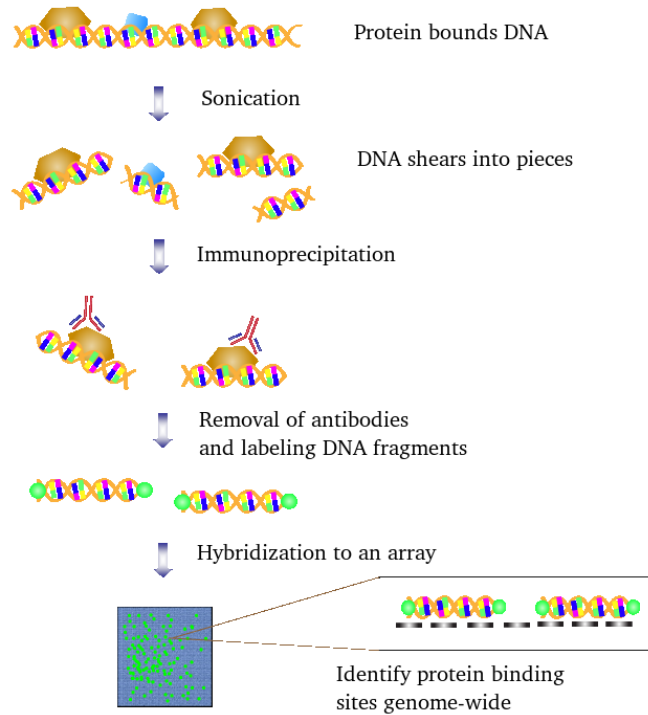


Figure 1: In figure the main steps for experimental setup of genome-wide Chip are reported. The main difference with respect to Chip is principally the last operation where the DNA is prepared to be analyzed using microarrays technique. The idea is to identify DNA regions by exploiting the hydrogen bonds of nucleotide base pairs. The natural bonds formation between A T and C G basis, leads to the hybridization of two complementary DNA nucleic acid sequences and the strength of the interactions depends explicitly on the number of complementary base pairs.

together with microarrays to discover the location of various transcription factors on a genome-wide basis as sketched in figure 1.

Two-Hybrid systems is a method to detect interacting proteins. It is based on the modular organization of many transcription factors TF. Indeed, many such TF are composed by two different functional domains: DNA-binding domain (DBD) and the activator domain (AD). It is of basic importance for the success of the experiment that, when the two parts are separated the TF is not longer able to activate the transcription of a target gene. The first protein used for this technique was the yeast protein GAL4 that is composed on two well defined domains [27]. Using the specificity of the interaction between the two domains, it is possible to test the interaction between two given proteins by looking the activation of the target gene usually called reporter. Now consider two different types hybrid: the first, called bait, contains the DBD fused to a protein of interest whereas the second (prey) is a protein fused to the activator domain. The bait can bind to the DNA, but cannot activate transcription because it does not contain an activation function (if it does, this procedure will not work). Hence, if we express these two hybrid in the same cell, those expressing the reporter gene are identified and purified for further characterization. This can be justified by the fact that interacting proteins contained in the bait and in the prey come close together starting the transcription of the target gene because the two domains are put in contact. This process is sketched in figure 2.

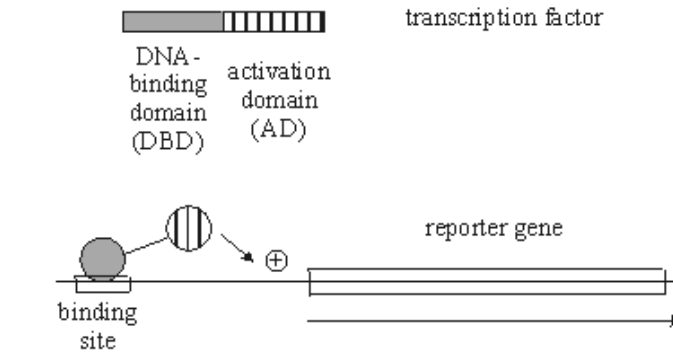
### Plan of the thesis

In the first chapter we give a short introduction to the cavity method and show how it works on the traveling salesman problem. This is a well-known problem, where the method was firstly introduced, which aims at identifying the shortest loop connecting  $N$  different cities. Using a standard argument á la De Gennes, we show how to obtain the optimal tour length from the  $O(m)$ -model, by performing the analytic continuation  $m \rightarrow 0$ . Using this representation, we are in principle able to obtain the minimum length tour by means of the cavity method. However, we experience several problems related to the *global* property of finding a single connected loop visiting all the cities. This seems to be related to the fact that local constraints can not enforce the property of the loop to be composed only of one cycle.

This problem naturally raises the following general question: how is it possible to identify subgraphs of a specific shape into large networks? The answer is intrinsically related to the problem of imposing global constraints by only local ones. Indeed, this can be rephrased as the quest for a clever

## Two-hybrid system

Structure-function properties of a typical **transcription factor**:



**Two-hybrid system:** two types of hybrids:

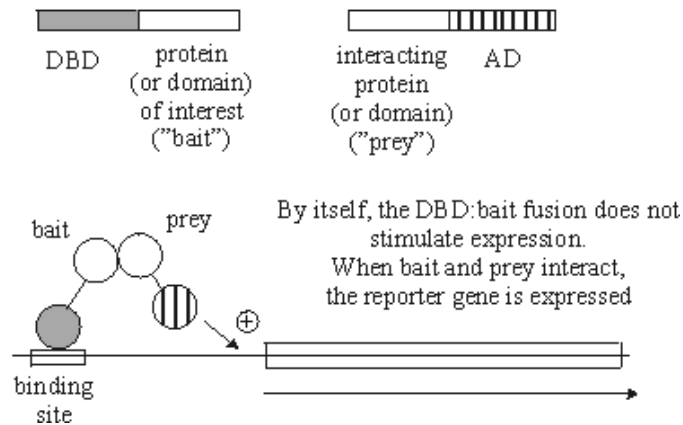


Figure 2: We sketch the description of two-hybrid system. The TF is composed by two different domain: DNA binding and activator domain. When the DBD binds to the cognate binding site in the genome, the activation domain is brought close to the promoter, allowing the activation domain to interact with the transcription machinery and resulting in activation of transcription. If we construct two different hybrid composed by these two domains plus some different proteins, the expression of the reporter is produced only when the two added proteins interact. Indeed if the two proteins in the hybrids do not talk to each other the two domains remain well separated and by itself the bait fusion does not stimulate the expression of the reporter.

representation that allows to scan properties of networks by means of only local checks. This issue represents the main topic of this thesis. We will show a specific example where this goal has been achieved and give a generalization of the cavity method to deal with global constraints where the nature of the problem is explicitly non-local.

In the second chapter, we show how to search for bounded trees of minimum weight in fully connected weighted networks, imposing the global topological property of *being a tree* with local constraints by paying the cost to add a new set of variables. This idea, inspired by some recent works on the Steiner tree problem, has shown an interesting application to efficiently cluster large dataset according on some notion of similarity between data. We show that the algorithm interpolates between two well known clustering methods: Affinity Propagation and Single Linkage. Later on we provide two biological/medical data clustering problems for which external information can be used to validate the algorithmic performance. First, we tackle the problem of clustering homologous proteins based only on their amino acid sequences and finally, we consider a clustering problem arising in the analysis causes-of-death in regions where vital registration systems are not available.

In the third chapter we generalize the method to deal with subgraphs of any given shape, testing it on the maximum clique problem. In this limit the algorithm shows good agreement with theoretical results. The algorithm allows to analyze large graphs and may find applications in fields such as computational biology. In the last chapter we show two different biological applications. Firstly we use it to align the similarity graphs of two interacting protein families involved in bacterial signal transduction, and to predict actually interacting protein partners between these families. Secondly we show how it performs on finding and counting the number of directed different subgraphs in transcriptional regulation networks.

# Chapter 1

## An Overview on the Cavity Method

This chapter is an overview on the cavity method which is the main tool used throughout this thesis. We present the method in a general framework, discuss the hypothesis on which it is based and give its physical interpretation. As an application, we study the traveling salesman problem. This is a long debated hard optimization problem that introduces the main topic of this thesis, namely, the role of global topological constraints.

### 1.1 Optimization and statistical physics

Optimization is a common concept in many research fields from biology to computer science. It typically involves a large number of variables, which are required to simultaneously satisfy a series of constraints. One can equivalently define an energy function as the number of unsatisfied constraints of a given assignment of the variables, and rephrase that problem as the quest for a zero-energy ground state configuration. This analogy with low temperature physics triggered an intensive research effort within the statistical mechanics community[68]. More precisely, a line of approach for these problems consists in looking for typical properties of randomly generated large instances. The introduction of a source of randomness leads naturally to the definition of *ensembles*. This concept summarizes the idea of taking a large number copies of the system, which are under the same constraints and are macroscopically equivalent. Of course they may appear very different at microscopic level, but when looking at self-averaging quantities like energy or entropy, they are identical.

From another point of view, one can be interested in finding explicitly the

single instance assignment of zero energy. This line of research was completely inaccessible for hard problem using approximated methods and can be seen as the main feature of the algorithmic implementation of the cavity method.

As a first step it is interesting to understand how difficult it is to solve the instances. Theoretical computer scientists developed the computational complexity theory in order to quantify how hard problems can be in the worst possible case [76]. The most important and discussed complexity classes are the P, NP and NP-complete. A problem is in the P (polynomial) class when it can be solved by an algorithm (or a deterministic Turing machine) for a given size  $N$  of the input data, in at most  $cN^k$  steps, where  $k$  and  $c$  are independent on the size  $N$ . On the other hand, the NP class is more general and defines the set of all problems whose solution can be checked in polynomial time and, in principle, can be solved by a non-deterministic Turing machine in polynomial time. Although the majority of scientists believe that the two classes are different, it has not been demonstrated that  $\text{NP} \neq \text{P}$ .

The concept of NP-completeness was introduced by Cook for the Boolean satisfiability problem [18]. This complexity class is a subset of NP and it is defined by the following rule: any other NP problem can be converted into one of these NP-complete problems by a polynomial time transformation. Such type of problems turn out to be very difficult and one way of approaching is to use approximation techniques. For a complete list of NP-complete problems see the article of Karp [44]. However, in many everyday problems, the solution is known also for NP-complete problems because the typical instances are much easier with respect to the worst ones. This happens when the difficult cases are very rare while the typical cases are easy to solve, meaning that the time scales polynomially in contrast to their complexity class.

### 1.1.1 Graphical Representation

Inference problems in statistical physics can be reformulated in terms of computing marginal probability on a graphical model. Let  $X_1, \dots, X_N$  be a set of  $N$  discrete-valued random variables and  $x_i$  be a possible outcome of the variable  $X_i$ . We consider the joint probability density function

$$p(X_1 = x_1, \dots, X_N = x_N) = p(\mathbf{x})$$

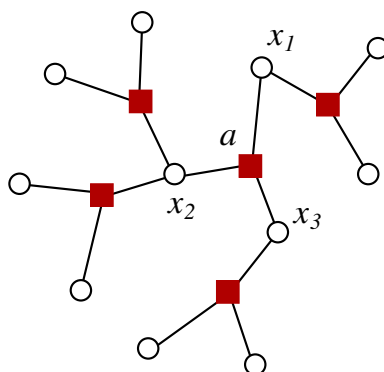


Figure 1.1: We show the factor graph, a bipartite graph composed by two types of nodes: the variable nodes (white circles), associated to each degree of freedom  $x_i$ , and the factor nodes (red squares) associated to function nodes  $f_a$ . This is the graphical representation of the probability defined in equation (1.1.1).

to be written as

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{a=1}^M f_a(\mathbf{x}_a), \quad (1.1.1)$$

where  $Z$  is a normalization constant and  $f_a(\mathbf{x}_a)$  is positive and well defined function of a subset of variables  $\mathbf{x}_a \subset \{x_1, \dots, x_N\}$ . Physically we can interpret the function  $f_a$  as a constraint involving a finite fraction of variables which decreases the available volume of the phase space. In the statistical physics community, these constraints are given by the usual Boltzmann factor

$$f_a(\mathbf{z}) = e^{-\beta e_a(\mathbf{z})} \quad (1.1.2)$$

having properly defined the inverse temperature of the system  $\beta$  and the energy by means of local terms  $e_a(\mathbf{x}_a)$ ,  $E(\mathbf{x}) = \sum_a e_a(\mathbf{x}_a)$ . The relation (1.1.2) authorizes us to identify the normalization  $Z$  with the partition function of the system.

The graphical representation of the probability in equation (1.1.1) is showed explicitly in figure 1.1 and is based on the following two rules. First of all, we associate to each  $x_i$  a variable node (circle), and to each given constraint  $f_a$  a factor node (square). Then, we draw a link between variable node  $i$  and function nodes  $a$ , if  $x_i$  is an argument of the constraint, namely  $x_i \in \mathbf{x}_a$ . Of course, this is a bipartite graph in the sense that every square has only  $k_a$  neighbor circles, where the number  $k_a = |\mathbf{x}_a|$  depends explicitly on the nature of the interactions. Vice-versa, the neighborhood of the variables nodes is composed only by factor nodes (squares) and the number of squares,

$q_i$ , counts how many times the variable node  $i$  interacts. The previous variables satisfy the following natural condition

$$M\bar{k} = \sum_{i=1}^N q_i \quad \text{and} \quad M\bar{k} = \sum_{a=1}^M k_a \quad (1.1.3)$$

where  $\bar{k}$  is the average number of variables involved in the constraint. Let us consider as an example, pairwise interactions in a regular lattice, like in Ising ferromagnet. In this case the factor graph remarkably simplifies because  $q_i$  depends on the dimension  $D$  of the lattice and  $k_a = 2 \forall a$  so the factor nodes (squares) become a redundant way to represent interactions between couples, usually drawn as simple links.

From the joint distribution function we can compute also some important quantities like the most probable configuration  $\mathbf{x}^* : \max_{\mathbf{x}} p(\mathbf{x}) = p(\mathbf{x}^*)$  or the marginals and multi-nodes marginals, defined as a summation over all variables but not the considered ones

$$p_i(x_i) = \sum_{\mathbf{x}/x_i} p(\mathbf{x}) \quad p_a(\mathbf{x}_a) = \sum_{\mathbf{x}/\mathbf{x}_a} p(\mathbf{x}). \quad (1.1.4)$$

Finally, we can also compute the entropy and energy of the system as

$$S = \sum_{\mathbf{x}} p(\mathbf{x}) \log p(\mathbf{x}) \quad E = \sum_{\mathbf{x}} E(\mathbf{x}) p(\mathbf{x}) \quad (1.1.5)$$

In case of complete factor graphs for pairwise interactions, statistical properties are characterized in great detail [77]. In that case the notion of distance in the graph is lost because all variables are neighbors of all the others. The mean field solution turns out to be exact in the renormalization group sense also for model defined on  $d$ -dimensional regular lattice above the upper critical dimension  $d_c$ . Unfortunately, most of the interesting cases in nature are not within this range or are diluted systems without any finite dimensional structure. For this reason, in last years new methods to deal with sparse graphs i.e. Erdős-Rényi random graphs, random regular graphs, or configurational models, have been developed. Among those is the cavity method.

## 1.2 Cavity Method and Belief Propagation

The Cavity Method is a tool introduced in the field of disordered diluted spin systems by Parisi and Mézard in [61] and further developed in [63] to deal with non-trivial correlations. In principle it allows to compute the



marginal probability of the Boltzmann measure for local degrees of freedom in an acyclic graphical model. It has been successfully applied in many combinatorial optimization and condensed matter problems on sparse graphs [71, 53] and could be seen as complementary to the mean field method. The latter case is valid in fully connected graphs while the former is exact for diluted graphs where a notion of distance and the concept of *locality* is somehow recovered. The cavity method offers new tools to study the physics of diluted systems.

The natural algorithmic implementation of this method is called Belief Propagation (BP), which is well known in the computer science community since the 60's [83] and further developed to study inference problems by Pearl in [79]. BP provides an efficient scheme to solve by iteration a set of mean field equations in sparse graph and can be seen as a method to organize the calculation based on distributed simple elements that operate in parallel.

It is worth mentioning that this method has been introduced several times. For what concerns the physical framework, Bethe [10] firstly introduced the free energy functional in terms of marginals to compute the partition function of the Ising model. Its generalization to inhomogeneous systems is principally due to Thouless, Anderson and Palmer in their seminal work on the spin glass problem [92]. Nevertheless, the equivalence of the two forms, BP and Bethe approximation for inhomogeneous systems in loopy graphs, was demonstrated only recently [99, 100]. In these articles the authors introduced the derivation of the BP equations inspired by a variational technique which shows that the method is a reliable approximation.

From a strictly mathematical point of view, both the cavity method and BP are poorly understood, although a number of steps further have been made in the last decade, showing its validity in locally tree graphs [69].

The idea behind the cavity method is to compare the properties of the system after having created a *cavity*, namely having removed one node  $i$  from the system and, consequently, the variable node  $i$  and all its nearest neighbors factor nodes  $a \in \partial(i)$  from the associated factor graph. We define  $\partial(i)$  as the set of neighbor factor nodes of  $i$ . This assumption naturally leads to a set of self consistent equations for the marginal probability (1.1.4) of the removed node  $i$  as a function of the probability in its absence  $p_i(x_i) = \mathcal{F}(p_{j \rightarrow a}(x_j))$  as represented in figure 1.2. Here  $p_{j \rightarrow a}$  is a notation to define the probability density function of node  $j$  removing factor node  $a \in \partial(i)$  and  $\mathcal{F}$  is a generic functional form. In general, this set of equations involves the solution of the equivalent model with  $N - 1$  nodes. Despite this difficulty, assuming the absence of correlations among variables after

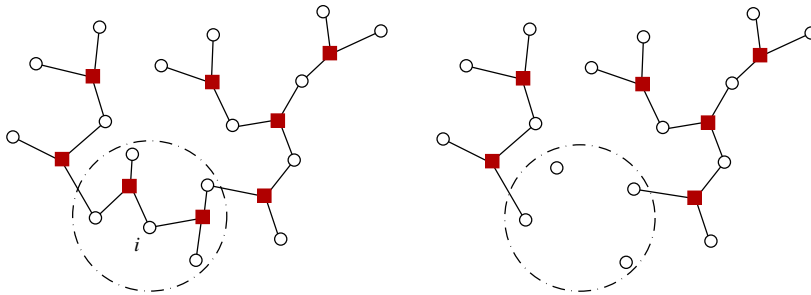


Figure 1.2: The system with  $N$  nodes is compared to the system after removing the node  $i$ . This leads to a set of self consistent equations that, in the absence of correlations among neighbors of the node  $i$  turn out to be exact, like for example in tree or in fully connected graphs. The basic hypothesis behind the method at this level is the presence of a single pure state or in spin glass terms the replica symmetry is not broken.

having created the cavity, we can obtain a set of self consistent equations depending only on the cavity probabilities. This assumption is completely justified when the beneath graph is a tree. In that particular case, after having removed one node, the graph splits up into two or more disconnected components and therefore the joint probability density function factorizes. A word of caution has to be added at this point. In physical systems with short range interactions, correlations usually decay on the scale of few microscopic units of length. However, close to critical points, where spontaneous symmetry breaking occurs the above assumption of decay of correlations generally fails since the system develops long range order. If this is the case, we need to select one single pure state to ensure the absence of correlation<sup>1</sup>. This can be very easy to do, such as in the ferromagnetic/paramagnetic Ising critical point or much less trivial as in spin glass systems, where in the low temperature phase replica symmetry breaking occurs. We will not address here this very interesting topic and refer the reader to the original literature where the proper generalization of the cavity method has been introduced to deal with non trivial correlations[64, 50].

Let us start by describing the method. Firstly we provide an algorithmic scheme which could be also used for computational calculations in order to

<sup>1</sup>A very important feature of the pure state is the clustering property. In essence it states that connected correlation between two different points goes to zero when their distance goes to infinity. Take for example the paramagnetic phase below the critical temperature, we have that  $\langle \sigma_i \sigma_j \rangle_c = m^2$ , meaning that this state is not a pure state when  $T < T_c$ . But after projecting the system by means of an external field, the correlation vanishes and the cavity method still holds.

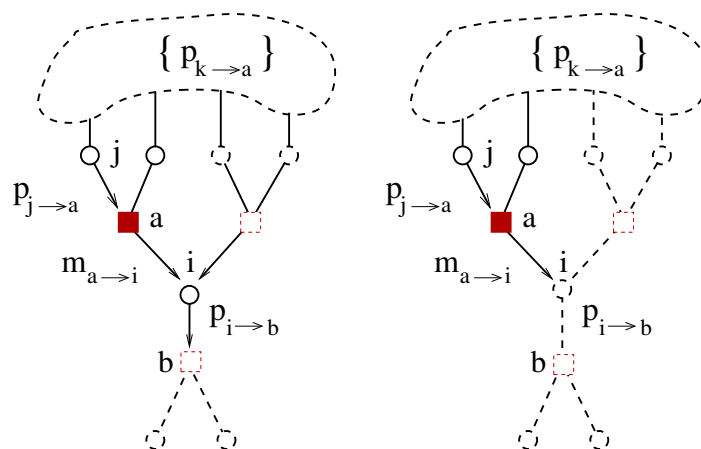


Figure 1.3: After having created a cavity in the graph, we remove a variable node and all its neighbor factor nodes. Let us define messages from factor nodes to variable nodes as a vector over the possible states of  $x_i$ , and viceversa cavity probabilities from variable nodes to factor nodes as the probability of the variable  $x_i$  in absence of the constraint  $a$ .

find a single instance solution. Then, we generalize the results in the whole graph ensemble, to obtain a well defined description of the method in the thermodynamic limit.

### 1.2.1 The symmetric solution on a single graph

Let us first introduce *messages* for each link of the factor graph. We can distinguish between two type of vector messages. The one going from the variable node  $i$  to the factor node  $a$   $p_{i \rightarrow a}(x_i)$  and vice-versa let  $m_{a \rightarrow i}(x_i)$  be the vector message over the possible states of  $x_i$  from the factor nodes  $a$  to the variable  $i$ . The former term is the cavity marginal probability in absence of the constraint  $f_a(\mathbf{x}_a)$  whereas the latter can be physically interpreted as a statement from the factor node  $a$  to the variable  $i$  about the relative probabilities of  $i$  to be in one of its possible state  $x_i$  related to the constraint  $f_a(\mathbf{x}_a)$ , as sketched in the cartoon 1.3. The structure of messages is determined by the underlying factor graph and they satisfy the

following equations

$$p_{i \rightarrow a}(x_i) = \frac{1}{Z^{i \rightarrow a}} \prod_{c \in \partial(i)/a} m_{c \rightarrow i}(x_i) = \mathcal{F}_p(\{m_{c \rightarrow i}\}) \quad (1.2.1)$$

$$m_{a \rightarrow i}(x_i) = \frac{1}{Z^{a \rightarrow i}} \sum_{\mathbf{x}_a/x_i} f_a(\mathbf{x}_a) \prod_{j \in \partial(a)/i} p_{j \rightarrow a}(x_j) = \mathcal{F}_m(\{p_{j \rightarrow a}\}), \quad (1.2.2)$$

where  $\partial(i)/a$  indicates the set of factor nodes close to the variable node  $i$  except  $a$  and  $\partial(a)/i$  denotes all the variable nodes which are nearest neighbors of the factor node  $a$  but  $i$ . Notice that, in presence of correlations, the true condition for the messages would be  $m_{a \rightarrow i}(x_i) \propto \sum_{\mathbf{x}_a/x_i} f_a(\mathbf{x}_a) P_{a \rightarrow i}(\mathbf{x}_a/i)$  where  $P_{a \rightarrow i}(\mathbf{x}_a/i)$  is the joint probability density function of the variables  $x_j \in \{\mathbf{x}_a\} - x_i$  in absence of node  $i$ . Of course this term is non trivial and can not be expressed in terms of messages but, within the hypothesis of independence, the joint probability factorizes as a product of terms  $P_{a \rightarrow i}(\mathbf{x}_a/i) = \prod_{j/i} p_{j \rightarrow a}(x_j)$  leading to (1.2.2).

We can replace the equation for the  $p_{i \rightarrow a}$  (1.2.1) into the equation (1.2.2) and obtain a set of self consistent equations, to be solved by iteration. Therefore, we introduce the time dependent messages  $m_{a \rightarrow i}^t(x_i)$  and the cavity probabilities  $p_{j \rightarrow a}^t(x_j)$  and define at each time step an updating scheme by using the previous equations, to obtain

$$p_{i \rightarrow a}^t(x_i) = \frac{1}{Z^{i \rightarrow a}} \prod_{c \in \partial(i)/a} m_{c \rightarrow i}^t(x_i) \quad (1.2.3)$$

$$m_{a \rightarrow i}^t(x_i) = \frac{1}{Z^{a \rightarrow i}} \sum_{\mathbf{x}_a/x_i} f_a(\mathbf{x}_a) \prod_{j \in N(a)/i} p_{j \rightarrow a}^{t-1}(x_j). \quad (1.2.4)$$

These equations are called the Belief Propagation equations or the *sum-product* equations because they involve a summation of a product of terms whose fixed points are the solution of the equations (1.2.1) and (1.2.2).

The messages are usually initialized to 1 but other non-negative initializations, that keep the messages positive-definite, are also possible. If the BP equations converge, we obtain the fixed point messages and in term of these we can define the BP marginals or beliefs  $b_i(x_i)$ . The beliefs are defined as

$$b_i(x_i) \propto \prod_{a \in \partial(i)} m_{a \rightarrow i}(x_i) \quad (1.2.5)$$

and, more generally, we can define the *multi-nodes* beliefs which involve all the variable of the constraint  $a$

$$b_a(\mathbf{x}_a) \propto f_a(\mathbf{x}_a) \prod_{i \in N(a)} \prod_{c \in \partial(i)/a} m_{c \rightarrow i}(x_i). \quad (1.2.6)$$

Naturally the BP equations (1.2.1) (1.2.2) could be obtained from the marginalization condition  $b_i(x_i) = \sum_{\mathbf{x}_a/x_i} b_a(\mathbf{x}_a)$ , starting from the definition (1.2.6). In term of these marginals, it is possible to construct a joint probability density of this form

$$p(x) = \frac{\prod_a b_a(\mathbf{x}_a)}{\prod_i b_i(x_i)^{q_i-1}} \quad (1.2.7)$$

Despite, it is not possible to show that it is correctly normalized on a general ground, it is true in graph with a local tree structure. We will see later on that marginalization property together with the normalization of the belief, comes out as a natural consequence from a variational method [99, 100] although the lack of normalization for the joint probability density function prevents the Bethe free energy to be an upper bound to the real free energy.

### Free energy

Physical quantities like energy or entropy can be computed on the basis of the beliefs, given the identities in (1.1.2)

$$\begin{aligned} E_{BP} &= -\frac{1}{\beta} \sum_a \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) \ln f_a(\mathbf{x}_a) = \sum_a \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) e_a(\mathbf{x}_a) \\ S_{BP} &= -\sum_a \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) \ln b_a(\mathbf{x}_a) + \sum_i (q_i - 1) \sum_{x_i} b_i(x_i) \ln b_i(x_i) \end{aligned} \quad (1.2.8)$$

Therefore the free energy can be written as usual

$$F_{BP} = E_{BP} - \frac{1}{\beta} S_{BP}. \quad (1.2.9)$$

At this point we can use a more intuitive approach in computing the free energy given the cavity probabilities and messages that fulfill (1.2.1) and (1.2.2). This method can elucidate the factorized form of the Bethe probability grasping intuition from a more physical framework.

Let us start saying that when we create a cavity in the graph, namely we remove a variable node from the hypergraph, the free energy of the system is decreased by a factor due to the deletion of the node and of all its clauses. We can think to the dual case of removing a function node  $a$  and consequently all the attached nearest variable nodes, where  $\Delta F^{a+i \in \partial(a)}$  is the shift in free energy associated to the deletion of  $a$  and  $i \in \partial(a)$  while  $\Delta F^i$  is that related to the elimination of  $i$ . This method has been introduced in [62] by taking advantage of the physical intuition that the perturbation associated

to these variations remains localized in the graph and does not propagate. The  $\Delta F^{a+i \in \partial(a)}$  can be written in term of cavity messages  $m_{a \rightarrow i}(x_i)$  as

$$e^{-\beta \Delta F^{a+i \in \partial(a)}} = \sum_{\mathbf{x}_a} f_a(\mathbf{x}_a) \prod_{i \in \partial(a)} p_{i \rightarrow a}(x_i) \quad (1.2.10)$$

while the contribution obtained adding one node reads

$$e^{-\beta \Delta F^i} = \sum_{x_i} \prod_{a \in \partial(i)} m_{a \rightarrow i}(x_i). \quad (1.2.11)$$

The total free energy can thus be written as the summation over all the constraints of the increasing amount of free energy associated to the function node  $a$  and its nearest neighbors, minus the terms counted twice. So that

$$F_{BP} = \sum_a \Delta F^{a+i \in \partial(a)} - \sum_i (q_i - 1) \Delta F^i. \quad (1.2.12)$$

An intuitive interpretation of this result is that in order to go from 0 to  $M$  constraints one should add the energy of all the constraints. But in this sum each variable  $i$  appears in  $q_i$  different clauses and we have to subtract explicitly  $q_i - 1$  energetic terms in order to remove the degeneracy. It is possible to show explicitly that the forms (1.2.12) and (1.2.9) are equivalent by using the following relations

$$\begin{aligned} \sum_{\mathbf{x}, a} b_a(\mathbf{x}) \ln b_a(\mathbf{x}) &= \sum_i (q_i - 1) \sum_{a \in \partial(i), x} b_i(x) \ln m_{a \rightarrow i}(x) - \beta \sum_{\mathbf{x}, a} b_a(\mathbf{x}) e_a(\mathbf{x}) \\ &\quad - \sum_a \ln \left[ \sum_{\mathbf{x}} e^{-\beta e_a(\mathbf{x})} \prod_{i \in \partial(a)} \prod_{b \in \partial(i)/a} m_{b \rightarrow i}(x_i) \right] \end{aligned} \quad (1.2.13)$$

$$\sum_{x, i} b_i(x) \ln b_i(x) = \sum_{i, a \in \partial(i), x} b_i(x) \ln m_{a \rightarrow i}(x) - \sum_i \ln \left[ \sum_x \prod_{a \in \partial(i)} m_{a \rightarrow i}(x) \right] \quad (1.2.14)$$

Substituting the definition of entropy and energy given in (1.2.8) in the free energy (1.2.9), the previous conditions for the marginals allows for the following consideration: the energetic term  $E_{BP}$  is annihilated by the second term in the right hand side of equation (1.2.13). Then we can recognize that the first term in same equation (1.2.13) cancel exactly the first of (1.2.14) after multiplied by  $q_i - 1$ . The remaining terms are easily identified with the  $\Delta F^i$  and  $\Delta F^{a+i \in \partial(a)}$  factors defined respectively in (1.2.10) and (1.2.11). This calculation ends the demonstration that the two ways of computing the free energy, one from the definition of joint probability distribution function and the other from physical consideration about the local tree geometry, are equivalent and lead to the same value for the free energy.

### 1.2.2 Average over the graph ensemble

We now turn to the case of random ensembles. Let us introduce a source of randomness in the realization of the interaction networks and move the attention from a single factor graph to the whole set of different factor graphs constructed following several rules. This assumption is justified a posteriori by the fact that in real networks, a large number of different realizations seems to perform the same action or function and, in a certain sense, belongs to the same class. The source of randomness in graphs could be in the choice of the variable nodes that interact together or in the number of times a single variable enters in a constraint.

In this probabilistic description, some extensive quantities are interesting, i.e. the total number of edges, the node degree sequence or the number of connected components in which the graphs split.

Let us introduce the *random factor graph ensembles* which can be seen as a generalization to graph theory of the usual thermodynamic ensembles [22].

#### Classical random graphs and random factor graphs

The interest in random graphs mainly concerns on the uncorrelated graph ensembles, where the architecture is completely determined by the degree distribution  $\mathcal{S}(q)$ . The two simplest models for random graphs are the *Gilbert* graph ensemble  $G_{Np}$  [32] and the *Erdős-Rényi* ensemble  $\mathcal{G}_{NM}$  [26]. The  $\mathcal{G}_{Np}$  is the set of all graphs with probability  $p$  of having an edge between each pair of the  $N$  nodes. The microcanonical version of this ensemble, or the  $G_{NM}$ , is defined as the set of graphs with  $N$  nodes and fixed number of edges  $M$ . In the former case the number of edges fluctuates from different realizations and is a Poissonian random variable with mean value  $M = p \times N(N - 1)/2$ . Naturally, it follows that in the thermodynamic limit this two different ensembles share the same statistical properties when the external parameter  $p$  and  $M$  satisfy the relation  $p = 2M/(N(N - 1))$ . The properties of the Gilbert or Erdős-Rényi graphs strongly depend on how the parameter  $p$  and respectively  $M$  scales with  $N$ . Of course, for fully connected graph we have  $p = 1$  or equivalently  $M = N(N - 1)/2$  and for null graph  $p = 0$  and  $M = 0$ . Moreover, introducing the density of links  $\alpha = M/N$ , the graph is sparse if  $\alpha/N \rightarrow 0$  for large  $N$ . This means that the parameter  $p \sim o(1/N)$  and consequently  $M = o(N)$ . In *Gilbert* cases, where the mean degree  $\langle q \rangle = c$  is fixed, we have that the degree  $q$  for each node has a Poissonian distribution  $\mathcal{S}(q) = e^{-c}c^q/q!$ .

These Poissonian ensembles can be generalized to the factor graphs de-

noting by  $\mathcal{R}(k)$  the degree distribution of the factor nodes, where the density of constraints can be written as

$$\alpha = \frac{M}{N} = \frac{c}{\langle k \rangle}. \quad (1.2.15)$$

If we fix the degree of factor node  $k = K$  and the number of variable to  $N$  we have  $\binom{N}{K}$  possible choices for a given link in a constraint. We define the equivalent of Gilbert ensemble for hypergraphs assuming that each link has equal probability to be selected and is

$$p = \frac{cN}{K \binom{N}{K}}$$

so that the total number of constraints is a random variable with average value  $M = cN/K$ , recovering the *Gilbert* random ensemble for  $K = 2$ .

The sparse regime  $c = O(1/N)$  is particularly interesting for our purpose, since it follows that graphs are locally tree with only infinitely large loops. This can be proved by computing the average length of the shortest cycle going through variable  $i$ . Let us consider a particle which diffuses on this graph. The probability for it to come back to the starting point  $i$  after  $d$  steps, without coming back from the same edges, reads

$$1 - \left(1 - \frac{1}{N}\right)^{\sum_{j=1}^d (\gamma_q \gamma_k)^j}, \quad (1.2.16)$$

where  $\gamma_q = \frac{\langle q^2 \rangle - \langle q \rangle}{\langle q \rangle}$  and  $\gamma_k = \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}$ . The probability of coming back in a number of steps  $d \ll \log N / \log(\langle q^2 \rangle - \langle q \rangle / \langle q \rangle)$  vanishes with  $N$ , meaning that cycles smaller than  $\log N$  become very rare in large size limit.

### 1.2.3 Belief propagation equations over the ensemble

We study the properties of the typical instances of the random ensemble by denoting with  $\langle \cdot \rangle$  the average over the realization of graphs. We assume that the factor graph ensemble is described by a given degree distribution  $\mathcal{S}(q)$  for variable nodes and  $\mathcal{R}(k)$  for constraints. If we introduce the distribution of the messages  $m$ 's,  $\mathcal{P}(m)$ , and of the cavity probability  $p$ 's,  $\mathcal{Q}(p)$ , we have

$$\begin{aligned} \mathcal{P}(m) &= \sum_{q=1}^{\infty} \frac{q \mathcal{S}(q)}{\langle q \rangle} \int \prod_{i=1}^q [dp^i \mathcal{Q}(p^i)] \delta(m - \mathcal{F}_m(\{p_i\})) \\ \mathcal{Q}(p) &= \sum_{k=1}^{\infty} \frac{k \mathcal{R}(k)}{\langle k \rangle} \int \prod_{i=1}^k [dm^i \mathcal{P}(m^i)] \delta(p - \mathcal{F}_p(\{m_i\})) \end{aligned} \quad (1.2.17)$$



where  $\mathcal{F}_p$  and  $\mathcal{F}_m$  are the functional form defined in equations (1.2.1) and (1.2.2). Once solved the previous equations, we have the explicit form of the distributions of messages and cavity probabilities. This allows to compute all the interesting quantities like free energy and entropy, by averaging the equations 1.2.8 over  $\mathcal{P}$ .

The solution of those equations is not straightforward and usually it is not solvable analytically, however a very efficient numerical technique called population dynamics [63] has been developed. The basic idea of the method is to represent the distributions in term of a population, or a sample of  $N_{pop}$  elements drawn at random from it.

Since at the beginning we do not know the distribution, therefore we start by sampling the population with uniform probability and then update their values according to the BP equations. We just take randomly one term in the population and compute its value using the equations (1.2.1)-(1.2.2) in terms of the other randomly chosen elements. After  $T$  updating iterations we arrive, up to an arbitrary precision, to the desired fixed point. The probability distribution is then obtained by computing the histogram of the  $N_{pop}$  terms. Of course, this method is approximated but provides reliable results and the error can be estimated in function of  $N_{pop}$  and the time  $T$ .

### 1.3 Extracting the single instance solution

In general, the BP equations show some difficulties in the convergence and usually to extract a single solution from local marginals is not an easy task. For the latter problem there are some *decimation* technique<sup>2</sup> [64, 65, 78, 70] devoted to the search of a solution starting from the BP beliefs. The *decimation* scheme proceeds as follows:

- Run the BP algorithm.
- Use the marginals to choose a variable  $i$  and identify its value  $x_i^*$ .
- Fix the value of  $x_i = x_i^*$  and run again BP algorithm with the constraints  $p_i(x_i) = \delta(x_i - x_i^*)$  or adding an external field  $h_i = \infty$  directed along the  $x_i^*$ .

If the algorithm does not produce a contradiction, it stops only when all the variables are fixed, thus providing a solution of the problem. The performance of the method are deeply related to the selection rule of the second

---

<sup>2</sup>These techniques have been developed in the context of Survey Propagation and only successively extended to BP.

point and of course we have to run  $N$  times the BP algorithm to fix all the variables.

In what follow we are going to illustrate a different procedure, called *reinforcement*, a sort of smooth decimation introduced to solve both the problems of convergence and selection of optimal assignments [16, 15].

### Reinforcement

The idea behind the reinforcement technique is to smoothly decimate during the iteration procedure. This allows to run the BP algorithm only one time, improving the convergence and finding the optimal assignment of the variables at once. The idea is to introduce an extra term into the equation (1.2.3), which can be defined as an external field and precisely

$$p_{i \rightarrow a}^{t+1}(x_i) = \frac{[b_i^t(x_i)]^{\gamma_t}}{Z^{i \rightarrow a}} \prod_{c \in \partial(i)/a} m_{c \rightarrow i}(x_i) \quad (1.3.1)$$

where  $\gamma_t = 1 - \gamma_0^t$  and  $\gamma_0 \in [0, 1]$  such that  $\gamma_t \rightarrow 1$  with increasing time. The two trivial choices,  $\gamma_0 = 1/0$ , correspond respectively to obtain back the BP equations when  $\gamma_0 = 1$  and for  $\gamma_0 = 0$  to add an external field  $h_i = \log b_i^t(x_i)$  that depends on the last computed belief  $b_i^t(x_i) = \prod_a m_{a \rightarrow i}^t(x_i)$ . If the belief equals the exact solution  $b_i^t(x_i) = p_i(x_i) = \delta(x_i - x_i^*)$ , the right choice would be  $\gamma_0 = 0$ , since we find instantaneously the solution by polarizing the cavity field in the right direction. Of course, at  $t = 0$ , the beliefs are randomly initialized so in general it is misleading to select  $\gamma_0 = 0$ . The choice of this parameter is crucial, since it modules the weight of the computed beliefs by introducing a non trivial factor in front of the external field  $h_i = \gamma_t \log b_i^t(x_i)$ . In order to understand better how to tune this parameter, it is worth stressing the fact that it determines the threshold time  $T^{th}$  at which  $\gamma_t$  reaches 1, giving the maximum weight of the previously computed local belief, which scales as  $T^{th} \sim -\frac{\alpha}{\log \gamma_0}$ .

We numerically find that after having introduced a reinforcement term, the dynamics of the equation oscillates a lot. This is showed by the error, namely the maximum difference between two successive iterations, which decreases monotonously and then changes by starting to increase again. The amplitude of oscillations mostly seems to reduce, restoring the convergence after a time which depends on the choice of  $\gamma_0$ . However we will enter into the details of the results in the rest of the thesis for the specific problem under consideration.

## 1.4 Exact variational method

It is possible to show that the BP equations are valid in all graphs which are locally trees[69]. This has been finally demonstrated but it was already expected since loops are of order  $\log N$  when correlations decay fast enough [56]. Moreover, this method provides good estimate of the free energy also in case when graphs have short loops.

In fact, it is possible to show that the fixed points solutions of the BP equations are the stationary points of a variational free energy. The physical intuition dates back to the work of Bethe [10] and successively by Kikuchi [48]. Variational method are based on the following assumption: ignoring the true distribution of the system  $p(\mathbf{x})$ , a trial probability  $b(\mathbf{x})$  is defined which is correctly normalized and positive definite, so that it is possible to define a variational free energy

$$F(b) = E(b) - TS(b) \quad (1.4.1)$$

which differs from the real free energy by a factor  $D(b||p)$  called the Kullback-Leibler divergence

$$D(b||p) = \sum_{\mathbf{x}} b(\mathbf{x}) \log \frac{b(\mathbf{x})}{p(\mathbf{x})}. \quad (1.4.2)$$

It can be proved that this term is always non-negative and is zero only when the two distributions coincide [20], thus providing an upper bound to the exact free energy. Minimizing  $F(b)$  with respect to the trial probability is a legitimate procedure to find the best approximation of the real free energy.

This is the basic idea behind the mean field method where the trial probability is factorized as a product of marginal probabilities and in this spirit the Bethe approximation is going a step further, by considering more complicated forms for the joint probability  $b(\mathbf{x})$  which involves the multi-nodes beliefs  $b_a(\mathbf{x}_a)$ .

Let us now introduce the trial probability in terms of the marginals

$$b(x_1, \dots, x_N) = \frac{\prod_a b_a(\mathbf{x}_a)}{\prod_i b_i(x_i)^{q_i-1}} \quad (1.4.3)$$

where the marginals have to be *locally consistent*. In other words they have to fulfill the normalization conditions

$$\sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) = 1 \quad \sum_{x_i} b_i(x_i) = 1 \quad \sum_{\mathbf{x}_a/x_i} b_a(\mathbf{x}_a) = b_i(x_i) \quad (1.4.4)$$

such that the associated free energy reads

$$F(b) = - \sum_a b_a(\mathbf{x}_a) \log \frac{b_a(\mathbf{x}_a)}{f_a(\mathbf{x}_a)} + \sum_i (q_i - 1) b_i(x_i) \log b_i(x_i). \quad (1.4.5)$$

Unfortunately, it is not generally possible to construct the trial joint distribution consistent with the marginals beliefs  $b_a(\mathbf{x}_a)$  and  $b_i(x_i)$ . Therefore the free energy defined in (1.4.5) is not rigorously an upper bound of the free energy  $F$ , but still it is an accurate approximation giving good results. The analogy with variational methods suggests that the stationary points of the Bethe free energy should play an important role. In fact, the fixed point of BP equations (1.2.1) and (1.2.2) are in one-to-one correspondence to the stationary points of  $F(b)$  as we will show in the following in the context of Lagrangian theory. Nevertheless, the reason why the BP equations give good results in general is still unclear and a matter of intensive research.

We introduce a set of Lagrange multipliers:  $\lambda_i$  corresponds the normalization of the belief  $b_i(\cdot)$  and  $\lambda_{ai}(x_i)$  is associated to normalization of  $b_a(\cdot)$ . We have totally introduced  $N$  plus  $N \times M$  parameters which are used in the definition of the Lagrangian

$$\mathcal{L} = F(b) - \sum_i \lambda_i \left[ \sum_{x_i} b_i(x_i) - 1 \right] - \sum_{a,i \in N(a), x_i} \lambda_{ai}(x_i) \left[ \sum_{\mathbf{x}_a/x_i} b_a(\mathbf{x}_a) - b_i(x_i) \right]. \quad (1.4.6)$$

Imposing the stationary conditions according to the beliefs, we obtain that

$$b_i(x_i) = e^{1 - \frac{1}{q_i - 1} (\sum_a \lambda_{ai}(x_i) - \lambda_i)} \quad b_a(\mathbf{x}_a) = f_a(\mathbf{x}_a) e^{-\sum_{i \in N(a)} \lambda_{ai}(x_i) - 1}. \quad (1.4.7)$$

The multipliers have to be fixed by imposing the normalization conditions: the first two equations, reported in (1.4.4), fix the values of  $\lambda_{ai}$  and  $\lambda_i$ , whereas the last  $\sum_{\mathbf{x}_a/x_i} b_a(\mathbf{x}_a) = b_i(x_i)$  gives rise to a set of self consistent conditions for the Lagrangian parameters.

In order to see the correspondence with the BP equation (1.2.1) (1.2.2) we introduce the following definitions

$$p_{i \rightarrow a}(x_i) \propto e^{-\lambda_{ai}(x_i)}, \quad m_{a \rightarrow i} \propto \sum_{\mathbf{x}_a/x_i} f_a(\mathbf{x}_a) e^{-\sum_{j \in N(a)/i} \lambda_{aj}(x_j)}. \quad (1.4.8)$$

It is easy to show that  $m_{a \rightarrow i}$  fulfills the condition

$$m_{a \rightarrow i}(x_i) = \frac{1}{Z_{a \rightarrow i}} \sum_{\mathbf{x}_a/x_i} f_a(\mathbf{x}_a) \prod_{j \in N(a)/i} p_{j \rightarrow a}(x_j). \quad (1.4.9)$$

Performing some manipulations, reported in detail in [71], we can also identify the  $p_{i \rightarrow a}$  as the cavity probability

$$p_{i \rightarrow a}(x_i) = \prod_{b \in \partial(i)/a} m_{a \rightarrow i}(x_i). \quad (1.4.10)$$

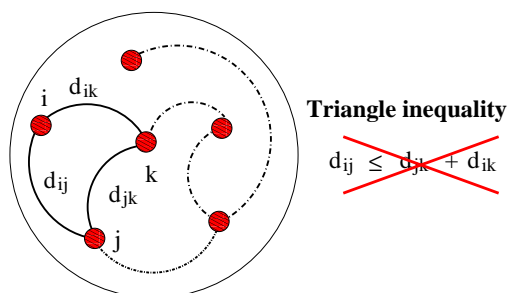


Figure 1.4: The problem of finding the shortest tour among  $N$  cities can be studied into different ensembles. The first one is called Euclidean and the cities are putting randomly in a real  $D$  dimensional space with euclidean distances. On the contrary, we can think to select randomly the distances between the cities from a power law distribution  $\rho(d)$  defined in (1.5.14) to mimic spatial correlations. In that case, the use of the term distance is an abuse because they are not real distances since they do not satisfy the triangle inequality.

Viceversa, for any given solution of the BP equations one is able to define a set of Lagrangian parameters using the relation (1.4.8). This ends the proof of the relation between the stationary points of the Bethe free energy and the solutions of the BP equations.

This correspondence has important consequences for the existence and unicity of the BP fixed points. For example in acyclic graphs the convexity of the free energy is useful to show that the BP fixed point is unique whereas in loopy-structures there could be multiple stationary points[71].

## 1.5 Traveling Salesman Problem

In the following we show how the general formalism described in this chapter applies to solve the problem of identifying the shortest tour among a set of  $N$  cities the so called traveling salesman problem (TSP). The TSP has interested the scientific community since the last twenty years mainly because it is an hard combinatorial optimization problem that can be mapped onto the ground state of a statistical mechanical model[59, 49, 58, 60, 74].

Let us introduce the problem: we have a set of  $N$  cities connected two by two by independent, identically distributed, random distances,  $d_{ik}$ , which can be interpreted as the costs for traveling from a city  $i$  to another  $k$  and do not satisfy the triangle inequality (see figure 1.4). The set of distances defines a  $N \times N$  matrix that is random, symmetric and positive definite. The TSP is focused on the research of the shortest, among all possible tours, passing

through all the cities. Of course, doing it iteratively, requires a time which increases exponentially with the system size  $N$ .

It is straightforward to show that the shortest tour is in the class of closed paths, defined as the closed walks or loops passing through all the cities only once. Hence that it can be identified with the  $N - 1$  links connecting the cities  $\Gamma^* = \{e_1, e_2, \dots, e_{N-1}\}$  and its length is  $E = \sum_{e \in \Gamma^*} w_e$ , where  $w_e$  is the weight associated to the  $e$  link. We can define a partition function as the summation over all the possible  $N$ -city loops,  $\Gamma$ , following the Boltzmann prescription as

$$Z(\beta) = \sum_{\Gamma} e^{-\beta E(\Gamma)} = \sum_{n_{ij}=1,0} e^{-\beta \sum_{i<j} n_{ij} d_{ij}} \quad (1.5.1)$$

where  $\beta$  is a fictitious temperature and  $E = \sum_{i<j} n_{ij} d_{ij}$ . Here the  $n_{ij}$  is a 0 – 1 variable which can be interpreted as the probability that the link  $\langle ij \rangle$  is included in the  $N$ -city loop. The average energy and average occupation number is related to the partition function

$$\langle E \rangle = -\frac{\partial \ln Z}{\partial \beta} \quad \langle n_{ij} \rangle = -\frac{\partial \ln Z}{\partial d_{ij}} \quad (1.5.2)$$

and in the limit  $\beta \rightarrow \infty$  the minimum length tour is obtained. In order to proceed it is convenient to formulate the TSP in terms of an  $m$ -component spin model in the  $m \rightarrow 0$  limit [31]. This model is defined on a fully connected graph of  $N$  vertices with exponentially decaying two-bodies interactions  $u_{ij} = e^{-\beta d_{ij}}$ . For each vertex  $k$  we assign an  $m$ -component spin vector  $\mathbf{S}_k$  satisfying the normalization condition  $\mathbf{S}_i \cdot \mathbf{S}_i = m$ . Thus the partition function reads

$$G(\beta, m, \omega) = \int \prod_{q=1}^N d\mu(\mathbf{S}_q) \exp \left( \omega \sum_{i<j} u_{ij} \mathbf{S}_i \cdot \mathbf{S}_j \right) \quad (1.5.3)$$

where  $d\mu$  is the measure over the  $m$ -dimensional sphere normalized to one and  $\omega$  is an external parameter. It is possible to show by a direct reasoning that the TSP is mapped exactly into the  $O(m)$  spin model taking the proper limits:

$$\lim_{\substack{m \rightarrow 0 \\ \omega \rightarrow \infty}} \frac{G - 1}{m\omega^N} \equiv \sum_{\Gamma} \exp \left( -\beta \sum_{i<j} n_{ij} d_{ij} \right). \quad (1.5.4)$$

This equality can be obtained by using a classical diagrammatic argument and expanding the partition function in terms of  $\omega$ . Therefore, noticing that only closed diagrams with single loop correspond to non vanishing terms in

the sum we can identify the partition function of the TSP with that of the  $m$ -component spin model. An other important consequence of the analytic continuation  $m \rightarrow 0$  is the fact that only diagrams up to order  $\omega^N$  will survive since any closed diagram with more than  $N$  links must necessarily contain more than one loop. Finally, taking the limit  $\omega \rightarrow \infty$  the dominating term of the expansion is the leading one, which represents precisely a closed tour passing through all  $N$  cities.

We start by noticing that the factor graph simplifies to a fully connected graph because we are dealing with two-bodies interactions and the degree of the function node is  $k_a = 2 \forall a \in [1 \dots M]$ . The weight of function node can be written in terms of all the  $N(N-1)/2$  possible couples  $a = \{ij\} \forall i \in [1, \dots, N]$  and  $j \in \{[1, \dots, N] : j > i\}$  and thus reads

$$f_{ij}(\mathbf{S}_i, \mathbf{S}_j) = e^{\omega u_{ij} \mathbf{S}_i \cdot \mathbf{S}_j} = 1 + \omega u_{ij} \mathbf{S}_i \cdot \mathbf{S}_j + \frac{\omega^2}{2} u_{ij}^2 (\mathbf{S}_i \cdot \mathbf{S}_j)^2 + \dots \quad (1.5.5)$$

From (1.2.1) and (1.2.2) we obtain

$$p_{i \rightarrow j}(\mathbf{S}_i) = \prod_{k/i} m_{k \rightarrow i}(\mathbf{S}_i), \quad (1.5.6)$$

$$m_{i \rightarrow j}(\mathbf{S}_j) = \int d\mu(\mathbf{S}_i) f_{ij}(\mathbf{S}_i, \mathbf{S}_j) p_{i \rightarrow j}(\mathbf{S}_i) \quad (1.5.7)$$

where  $m_{i \rightarrow j}(\mathbf{S}_i)$  is the constraint the site  $j$  feels from the neighbors  $i$ . The cavity probability  $p_{i \rightarrow j}$  can be parametrized as usual in terms of the cavity magnetization along the first of the  $m$  component<sup>3</sup>  $x_{i \rightarrow j} = \langle S_i^1 \rangle_{N-1}$

$$p_{i \rightarrow j}(S_i^1) = \frac{1 + x_{i \rightarrow j} S_i^1}{2}. \quad (1.5.8)$$

By using the spherical symmetry  $\int d\mu(\mathbf{S}) \mathbf{S}_k^{2n+1} = 0 \forall n = [0, 1, \dots, \infty]$  and the nilpotency when  $m \rightarrow 0$ , if we replace (1.5.8) into the (1.5.7), we obtain that only the first two terms of the expansion survive and the vector of messages becomes  $m_{i \rightarrow j}(S_j^1) = 1 + \omega u_{ij} x_{i \rightarrow j} S_j^1/2$ . After using it into the (1.5.6) and properly normalizing the probability, we obtain a self consistent equation for the magnetization

$$x_{i \rightarrow j} = \frac{\omega \sum_{k \neq j} u_{ki} x_{k \rightarrow i}}{1 + \omega^2 \sum_{k < k'/j} u_{ki} u_{k'i} x_{k \rightarrow i} x_{k' \rightarrow i}} \quad (1.5.9)$$

<sup>3</sup>This is true because without loss of generality we can always think to break the symmetry along one of the  $m$  components, i.e. the first one, by applying an infinitesimally small field directed along the chosen component

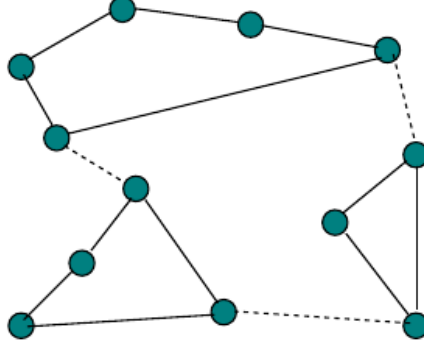


Figure 1.5: The 2-matching problem finds a set of disconnected cycles spanning all the cities with minimum cost whereas the TSP aims at finding the connected optimal tour passing through all the cities.

In terms of the magnetization we can compute also the occupation number using the definition (1.5.2) as

$$n_{ij} = \frac{\omega u_{ij} x_{i \rightarrow j} x_{j \rightarrow i}}{1 + \omega u_{ij} x_{i \rightarrow j} x_{j \rightarrow i}}. \quad (1.5.10)$$

We now take the limit  $\omega \rightarrow \infty$  to select the closed path visiting all the  $N$  cities and after properly rescaling the magnetization  $x_{i \rightarrow j} = x_{i \rightarrow j} / \sqrt{\omega}$  the equations (1.5.9) and (1.5.10) become

$$x_{i \rightarrow j} = \frac{\sum_{k/j} e^{-\beta d_{ki}} x_{k \rightarrow i}}{\sum_{k < k'/j} e^{-\beta d_{ki}} e^{-\beta d_{k'i}} x_{k \rightarrow i} x_{k' \rightarrow i}} \quad (1.5.11)$$

$$n_{ij} = \frac{e^{-\beta d_{ij}} x_{i \rightarrow j} x_{j \rightarrow i}}{1 + e^{-\beta d_{ij}} x_{i \rightarrow j} x_{j \rightarrow i}}. \quad (1.5.12)$$

In the low temperature limit the solution of (1.5.11) selects the shortest cycle identifying all the edges present in the tour. In order to correctly perform the limit, we make the hypothesis that the magnetization in each site scales exponentially as a function of the inverse temperature introducing the cavity fields [49]

$$x_{i \rightarrow j} = \exp(\beta \phi_{i \rightarrow j}).$$

Thus the BP equations (1.5.11) reduce to

$$\begin{aligned} \phi_{i \rightarrow j} &= \widehat{\min}_{k/j} d_{ki} - \phi_{k \rightarrow i} \\ n_{ij} &= \begin{cases} 1 & \text{if } d_{ij} - \phi_{i \rightarrow j} - \phi_{j \rightarrow i} < 0 \\ 0 & \text{otherwise} \end{cases} \end{aligned} \quad (1.5.13)$$



where  $\widehat{\min}_{k/j} \chi_k$  is the second minimum value of the function  $\chi_k$  over all possible  $k$  except  $j$ . In [49] and [59] the authors are interested in computing the length of the shortest tour in the random distance (RD) ensemble, where the distances are distributed according to

$$\rho(d) = \frac{2\pi^D/2}{\Gamma(d/2)} d^{D-1}. \quad (1.5.14)$$

to mimic what happens in euclidean space. The parameter  $D$  plays the role of the dimension in real space and  $\Gamma(x)$  is the Gamma function. Using the cavity method the authors obtain very good results, with respect to direct numerical simulations, for the rescaled length of the optimal tour  $L(D)$ . In [49] they get

$$\ell^c(D) = \lim_{N \rightarrow \infty} N^{1/D} L^c(D)$$

and for  $D = 1$  the cavity prediction  $\ell^c(D = 1) = 2.0416$  perfectly matches the numerical results by Johnson *et al.*  $\ell(1) = 2.0418 \pm 0.0004$  [41] and later confirmed by Sournas [90] with an investigation of the low temperature properties of the model, leading to strong credence to the cavity value for  $D = 1$ . For what concern higher dimensions, where the problems come out from the non-validity of the triangle inequality, we refer the reader to the work of [80], where the authors provide numerical evidence that the  $\ell^c(D)$  is self averaging and the cavity method at the replica symmetric level gives good results also for  $D > 1$ .

We can eventually be interested in obtaining the shortest tour in a given graph realization by numerically solving the set of self consistent equations for a single instance problem. The algorithmic scheme to compute iteratively the solution is summarized in the following:

- First of all, initialize the fields  $\phi_{i \rightarrow j}$  with some random values.
- Iterate the BP equations (1.5.13) until convergence has been reached:
  - At each step update every message  $\phi_{i \rightarrow j}$  in a random order, and compute the difference from the previous value  $\Delta_{ij} = |\phi_{i \rightarrow j}^t - \phi_{i \rightarrow j}^{t-1}|$ .
  - If the maximum difference  $\max_{ij \in E} \Delta_{ij}$  is below a given threshold, end the update procedure and exit.
- Use the  $\phi_{i \rightarrow j}$  to obtain the solutions of the BP equations and then check the condition for a given link  $\langle kl \rangle$  to be present  $d_{kl} - \phi_{k \rightarrow l} - \phi_{k \rightarrow l} < 0$ . If the inequality is satisfied, set the probability of the link to 1 and obtain the set of edges  $\Gamma^*$  selected by the minimum length tour.

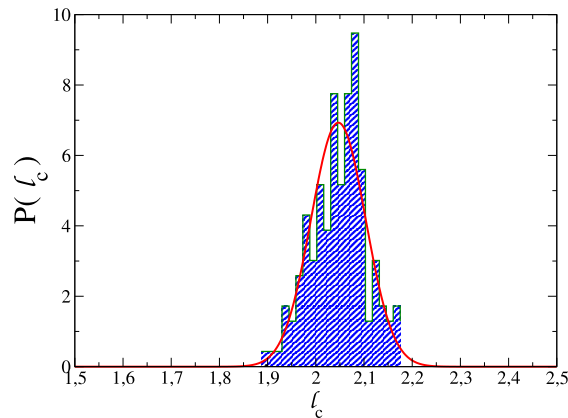


Figure 1.6: In figure we show the histogram of the length tour for  $N = 100$  averaged over 247 samples. The mean value for the tour length is  $\ell^c = 2.046$  in good agreement with the one predicted by [49].

In figure 1.6 we plot as an example the histogram of the length tour founded by the BP algorithm which turns to be peaked around the average value predicted in [49]  $\ell^c = 2.046$  for  $N = 100$ . However if we look at the fixed point solution of BP equations we notice that global connectivity is not correctly enforced namely that  $\Gamma^*$  is composed by a large cycle plus some small loops (see left panel of the figure 1.7). This result can be understood better, by noticing that the BP equations for the TSP are equivalent to the 2-matching problem where the global connectivity constraint is lost. Indeed in the 2-matching problem, we are interested in obtaining a graph  $G$ , spanning the original network, with fixed connectivity  $z = 2$ . Notice that the graph  $G$  forms a set of disjoint cycles which span all the vertices and therefore by construction the global constraint to have a single loop is not required as shown explicitly on the cartoon 1.5.

From this analysis we can conclude that from the energetic point of view, the cavity method seems to be predictive giving correct results for the rescaled minimum length  $\ell(D)$ . However this appears to be not enough in order to identify the optimal tour in a single instance problem. A novel approach, able to deal with global topological constraints, is thus needed.

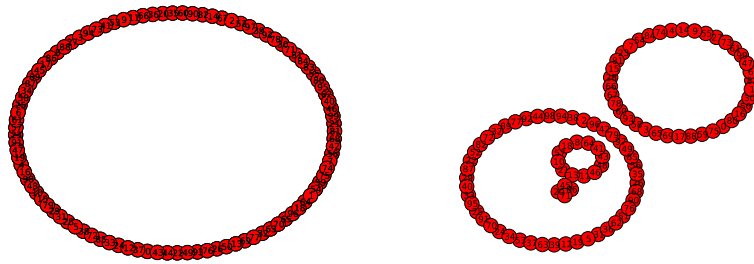


Figure 1.7: On that plot, we show two examples of the tour obtained by the BP algorithm: the first is a real connected tour while the second does not enforce the connectivity constraint thus showing the algorithm solves the two-matching problem.



## Chapter 2

# Finding trees in networks

As a first example of cavity method, we introduced a new algorithm (MST) which aims to find a bounded depth  $D$  spanning tree on a fully connected graph. Inspired by the recent work [7], we show how it is possible to enforce a global constraint by introducing new variables for each degree of freedom, physically interpreted as the *depth* from a root node. The MST has some interesting applications in clustering large dataset, on the basis of some similarity measure, and can be shown to interpolates between two highly used *complementary* strategy: partitioning methods and hierarchical clustering. Preliminary applications on two different biological datasets have shown that it is indeed possible to exploit the deviation from the purely  $D = 2$  spherical limit to gain some insight into the data structures. Our method has properties which are of generic relevance for large scale datasets, namely scalability, simplicity and parallelizability.

### 2.1 Global vs Local constraints

As we just mention at the end of the previous chapter, enforcing global property by means of purely local constraints seems to be a challenging task. This relative difficulty is reasonable if we think how to check local properties with respect to the global ones. The former needs basically a local examination of the graph whereas in order to control the latter, we need to run a search algorithm which spans the whole network. The best example that enhances this topic is the cavity formalism of the TSP problem. Using the standard argument á la De Gennes we have shown in section 1.5, how to obtain the optimal tour length from the  $O(m)$ -model, realizing that something is lost by performing the analytic continuation. In fact, the BP equations for the TSP are equivalent to that of the 2-matching problem,

which aims at finding the optimal set of disconnected cycles.

The failure of the previous technique suggests the need to enforce explicitly the global constraint. In this sense, the work [7] introduces a new representation of the problem to exploit the topology of the tree and enforcing it by local constraint. This method works only on tree which are well defined by the simple local property: if two nodes  $i$  and  $j$  are nearest neighbors, necessarily their *distance* with respect to any selected node, called root, has to satisfy the condition  $|d_i - d_j| = 1$ . This intuition has inspired the *arborescent* representation of the problem which allows to implement explicitly global connectivity constraints in terms of local ones. The term arborescent is associated to the rooted tree construction. Each node of the rooted tree is identified by two degrees of freedom. The first one is the distance  $d_i \forall i \in [1, \dots, N]$  that assumes all integer value from zero to the maximum distance  $D$  from the root node. Indeed, the root node conventionally has zero distance. Secondly, the spin-like degrees of freedom  $s_i$  are substituted with  $\pi_i$ , a set of  $N$ -valued integer variables ranging from  $[1, \dots, N]$ . We describe in detail the new representation in the following and see how it is enough to impose the global connectivity.

Finally, in this chapter we are going to illustrate how the method can be used to obtain an efficient algorithm to cluster large dataset which interpolates between two well known clustering methods: Affinity Propagation [28] and Single Linkage [24].

## 2.2 Clustering

A standard approach to data clustering, that we will also follow here, involves defining a distance measure between objects, called dissimilarity. In this context, generally speaking data clustering deals with the problem of classifying objects so that those, within the same class or cluster, are more similar than those belonging to different classes. The choice of both the measure of similarity and the clustering algorithms are crucial in the sense that they define an underlying model for the cluster structure. In this chapter we discuss two somewhat opposite clustering strategies, and show how they nicely fit as limit cases of a more general scheme that we propose.

Two well-known general approaches that are extensively employed are partitioning methods and hierarchical clustering methods [38]. Partitioning methods are based on the choice of a given number of *centroids* – *i.e.* reference elements – to which the other elements have to be compared. In this sense the problem reduces to finding a set of centroids that minimizes the cumulative distance to points on the dataset. Two of the most used partition-

ing algorithms are  $K$ -means (KM) and Affinity Propagation (AP)[52, 28]. Behind these methods, there is the assumption of spherical distribution of data: clusters are forced to be loosely of spherical shape, with respect to the dissimilarity metric. These techniques give good results normally only when the structure underlying the data fits this hypothesis. Nevertheless, with Soft Affinity Propagation [52] the hard spherical constraint is relaxed, allowing for cluster structures including deviation from the regular shape. This method however recovers partially information on hierarchical organization. On the other hand, Hierarchical Clustering methods such as single linkage (SL) [24], starts by defining a cluster for each element of the system and then proceeds by repeatedly merging the two closest clusters into one. This procedure provides a hierarchic sequence of clusters.

Recently an algorithm to efficiently approximate optimum spanning trees with a maximum depth  $D$ , that has nothing to do with the space dimension, was presented in [7]. We show here how this algorithm may be used to cluster data, in a method that can be understood as a generalization of both (or rather an interpolation between) the AP and SL algorithms. Indeed in the  $D = 2$  and  $D = N$  limits - where  $N$  is the number of object to cluster - one recovers respectively AP and SL methods. As a proof of concept, we apply the new approach to a collection of biological and medical clustering problems on which intermediate values of  $D$  provide new interesting results. In the next section, we define the objective function for clustering based on the cost of certain trees over the similarity matrix, and we devise a message-passing strategy to find an assignment that optimize the cost function. Then we recover the two known algorithms, AP and SL, which are shown to be special cases for appropriately selected values of the external parameters  $D$ . Finally, in the last section we use the algorithm on two biological/medical data clustering problems for which external information can be used to validate the algorithmic performance. First, we tackle the problem of clustering homologous proteins based only on their amino acid sequences and finally, we consider a clustering problem arising in the analysis causes-of-death in regions where vital registration systems are not available.

### 2.3 A Common Framework

Let us start with some definitions. Given  $N$  data points, we introduce the similarity matrix between pairs  $s_{ij}$ , where  $i, j \in [1, \dots, N]$ . This interaction could be represented as a fully connected weighted graph  $G(N, S)$  where  $S$  is the set of weights  $S$  associated to each edge. This matrix constitutes the only data input for the clustering methods we present hereafter. We refer in

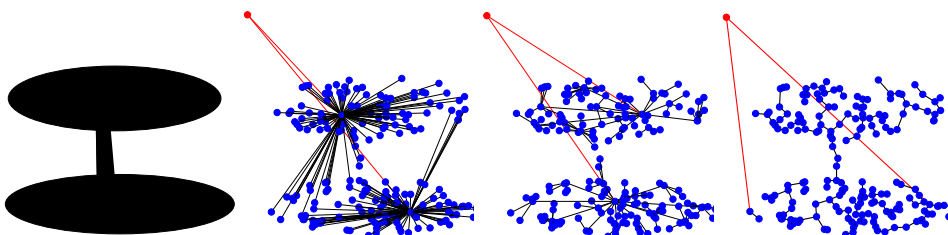


Figure 2.1: Clustering an artificial 2D image. The black image on the left was randomly sampled and the euclidean distance was used as a measure of dissimilarity between nodes. Clustering by  $D$ -MST was then attempted on the resulting graph. One external root vertex  $v^*$  (red point) was added, with distance  $\lambda$  to every other points. The output of the algorithm consists in a minimum weight rooted spanning tree of depth  $D$  pointed out by bold links. The last three figures concern the resulting clustering for different choices of the depth limit  $D = 2, 4, > N$  respectively. Different clusters with a complex internal structure can be recovered after removing the red node  $v^*$ . In the case of AP  $D = 2$  (second figure) the spherical clusters do not fit the ellipsoidal shape of the original figure while for 4-MST (third figure) the structure of two ellipses can be recovered. The fourth and last figure corresponds to SL ( $D > N$ ): in this case nodes are split into two arbitrary components disregarding the original shape.

the following to the neighborhood of node  $i$  with the symbol  $\partial(i)$ , denoting the ensemble of all nearest neighbors of  $i$ . By adding to the graph  $G$  one artificial node  $v^*$ , called *root*, whose similarity to all other nodes  $i \in G$  is a constant parameter  $\lambda$ , we obtain a new graph  $G^*(N + 1, S^*)$  where  $S^*$  is a  $(N + 1) \times (N + 1)$  matrix with one added row and column of constant value to the matrix  $S$  (see figure 2.1).

We will employ the following general scheme for clustering based on trees. Given any tree  $T$  that spans all the nodes in the graph  $G^*(N + 1, S^*)$ , consider the (possibly disconnected) subgraph resulting of removing the root  $v^*$  and all its links. We will define the output of the clustering scheme as the family of vertex sets of the connected components of this subgraph. That is, each cluster will be formed by a connected component of the pruned  $T \setminus v^*$ . In the following, we will concentrate on how to produce trees associated to  $G^*$ .

The algorithm described in [7] was devised to find a tree of minimum weight with a depth bounded by  $D$  from a selected root to a set of terminal nodes. In the clustering framework, all nodes are terminals and must be reached by the tree. As a tree has exactly  $N - 1$  links, for values of  $D$  greater



or equal than  $N$  the problem becomes the familiar (unconstrained) minimum spanning tree problem. In the rest of this section we will describe the  $D$ -MST message passing algorithm of [7] for Steiner trees in the simplified context of (bounded depth) spanning trees.

To each node of the graph we associate two variables  $\pi_i$ , and  $d_i$  as sketched in figure 2.2, where  $\pi_i \in \partial i$  could be interpreted as a pointer from  $i$  to one of the neighboring nodes  $j \in \partial(i)$ . Meanwhile  $d_i \in [0, \dots, D]$  is thought as a discrete distance between the node  $i$  and the root  $v^*$  along the tree. Necessarily only the root has zero distance  $d_{v^*} = 0$ , while for all other nodes  $d_i \in [1, \dots, D]$ . In order to ensure global connectivity of the  $D$ -MST, these two variables must satisfy the following condition:  $\pi_i = j \Rightarrow d_i = d_j + 1$ . This means that if node  $j$  is the parent of node  $i$ , then the depth of node  $i$  must exceed the depth of the node  $j$  by precisely one. This condition avoids the presence of loops and forces the graph to be connected, assigning non-null weight only to configurations corresponding to trees. The energy function thus reads

$$E(\{\pi_i, d_i\}_{i=1}^N) = \sum_i s_i \pi_i - \sum_{i,j \in \partial(i)} e_{ij}(\pi_i, \pi_j, d_i, d_j) + e_{ji}(\pi_j, \pi_i, d_j, d_i), \quad (2.3.1)$$

where  $e_{ij}$  is defined as

$$e_{ij} = \begin{cases} 0 & \{\pi_i = j \Rightarrow d_i = d_j + 1\} \\ -\infty & \text{else} \end{cases} \quad (2.3.2)$$

In this way only configurations corresponding to a tree are taken into account with the usual Boltzmann weight factor  $e^{-\beta s_i \pi_i}$  where the external parameter  $\beta$  fixes the value of energy level. Thus the partition function is

$$Z(\beta) = \sum_{\{\pi_i, d_i\}} e^{-\beta E(\{\pi_i, d_i\})} = \sum_{\{\pi_i, d_i\}} \prod_i e^{-\beta s_i \pi_i} \times \prod_{ij \in \partial(i)} f_{ij}, \quad (2.3.3)$$

where we have introduced an indicator function of pairwise interactions

$$f_{ij} = g_{ij} g_{ji}.$$

Each term  $g_{ij} = 1 - \delta_{\pi_i, j} (1 - \delta_{d_j, d_i - 1})$  is equivalent to  $e^{e_{ij}}$  while  $\delta_{ij}$  is the delta function. In terms of these quantities  $f_{ij}$  it is possible to derive the cavity equations, i.e. the following set of coupled equations for the cavity marginal probability  $p_{j \rightarrow i}(d_j, \pi_j)$  of each site  $j \in [1, \dots, N]$  after removing one of the nearest neighbors  $i \in \partial(j)$ :

$$p_{j \rightarrow i}(d_j, \pi_j) \propto e^{-\beta s_i \pi_i} \prod_{k \in \partial(j)/i} m_{k \rightarrow j}(d_j, \pi_j) \quad (2.3.4)$$

$$m_{k \rightarrow j}(d_j, \pi_j) \propto \sum_{d_k, \pi_k} p_{k \rightarrow j}(d_k, \pi_k) f_{jk}(d_j, \pi_j, d_k, \pi_k). \quad (2.3.5)$$

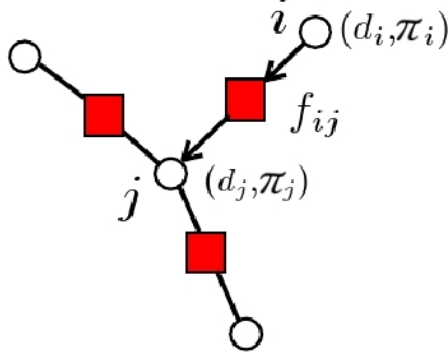


Figure 2.2: The graphical representation of the BP equation for the MST problem is sketched in figure. To each variable node two integer degrees of freedom are defined: one of them being the distance from the root and the other is interpreted as a pointer  $\pi_i$ . Every couple of nearest neighbors has a constraint  $f_{ij}$  shows as a red square in the cartoon. This representation is enough to enforce the global property of *being a tree* by means of local constraints.

These equations are solved iteratively and in graphs with no cycles they are guaranteed to converge to a fixed point that is the optimal solution. In terms of cavity probability we are able to compute beliefs using the equations (1.2.5)-(1.2.6)

$$b_j(d_j, \pi_j) \propto \prod_{k \in \partial j} m_{k \rightarrow j}(d_j, \pi_j) \quad (2.3.6)$$

$$b_{ij}(d_i, \pi_i, d_j, \pi_j) \propto p_{i \rightarrow j}(d_i, \pi_i) p_{j \rightarrow i}(d_j, \pi_j) f_{ij}(d_i, \pi_i, d_j, \pi_j). \quad (2.3.7)$$

From the algorithmic viewpoint in graph with cycles, the problem of non-convergence can be overcome by introducing a reinforcement perturbation term as in [7]. This leads to a new set of perturbed coupled equations that show good convergence properties.

$$p_{j \rightarrow i}^t(d_j, \pi_j) \propto b_j^{t-1}(d_j, \pi_j)^{\gamma t} e^{-\beta s_i \pi_i} \prod_{k \in \partial(j)/i} m_{k \rightarrow j}^t(d_j, \pi_j) \quad (2.3.8)$$

$$m_{k \rightarrow j}^t(d_j, \pi_j) \propto \sum_{d_k, \pi_k} p_{k \rightarrow j}^{t-1}(d_k, \pi_k) f_{jk}(d_j, \pi_j, d_k, \pi_k). \quad (2.3.9)$$

The  $\beta \rightarrow \infty$  limit is taken by considering the change of variable

$$\psi_{j \rightarrow i}(d_j, \pi_j) = \beta^{-1} \log p_{j \rightarrow i}(d_j, \pi_j) \quad (2.3.10)$$

$$\phi_{j \rightarrow i}(d_j, \pi_j) = \beta^{-1} \log m_{j \rightarrow i}(d_j, \pi_j) \quad (2.3.11)$$

thus the relations 2.3.4-2.3.5 reduce to

$$\psi_{j \rightarrow i}(d_j, \pi_j) = -s_i \pi_i + \sum_{k \in \partial(j)/i} \phi_{k \rightarrow j}(d_j, \pi_j) \quad (2.3.12)$$

$$\phi_{k \rightarrow j}(d_j, \pi_j) = \max_{d_k \pi_k: f_{kj} \neq 0} \psi_{k \rightarrow j}(d_k, \pi_k). \quad (2.3.13)$$

These equations are in the Max-Sum form and equalities hold up to some additive constant. In terms of these quantities, marginals are given by

$$\psi_j(d_i, \pi_j) = -s_j \pi_j + \sum_k \phi_{k \rightarrow j}(d_j, \pi_j) \quad (2.3.14)$$

and the optimum tree is the one obtained by  $\operatorname{argmax} \psi_j$ . If we introduce the variables

$$\begin{aligned} A_{k \rightarrow j}^d &= \max_{\pi_k \neq j} \psi_{k \rightarrow j}(d, \pi_k), \\ C_{k \rightarrow j}^d &= \psi_{k \rightarrow j}(d, j), \\ E_{k \rightarrow j}^d &= \max(C_{k \rightarrow j}^d, A_{k \rightarrow j}^d) \end{aligned} \quad (2.3.15)$$

we can compute all the variable  $\phi_{k \rightarrow j}(d_j, \pi_j) = A_{k \rightarrow j}^{d_j-1}, E_{k \rightarrow j}^{d_j}$  for  $\pi_j = k$  and  $\pi_j \neq k$  respectively. Using equations 2.3.12 and 2.3.13 we obtain the previous quantities satisfy the following set of equations:

$$\begin{aligned} A_{j \rightarrow i}^d(t+1) &= \sum_{k \in \partial(j)/i} E_{k \rightarrow j}^d(t) + \max_{k \in \partial(j)/i} \left( A_{k \rightarrow j}^{d-1}(t) - E_{k \rightarrow j}^d(t) - s_{jk} \right) \\ C_{j \rightarrow i}^d(t+1) &= -s_{ji} + \sum_{k \in \partial(j)/i} E_{k \rightarrow j}^d(t) \\ E_{j \rightarrow i}^d(t+1) &= \max \left( C_{j \rightarrow i}^d(t+1), A_{j \rightarrow i}^d(t+1) \right). \end{aligned} \quad (2.3.16)$$

It has been demonstrated [8] that a fixed point of these equations with depth  $D > N$  is an optimal spanning tree, meaning that the spanning tree found in the network is associated to the minimum weight. Indeed, for  $D < N$  the problem is not P and become difficult to solve and MST is an approximated algorithm.

In the following two subsections, we show how to recover the SL and AP algorithms. On one hand, by computing the (unbounded depth) spanning tree on the enlarged matrix and then considering the connected components of its restriction to the set of nodes removing  $v^*$ , we recover the results obtained by SL. On the other hand we obtain AP by computing the  $D = 2$  spanning tree rooted at  $v^*$ , defining the self-affinity parameter as the weight to reach this root node.

## 2.4 Single Linkage limit

Single Linkage is one of the oldest and simplest clustering methods, and there are many possible descriptions of it. One of them is the following: order all pairs according to distances, and erase as many of the pairs with largest distance so that the number of resulting connected components is exactly  $k$ . Define clusters as the resulting connected components.

An alternative method consists in removing initially all *useless* pairs (i.e. pairs that would not change the set of components when removed in the above procedure). This reduces to the following algorithm: given the distance matrix  $S$ , compute the minimum spanning tree on the complete graph with weights given by  $S$ . From the spanning tree remove the  $k - 1$  links with largest weight. Clusters are given by the resulting connected components. In many cases there is no *a priori* desired number of clusters  $k$  and an alternative way of choosing  $k$  is to use a continuous parameter  $\lambda$  to erase all weights larger than  $\lambda$ .

The  $D$ -MST problem for  $D > N$  identifies the minimum spanning tree connecting all  $N + 1$  nodes (including the root  $v^*$ ). This means each node  $i$  will point to one other node  $\pi_i = j \neq v^*$  if its weight satisfies the condition  $\min_j s_{ij} < s_{iv^*}$ , otherwise it would be cheaper to connect it to the root (introducing one more cluster). We will make this description more precise. For simplicity, let us assume no edge in  $G(N, S)$  has weight exactly equal to  $\lambda$ .

The Kruskal algorithm [42] is a classical algorithm to compute a minimum spanning tree. It works by iteratively creating a forest as follows: start with a subgraph all nodes and no edges. Then scan the list of edges ordered by increasing weight, and add the edge to the forest if it connects two different components (i.e. if it does not close a loop). At the end of the procedure, it is easy to prove that the forest has only one connected component that forms a minimum spanning tree. It is also easy to see that the edges added when applying the Kruskal algorithm to  $G(N, S)$  up to the point when the weight reaches  $\lambda$  are also admitted on the Kruskal algorithm for  $G(N + 1, S^*)$ . After that point, the two procedures diverge because on  $G(N, S)$  the remaining added edges have weight larger than  $\lambda$  while on  $G(N + 1, S^*)$  all remaining added edges have weight exactly  $\lambda$ . Summarizing, the MST on  $G(N + 1, S^*)$  is a MST on  $G(N, S)$  on which all edges with weight greater than  $\lambda$  have been replaced by edges connecting with  $v^*$ .

## 2.5 Affinity propagation limit

Affinity Propagation is a method that was recently proposed in [28], based on the choice of a number of “exemplar” data-points. Starting with a similarity matrix  $S$ , choose a set of exemplar data points  $X \subset V$  and an assignment  $\phi : V \mapsto X$  such that:  $\phi(x) = x$  if  $x \in X$  and the sum of the distances between datapoints and the exemplars they map to is minimized. It is essentially based on iteratively passing two types of messages between elements, representing *responsibility* and *availability*. The first,  $e_{i \rightarrow j}$ , measures how much an element  $i$  would prefer to choose the target  $j$  as its exemplar. The second  $a_{i \rightarrow j}$  tells the preference for  $i$  to be chosen as an exemplar by datapoint  $j$ . This procedure is an efficient implementation of the Max-Sum algorithm that improves the naive exponential time complexity to  $O(n^2)$ . The self-affinity parameter, namely  $s_{ii}$ , is chosen as the dissimilarity of an exemplar with himself, and *in fine* regulates the number of groups in the clustering procedure, by allowing more or less points to link with “dissimilar” exemplars.

Given a similarity matrix  $S$  for  $N$  nodes, we want to identify the *exemplars*, that is, to find a valid configuration  $\bar{\pi} = \{\pi_1, \dots, \pi_N\}$  such that  $\pi : [1, \dots, N] \mapsto [1, \dots, N]$  so as to minimize the function

$$E(\bar{\pi}) = - \sum_{i=1}^N s_{i\pi_i} - \sum_i \delta_i(\bar{\pi}), \quad (2.5.1)$$

where the constraint reads

$$\delta_i(\bar{\pi}) = \begin{cases} -\infty & \pi_i \neq i \cap \exists j : \pi_j = i \\ 0 & \pi_i = i \cup \{ \forall j \in [1, \dots, N] c_j \neq i \} \end{cases} \quad (2.5.2)$$

These equations take into account the only possible configurations, where node  $i$  either is an exemplar, meaning  $\pi_i = i$ , or it is not chosen as an exemplar by any other node  $j$ . The energy function thus reads

$$E(\bar{\pi}) = \begin{cases} - \sum_i s_{i\pi_i} & \forall i \{ \pi_i = i \cup \forall j \pi_j \neq i \} \\ \infty & \text{else} \end{cases} \quad (2.5.3)$$

The cavity equations are computed starting from this definition and after some algebra they reduce to the following update conditions for responsibility and availability [28]:

$$r_{i \rightarrow k}^{t+1} = s_{ik} - \max_{k' \neq k} (a_{k' \rightarrow i}^t + s_{k'i}) \quad (2.5.4)$$

$$a_{k \rightarrow i}^{t+1} = \min \left( 0, e_{k \rightarrow k} + \sum_{i' \neq k} \max(0, e_{i' \rightarrow k}^t) \right). \quad (2.5.5)$$

In order to prove the equivalence between the two algorithms, i.e.  $D$ -MST for  $D = 2$  and AP, we show in the following how the two employ an identical decomposition of the same energy function thus resulting necessarily to the same max sum equations. In the 2-MST equations, we are partitioning all nodes into three groups: the first one is only the root whose distance  $d = 0$ , the second one is composed of nodes pointing at the root  $d = 1$  and the last one is made up of nodes pointing to other nodes that have distance  $d = 2$  from the root. The following relations between  $d_i$  and  $\pi_i$  makes this condition explicit:

$$d_i = \begin{cases} 1 & \Leftrightarrow \pi_i = v^* \\ 2 & \Leftrightarrow \pi_i \neq v^* \end{cases} \quad (2.5.6)$$

It is clear that the distance variable  $d_i$  is redundant because the two kind of nodes are perfectly distinguished with just the variable  $\pi_i$ . Going a step further we could remove the external root  $v^*$  upon imposing the following condition for the pointers  $\pi_i = i \Leftrightarrow \pi_i = v^*$   $\pi_i = j \neq i \Leftrightarrow \pi_i \neq v^*$ . This can be understood by thinking at AP procedure: since nodes at distance one from the root are the exemplars, they might point to themselves, as defined in AP, and all the non-exemplars are at distance  $d = 2$  so they might point to nodes at distance  $d = 1$ . Using this translation, from Equation 2.3.2 it follows that

$$\sum_{ij \in \partial i} e_{ij} + e_{ji} = \begin{cases} 0 & \forall i \{ \pi_i = i \cup \forall j \neq i \pi_j \neq i \} \\ -\infty & \text{else} \end{cases} \quad (2.5.7)$$

meaning that the constraints are equivalent  $\sum_{ij \in \partial i} e_{ij} + e_{ji} = \sum_i \delta_i(\bar{\pi})$ . Substituting (2.5.7) into equation (2.3.1) we obtain that

$$E(\{\pi_i, d_i\}_{i=1}^n) = \begin{cases} -\sum_i s_{i\pi_i} & \forall i \{ \pi_i = i \cup \forall j \neq i \pi_j \neq i \} \\ \infty & \text{else} \end{cases} \quad (2.5.8)$$

The identification of the self affinity parameter and the self similarity,  $\lambda = s_{ii}$ , allows us to prove the equivalence between this formula and the AP energy given in equation (2.5.3) as desired.

## 2.6 Applications to biological data

In the following sections we shall apply the new technique to different clustering problems and give a preliminary comparison to the two extreme limits of the interpolation, namely  $D = 2$  (AP) and  $D = N$  (SL).

Clustering is a widely used method of analysis in biology, most notably in the recent fields of transcriptomics [25], proteomics and genomics[5], where huge quantities of noisy data are generated routinely. A clustering approach presents many advantages for this type for data: it can use all pre-existing knowledge available to choose group numbers and to assign elements to groups, it has good properties of noise robustness[23], and it is computationally more tractable than other statistical techniques. In this section apply our algorithm to structured biological data, in order to show that by interpolating between two well-known clustering methods (SL and AP) it is possible to obtain new insight.

### 2.6.1 Clustering of protein datasets

An important computational problem is grouping proteins into families according to their sequence only. Biological evolution lets proteins fall into so-called families of similar proteins - in term of molecular function - thus imposing a natural classification. Similar proteins often share the same three-dimensional folding structure, active sites and binding domains, and therefore have very close functions. They often - but not necessarily - have a common ancestor, in evolutionary terms. To predict the biological properties of a protein based on the sequence information alone, one either needs to be able to predict precisely its folded structure from its sequence properties, or to assign it to a group of proteins sharing a known common function. This second possibility stems almost exclusively from properties conserved in through the evolutionary time, and is computationally much more tractable than the first one. We want here to underline how our clustering method could be useful to handle this task, in a similar way as the one we used in the first application, by introducing a notion of distance between proteins based only on their sequences. The advantage of our algorithm is its global approach: we do not take into account only distances between a couple of proteins at a time, but we solve the clustering problem of finding all families in a set of proteins in a *global* sense. This allows the algorithm to detect cases where related proteins have low sequence identity.

To define similarities between proteins, we use the BLAST E-value as a distance measure to assess whether a given alignment between two different protein sequences constitutes evidence for homology. This classical score is computed by comparing how strong an alignment is with respect to what is expected by chance alone. This measure accounts for the length of the proteins, as long proteins have more chance to randomly share some subsequence. In essence, if the E-value is 0 the match is perfect while the more

E-value is high the more the average similarity of the two sequences is low and can be considered as being of no evolutionary relevance. We perform the calculation in a all-by-all approach using the BLAST program, a sequence comparison algorithm introduced by Altshul et al. [3].

Using this notion of distance between proteins we are able to define a matrix of similarity  $S$ , in which each entry  $s_{ij}$  is associated to the E-value between protein  $i$  and  $j$ . The  $D$ -MST algorithm is then able to find the directed tree between all the sets of nodes minimizing the same cost function as previously. The clusters we found are compared to those computed by other clustering methods in the literature, and to the “real” families of function that have been identified experimentally.

As in the work by [75], we use the Astral 95 compendium of SCOP database [73] where no two proteins share more than 95% of similarity, so as not to overload the clustering procedure with huge numbers of very similar proteins that could easily be attributed to a cluster by direct comparison if necessary. As this dataset is hierarchically organized, we choose to work at the level of superfamilies, in the sense that we want identify, on the basis of sequence content, which proteins belong to the same superfamily. Proteins belonging to the same superfamily are evolutionary related and share functional properties. Before going into the detail of the results we want to underline the fact that we do not modify our algorithm to adapt to this dataset structure, and without any prior assumption on the data, we are able to extract interesting information on the relative size and number of clusters selected (Fig.2.4). Notably we do not use a training set to optimize a model of the underlying cluster structure, but focus only on raw sequences and alignments.

One issue that was recently put forward is the alignment variability [98] depending on the algorithms employed. Indeed some of our results could be biased by errors or dependence of the dissimilarity matrix upon the particular details of the alignments that are used to compute distances, but in the framework of a clustering procedure these small-scale differences should stay unseen due to the large scale of the dataset. On the other hand, the great advantage of working only with sequences is the opportunity to use our method on datasets where no structure is known a priori, such as fast developing metagenomics datasets [95].

We choose as a training set 5 different superfamilies belonging to the ASTRAL 95 compendium for a total number of 661 proteins: a) Globin-like, b) EF-hand, c) Cupredoxin, d) Trans-Glycosidases and e) Thioredoxin-like. Our algorithm is able to identify a good approximation on the real number of clusters. Here we choose the parameter  $\lambda$  well above the typical weight



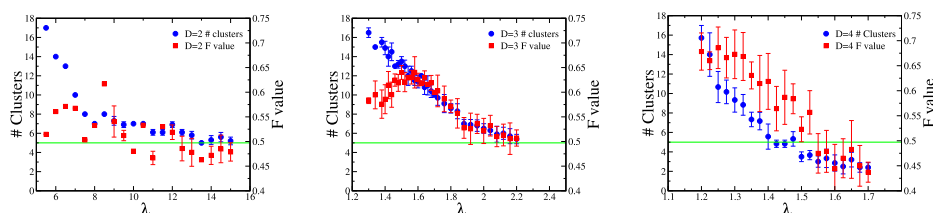


Figure 2.3: In the three panels we show the average number of clusters over the random noise as a function of the weight of the root for  $D = 2, 3, 4$  respectively. For each graph we show the number of clusters (circle) and the associated F value (square), computed as a function of precision and recall. We want to emphasize the fact the highest F values are reached for depth  $D = 4$  and weight  $\lambda \sim 1.3$ . With this choice of the parameters we found the number of clusters is of order 10, a good approximation of the number of superfamilies shown in figure as a straight line.

between different nodes, so as to minimize the number of groups found. As a function of this weight you can see the number of clusters found by the  $D$ -MST algorithm reported in figure 2.3, for the depths  $D = 2, 3, 4$ . In these three plots we see the real value of the number of clusters is reached for different values of the weight  $\lambda \sim 12, 2, 1.4$  respectively. The performance of the algorithm can be analyzed in terms of precision and recall. These quantities are combined in the  $F$ -value [75] defined as

$$F = \frac{1}{N} \sum_h n_h \max_i \frac{2n_i^h}{n^h + n_i}, \quad (2.6.1)$$

where  $n_i$  is the number of nodes in cluster  $i$  according to the classification  $\lambda$  we find with the  $D$ -MST algorithm,  $n^h$  is the number of nodes in the cluster  $h$  according to the real cluster classification  $K$  and  $n_i^h$  is the number of predicted proteins in the cluster  $i$  and at the same time in the cluster  $h$ . In both cases the algorithm performs better results for lower value of  $\lambda$ . This could be related to the definition of the  $F$  value because starting to reduce the number of expected clusters may be misleading in the accuracy of the predicted data clustering.

Since distances between datapoints have been normalized to be real numbers between 0 to 1, when  $\lambda \rightarrow \infty$  we expect to find the number of connected components of the given graph  $G(N, S)$ . While lowering this value, we start to find some configurations which minimize the weight respect to the single cluster solution. The role played by the external parameter  $\lambda$  could be seen as the one played by a chemical potential tuning from outside the average

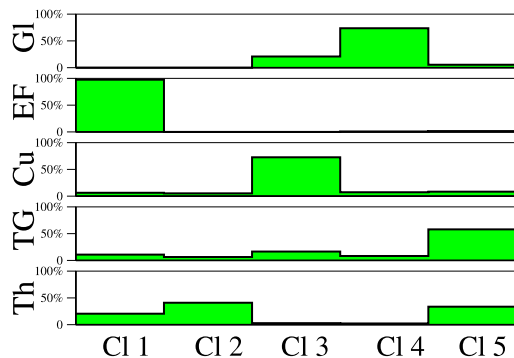


Figure 2.4: We show the results of clustering proteins of the 5 subfamilies Globin-like (GI), EFhand (EF), Cupredoxin (Cu), Trans-Glycosidases (TG), Thioredoxin-like (Th) using 4-MST with parameter  $\lambda=1.45$ . We see that most of the proteins of the first three families (GI, EF and Cu) are correctly grouped together respectively in cluster 4, 1 and 3 while the last two families are identified with clusters 2 and 5 with some difficulties.

number of clusters.

We compare our results to the ones in [75] for different algorithms and it is clear that intermediate values of  $D$  gives best results on the number of clusters detected and on the  $F$ -value reached without any a priori treatment of data. It is also clear that  $D$ -MST algorithm with  $D = 3, 4, 5$  gives better results than AP (case  $D = 2$ ) as can be seen in Fig. 2.5.

We believe that the reason is that clusters do not have an intrinsic spherical regularity. This may be due to the fact that two proteins having a high number of differences between their sequences at irrelevant sites can be in the same family. Such phenomena can create clusters with complex topologies in the sequence space, hard to recover with methods based on a spherical shape hypothesis. We compute the  $F$ -value also in the single linkage limit ( $D > N$ ) and its value is almost  $\sim 0.38$  in all the range of clusters detected. This shows that the quality of the predicted clusters improves reaching the highest value when  $D = 4$  and then decreases when the maximum depth increases.

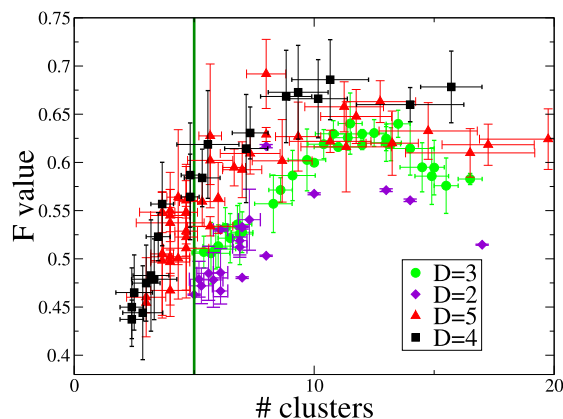


Figure 2.5: We plot the F value for depths  $D = 2, 3, 4, 5$  as a function of the number of clusters found by the  $D$ -MST algorithm. The case  $D = 2$  provides the AP results while  $D > N$  is associated to SL and gives value well below 0.4. The highest performance in terms of the F value is reached for depth  $D = 4$  and number of clusters  $\sim 10$ . We draw a line in correspondence to the presumed number of clusters which is 5 where again the algorithm with parameter  $D = 4$  obtains the highest performance score.

## 2.6.2 Clustering of verbal autopsy data

The verbal autopsy is an important survey-based approach to measuring cause-specific mortality rates in populations for which there is no vital registration system [72, 30]. We applied our clustering method to the results of 2039 questionnaires in a benchmark verbal autopsy dataset, where gold-standard cause-of-death diagnosis is known for each individual. Each entry in the dataset is composed of responses  $n = 47$  *yes/no/don't know* questions.

To reduce the effect of incomplete information, we restricted our analysis to the responses for which at least 91% of questions answered yes or no (in other words, at most 9% of the responses were “don’t know”). This leaves 743 responses to cluster (see [72] for a detailed descriptive analysis of the response patterns in this dataset.)

The goal of clustering verbal autopsy responses is to infer the common causes of death on the basis of the answers. This could be used in the framework of “active learning”, for example, to identify which verbal autopsies require further investigation by medical professionals.

As in the previous applications, we define a distance matrix on the verbal autopsy data and apply  $D$ -MST with different depths  $D$ . The questionnaires

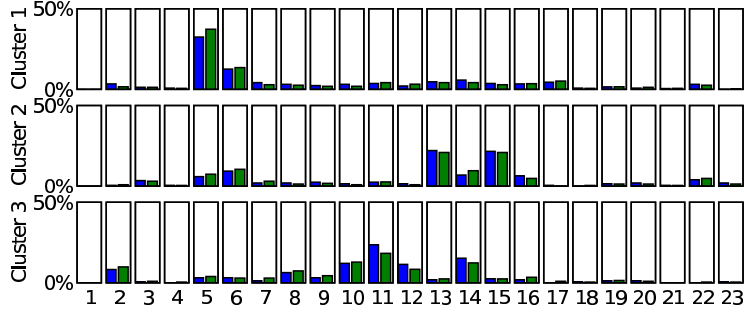


Figure 2.6: Cluster decomposition broken down by cause-of-death (from 1 to 23) produced by AP (blue) and  $D$ -MST (green). The parameter  $\lambda$  is chosen from the stable region, where the number of clusters is constant.

are turned into vectors by associating to the answers yes/no/don't know the values 0/1/0.5 respectively. The similarity matrix is then computed as the root mean square difference between vectors,

$$s_{ij} = \frac{1}{n} \sqrt{\sum_k (r_i(k) - r_j(k))^2},$$

where  $r_i(k) \in \{0, 1, 0.5\}$  refers to the symptom  $k \in [0, n]$  in the  $i$ -th questionnaire.

We first run 2-MST (AP) and 4-MST on the dataset and find how the number of clusters depend on  $\lambda$ . We identify a stable region which corresponds to 3 main clusters for both  $D = 2, 4$ . As shown in figure 2.6, to each cluster we can associate a different causes of death. Cluster 1 contains nearly all of the Ischemic Heart Disease deaths (cause 5) and about half of the Diabetes Mellitus deaths (cause 6). Cluster 2 contains most of the Lung Cancer deaths (cause 13) and Chronic Obstructive Pulmonary Disease deaths (cause 15). Cluster 2 also contains most of the additional IHD and DM deaths (30% of all deaths in the dataset are due to IHD and DM). Cluster 3 contains most of the Liver Cancer deaths (cause 11) as well as most of the Tuberculosis deaths (cause 2) and some of the other prevalent causes. For  $D = 2$  we find no distinguishable hierarchical structure in the 3 clusters, while for higher value we find a second-level structure. In particular for  $D = 4$  we obtain 57-60 subfamilies for value of  $\lambda$  in the region of 0.15 – 0.20. Although the first-level analysis (Fig.2.6) underlines the similarity of  $D$ -MST algorithm with AP, increasing the depth leads to a finer

sub-clusters decomposition.



## Chapter 3

# Identification of subgraphs

The problem of identifying trees in network can be exploit using the property of the distance: by defining a root node, it is not possible to have two nearest neighbor nodes at the same distance from the root. This property is peculiar only for tree-like structure and is not useful for other type of graphs. In this sense, we find the motivations to introduce a more general method able to identify any given possible subgraphs embedded in a large network. We now introduce a basic algorithm that solves more generally the Graph Alignment problem (GA) dealing with weighted graphs and can be specialized to Subgraphs Isomorphism problem (SI).

### 3.1 Graph Alignment Problem

We are going to present the BP implementation for the GA. It aims at finding a mapping between two weighted graphs  $g(N, S)$  and  $G(N', S')$ , where  $N \leq N'$ , such that they show major correlations. Weighted graphs are special classes of graphs whose edges are weighted. Numerical weights  $s_{ij} \in S$  are sometimes referred to as costs, especially if they are positive and we restrict ourselves to  $s_{ij} \in [0, 1]$  so to interpret them as probability. To facilitate the notation, primed quantities, always refer to  $G(N', S')$ . GA is a very general problem that displays several NP-hard special cases, like SI or TSP. The former is obtained when  $N < N'$  and the weights reduce to 0 – 1, namely  $S$  ( $S'$ ) becomes the adjacency matrix  $A$  ( $A'$ ). The  $A$  matrix summarizes the relations between the nodes

$$\begin{aligned} a_{ij} &= 1 && \text{if } i \in \partial(j) \\ a_{ij} &= 0 && \text{else} \end{aligned} \tag{3.1.1}$$

if there is a link between  $i$  and  $j$ ,  $a_{ij} = 1$  while it vanishes when there is not. On the other hand, if we look for a graph  $g$  such that  $s_{ij} \neq 0 \Leftrightarrow |i - j| = 1$

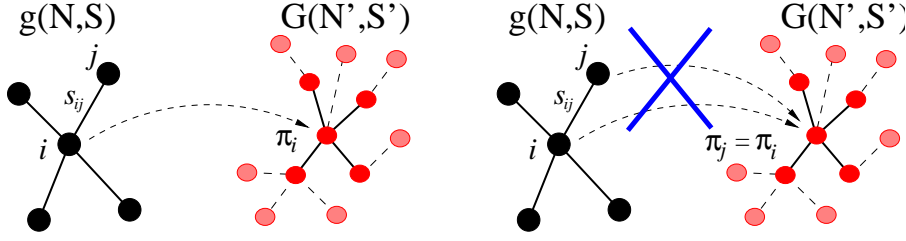


Figure 3.1: The application  $\pi$  has a clear interpretation has a mapping between the nodes of the two graphs as shown in the first panel. In the second picture, we show which are the mappings suppressed by the constraint  $f_{ij}$ , in order to enforce the injectivity constraint.

we recover the TSP.

We start providing a mathematical description of the GA suitable with the statistical formulation of the method. Let us introduce the usual application  $\{\pi : g \rightarrow G\}$ , namely a permutation between the sets of nodes  $V(g)$  and  $V(G)$  that can be visualized as a reshuffling of columns and lines of the matrix  $S'$  as shown in figure 3.2. We define the cost energy to be minimized as

$$E\{\pi\} = - \sum_{i,j>i} s_{ij} s'_{\pi_i \pi_j} - \sum_i h_{i \pi_i}, \quad (3.1.2)$$

such that the zero energy configuration is associated to the maximal overlap between the two graphs. If we define  $\mathcal{O}$ , the distance between the two graphs given the application  $\{\pi\}$ , as

$$\mathcal{O}(\{\pi\}) = \sum_{ij} [s_{ij} - s'_{\pi_i \pi_j}]^2 = \sum_{ij} s_{ij}^2 - 2 \sum_{ij} s_{ij} s'_{\pi_i \pi_j} + \sum_{ij} s'^2_{\pi_i \pi_j}, \quad (3.1.3)$$

it is evident that  $\mathcal{O}(\{\pi\}) \propto E(\{\pi\})$ . So that, minimizing the energy naturally leads to the minimization of the distance between the graphs.

A major problem in this respect, is to enforce the injectivity constraint: for  $i \neq j$  also  $\pi_i \neq \pi_j$  has to hold. The joint probability of a given configuration can thus be written in the form

$$P(\{\pi\}) = \frac{1}{Z} \prod_{i,j>i} f_{ij}(\pi_i, \pi_j). \quad (3.1.4)$$

where the constraints take into account the energetic term plus the injectivity factor

$$f_{ij} = (1 - \delta_{\pi_i, \pi_j}) e^{\beta(s_{ij} s'_{\pi_i \pi_j} + h_{i \pi_i} \delta_{ij})}. \quad (3.1.5)$$

The first term in the right hand side vanishes if two different nodes in the subgraph  $g$  are mapped into the same node  $\pi_i$  in  $G$ , see the cartoon in figure 3.1 for a graphical explanation.



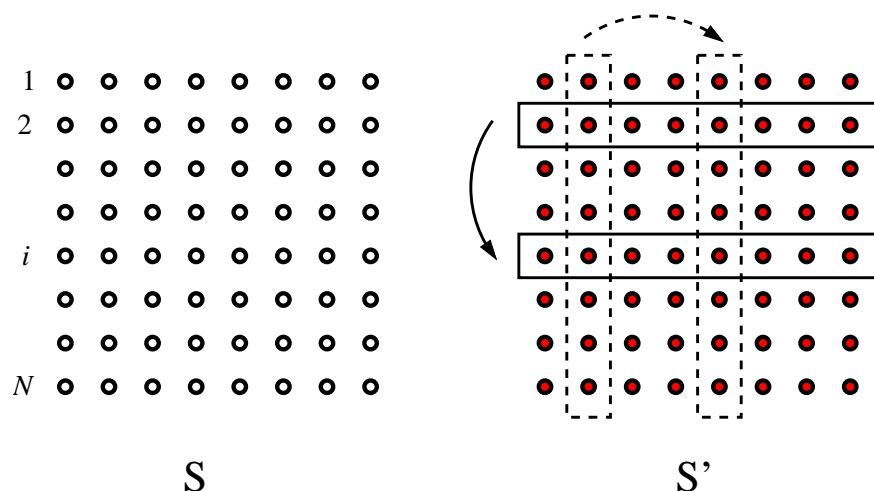


Figure 3.2: In figure we show how the permutation affect the graph  $G$ , through a reshuffling of the matrix  $S'$  in order to enforce the maximal overlap between the two graphs  $g$  and  $G$ .

Of course, imposing this *permutation* constraint has a cost in term of computational time because it requires a fully connected factor graph where the number of constraints  $M$  scale as  $O(N^2)$ . But still it is polynomial time calculation and gives reliable results. It is possible, however, to relax the permutation adding a extra term in the Hamiltonian with a new external parameter having the same role of the inverse temperature. This helps in reducing the number of constraints from  $N(N-1)/2$  to the number of links in  $g$  [14]. This method is inspired by the work of [52] on softening affinity propagation and is convenient principally when the graph  $g$  is sparse. On the other side, this simplification is not particularly relevant in identifying dense parts of the network because, in this case, the number of links of graph  $g$  scales as  $O(N^2)$  so that two implementations have the same computational cost.

The BP equations for this problem read

$$\begin{aligned}
 p_{i \rightarrow j}(\pi_i) &= e^{\beta h_i \pi_i} \prod_{k \neq j} m_{k \rightarrow i}(\pi_i) \\
 m_{i \rightarrow j}(\pi_j) &= \sum_{\pi_i \neq \pi_j} e^{\beta s_{ij} s'_{\pi_i \pi_j}} p_{i \rightarrow j}(\pi_i),
 \end{aligned} \tag{3.1.6}$$

where the equalities hold up to some normalization constant. The solution of these equations can be implemented by exchanging a vector of dimension  $N'$  along all the links in the factor graph requiring a time that scales polynomially with the system size  $O(N^2 N'^2)$ . In the  $\beta \rightarrow \infty$  limit, by assuming the

unicity of the zero-energy solution, the previous equations further simplify to

$$\psi_{i \rightarrow j}(\pi_i) = h_{i\pi_i} + \sum_{k/j} \phi_{k \rightarrow i}(\pi_i) \quad (3.1.7)$$

$$\phi_{i \rightarrow j}(\pi_j) = \max_{\pi_i \neq \pi_j} \left( s_{ij} s'_{\pi_i \pi_j} + \psi_{i \rightarrow j}(\pi_i) \right), \quad (3.1.8)$$

where  $\psi_{i \rightarrow j}(\pi_i) = \beta^{-1} \log p_{i \rightarrow j}(\pi_i)$  and  $\phi_{i \rightarrow j}(\pi_j) = \beta^{-1} \log m_{i \rightarrow j}(\pi_j)$ .

This is true in weighted graphs where it is very unlikely to have two different configurations of the same weight but we will see that it is not possible to assume it for non-weighted graphs, where the solutions can be highly degenerate such as the case of SI problem.

In the rest of the chapter we test the performance of the algorithm by applying it to the problem of the maximum clique. This is defined as the problem of identifying the maximum order of fully connected subgraph  $cl(G)$  present in a graph  $G$ .

We show first how the method correctly predicts the clique number that is self-averaging in the Gilbert ensemble.

## 3.2 The Maximum Clique Problem

In this section we show how the algorithm performs when looking for a particular kind of subgraphs, the cliques, in graphs belonging to the Gilbert ensemble. The cliques are fully connected non weighted graphs  $g(N, E)$  whose adjacency matrix reads  $a_{ij} = 1 \forall i, j \neq i$ . As it is well known the clique problem is a NP-complete problem[44] and difficult to approximate [13]. Moreover, identifying the fully connected subgraphs in a large network is interesting because reveals some important features of the network itself, i.e. the denser parts, the underlying community structures and so on. The algorithm we present is devoted to the search of cliques but, working at finite temperature, allows for non-perfect matching, thus providing the identification of the dense subgraphs.

We now specialize the GA method to the search of clique  $g(N, E)$ , where  $E$  is the set of edges, and then we use it to identify the maximum clique (MCP) present in the graph  $G \in \mathcal{G}_{N,p}$ . Searching for a clique can be rephrased in looking for a permutations  $i \rightarrow \pi_i$  and  $j \rightarrow \pi_j$  that satisfies the local adjacency relations. This means that the energy counts the number of conserved links of the permutation. In this regime the BP equation (3.1.6) reduces

$$m_{i \rightarrow j}(\pi_j) = C_{i \rightarrow j}(\pi_j) + e^{-\beta} D_{i \rightarrow j}(\pi_j) \quad (3.2.1)$$

where  $C_{i \rightarrow j}(\pi') = \sum_{\pi \in N(\pi')} p_{i \rightarrow j}(\pi)$  and  $D_{i \rightarrow j}(\pi') = 1 - C_{i \rightarrow j}(\pi') - p_{i \rightarrow j}(\pi')$ . In that case, the complexity simplifies remarkably when the graph  $G$  is sparse because this new updating scheme requires a time that scales as  $O(N^2(N' + M'))$ .

Performing the zero temperature limit allows to identify precisely all the possible cliques embedded in  $G$ . Since the cliques can be more than one and there is a high degeneracy due to the symmetry of the clique itself, we have to be very careful in doing it correctly. We are not allowed to select only the maximum value over the possible value of  $p_{i \rightarrow j}$ , as done since now, but take into account the degeneracy of this state so that the (3.2.1) becomes

$$m_{i \rightarrow j}(\pi_j) = C_{i \rightarrow j}(\pi_j). \quad (3.2.2)$$

In terms of the solution of the previous equation, we obtain the number of cliques present in  $G$  through the entropy

$$S_{BP} = - \sum_{ij > i} \sum_{\pi_i, \pi_j} b_{ij}(\pi_i, \pi_j) \ln b_{ij}(\pi_i, \pi_j) + \sum_i (N-2) \sum_{\pi_i} b_i(\pi_i) \ln b_i(\pi_i), \quad (3.2.3)$$

where the beliefs  $b_{ij}(\pi_i, \pi_j)$  and  $b_i(\pi_i)$  are defined in equations (1.2.5) and (1.2.6). The entropy is related to the number of solution through the following condition

$$S_{BP} = \log Y_N - \log N! \quad (3.2.4)$$

where  $Y_N$  is the number of different cliques in the graph and  $N!$  counts the degeneracy of each clique due to the internal symmetry of the permutation. Numerical results for  $S_{BP}$  as a function of  $N'$  for the Gilbert ensemble, are shown in figure 3.4.

At this point, it is useful to spend some more words on the high symmetry which is intrinsic of the clique topology and see how it can be exploited to further reduce the complexity of the algorithm <sup>1</sup>. Indeed we notice that the cavity probabilities and the messages are equal for each possible couple, since all the edges are identical. This can be formalized assuming that the quantities  $p_{i \rightarrow j}(\pi) = p^c(\pi)$  and  $m_{i \rightarrow j}(\pi) = m^c(\pi)$  are constants for each couple  $ij$ . Therefore BP equations naturally follow

$$\begin{aligned} m^c(\pi) &= \sum_{\pi' \in N(\pi)} p^c(\pi') \\ p^c(\pi) &= m^c(\pi)^{N-1}, \end{aligned} \quad (3.2.5)$$

up to some normalization factor. This is a remarkable reduction of the algorithmic time  $O(N' + M')$  and shows very good results in the entropic

<sup>1</sup>We thank the referee B for useful discussion.

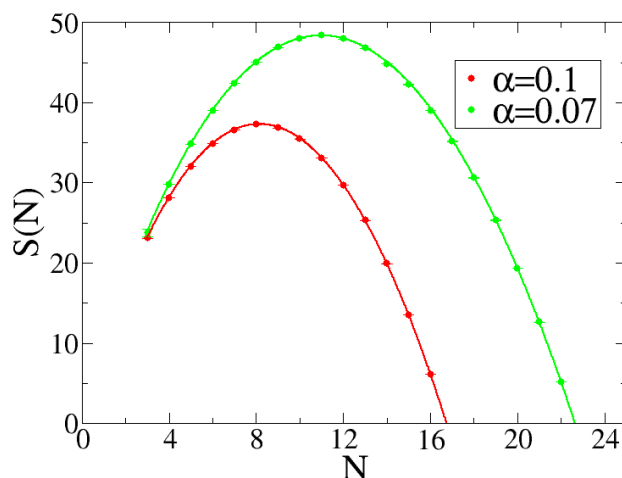


Figure 3.3: In figure is reported the BP entropy as a function of the clique size  $N$  for  $N' = 10000$ . This calculation has been obtained using equations 3.2.5. The solid lines are the theoretical prediction computed by means of  $\langle Y_N \rangle$  in 3.2.9. The numerical points are averaged over 10 realization and show a excellent agreement with the theoretical prediction.

estimation, as shown in figure 3.3 for numerical calculation in the Gilbert ensemble. Unfortunately we experienced some problems of convergence that start to be intractable for higher order of the clique  $N \sim N_0$ . This may be caused by the fact that this highly symmetric solution is dynamically unstable and in order to converge to a fixed point, we need to break the symmetry of problem.

Before presenting BP results for the Gilbert graph ensemble  $G \in \mathcal{G}_{N',p}$  for  $p \propto N'^{-\alpha}$  where  $\alpha \in [0, 1]$ , it is worth to briefly review the main theoretical bounds that has been obtained by Bollobas[12]. In particular we focus our attention on the maximum order of the clique present in the graph  $G$  namely the clique number  $cl(G)$ .

### Some theoretical results

Let  $G(N', E')$  be a graph realization of  $\mathcal{G}_{N',p}$ , where  $N'$  are the number of nodes,  $E'$  be the set of edges and the probability to have a link scales as  $p = N'^{-\alpha}$ .  $g(N, E)$  is called a subgraph of  $G$ ,  $g \subseteq G$  if its vertex set  $V(g) \subseteq V(G)$  and its edges  $E \subseteq E'$ . We will denote by  $\mathcal{K}(G)$  the set of

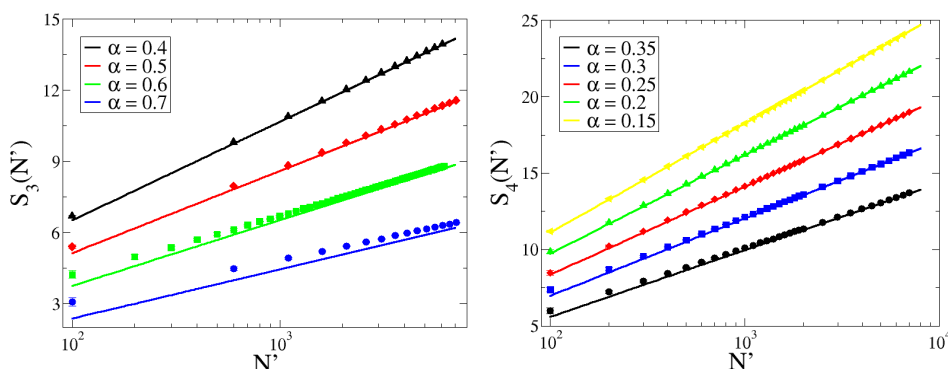


Figure 3.4: In the left (right) panel of figure we show the entropy for cliques of order  $N=3$  ( $N=4$ ), as a function of  $N'$  for various  $\alpha = -\log p/N'$ . Symbols are BP results (each averaged over 10 realizations), full lines give the logarithm of the expected number of cliques given in equation (3.2.9). These graphs show that the theoretical predictions are asymptotically approached for  $N' \rightarrow \infty$ .

cliques of  $G$  and with  $\mathcal{K}_M(G)$  the set of maximum cliques in  $G$ :

$$\mathcal{K}_M(G) := \{g \in \mathcal{K}(G) : |V(g)| = \max_{g' \in \mathcal{K}(G)} |V(g')|\}. \quad (3.2.6)$$

Of course the maximum clique is bounded by  $N'$  when  $G(N'E')$  is a fully connected graph ( $p = 1$  and  $\alpha = 0$ ). If the graph  $G \in \mathcal{G}_{N',p}$ , its clique sequence is almost entirely determined and its clique number has a narrow bound

$$N_0 - 2 \frac{\log \log N'}{\log N'} \leq cl(G) \leq N_0 + 2 \frac{\log \log N'}{\log N'} \quad (3.2.7)$$

with

$$N_0(N') = \frac{2}{\alpha} + 2 \frac{\log \alpha}{\alpha \log N'} + 2 \frac{\log(e/2)}{\alpha \log N'} + 1 + o(1). \quad (3.2.8)$$

This formula can be verified computing the number of cliques  $Y_N(G) = |\mathcal{K}(G)|$  of a given size  $N$ . Clearly its expected value averaged in the ensemble is

$$\langle Y_N \rangle = \binom{N'}{N} p^{\frac{N(N-1)}{2}} \quad (3.2.9)$$

therefore, for small value of  $N < N_0$  the expectation value  $\langle Y_N \rangle > 1$  and, increasing  $N$ , the expectation value  $\langle Y_{N_0} \rangle = 1$  and then drops below 1 rather suddenly, thus identifying  $N_0$  with the maximum order of the clique.

As demonstrated in [12], the probability the clique number deviates substantially from the typical value  $N_0$  vanishes with  $N'$ . Firstly, for  $N'$  sufficiently large, the probability the clique number deviates from  $N_0$  of a factor

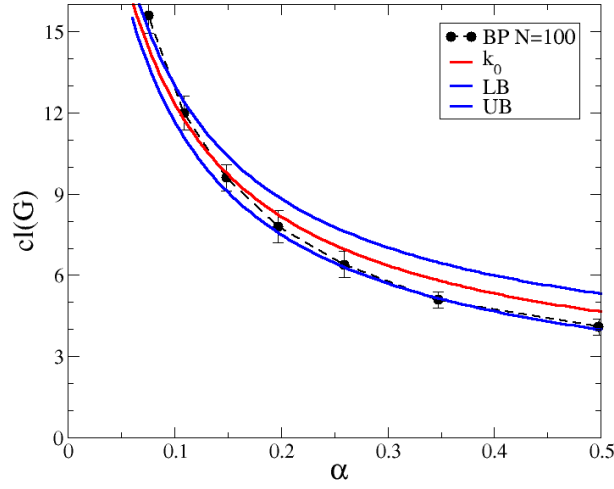


Figure 3.5: The graph in the left shows the clique number  $cl(G)$  for a graph of 100 nodes as a function of  $\alpha$  with reinforcement version of the equations (3.2.2). The reinforcement parameter is selected to be  $\lambda = 0.9999$  and the algorithm converges in almost 10000 iterations with a average time .... The blue lines are the theoretical bounds  $N_0 \pm 2 \log \log N' / \log N'$  while the red one represent the theoretical value  $N_0$ . The points are the BP solution found with reinforcement technique, averaged over ten different instances. Results are consistent with the theoretical bounds and demonstrate the clique number is self-averaging.

$\delta > 0$   $cl(G) = N_0 + \delta$  is dominated by

$$P\{cl(G) \geq N_0\} \leq \langle Y_N \rangle < N'^{-\delta}. \quad (3.2.10)$$

As well as the probability that clique number can be smaller than  $N_0$   $cl(G) = N_0 - \delta$  for  $0 < \delta < 2$ , is upper bounded by

$$P\{cl(G) \leq N_0 - \delta\} < \frac{2}{\xi^\delta} N'^{-\delta}, \quad (3.2.11)$$

where  $\xi$  does not depend on  $N'$  and is positive. This demonstrates that both the probabilities vanish with the size of the graph  $N'$  and deviations from the value  $cl(G) = N_0$  become exponentially rare with  $N'$  as shown in figure 3.5.

Despite the fact that the number of clique has such a small variability it is well known that large cliques of a random graph are very difficult to identify. This is due to the fact that there is a large number of cliques of

size  $N_0/2$ , but it decreases very rapidly for  $N > N_0/2$ . In particular for size of order  $(1 + \epsilon)N_0/2$  with  $\epsilon > 0$  their number is of order of  $N'^{-g(\epsilon)\log(N)}$  and hence is more than polynomially small[39]

We found numerically that the quenched ensemble averages  $\langle \log Y_N \rangle$  are well estimated by the annealed calculations  $\log \langle Y_N \rangle$ . In formulas this can be shown, after having introduced  $\delta Y_N = (Y_N - \langle Y_N \rangle) / \langle Y_N \rangle$ <sup>2</sup>, by the following relation

$$\langle \log Y_N \rangle = \log \langle Y_N \rangle + \sum_n (-1)^{n-1} \langle \delta Y_N^n \rangle.$$

This means that if the distribution of  $Y_N$  is peaked around its mean value, in the thermodynamic limit, we obtain that  $\langle \delta Y_N^n \rangle = [Y_N - \langle Y_N \rangle]^n / \langle Y_N \rangle^n \rightarrow 0$ . So that the  $\log \langle Y_N \rangle$  is directly comparable to the BP entropy obtained by simulation through equation (3.2.4). The good numerical agreement of the entropy is shown in both the figures 3.4 and 3.3 for all the range of  $N$  and  $N'$ , even for small number of averages, meaning the number of clique are well-behaving quantities in the Gilbert ensemble and the higher order corrections are negligible also for finite size systems.

### Discussion

The comparison between BP calculations and theoretical predictions are shown in figures 3.3 and 3.4. The first shows the entropy computed with (3.2.2) as a function of the size of the graph  $N'$  while the latter refers to entropy versus  $N$ , computed using the symmetric ansatz (3.2.5). Solid lines give the theoretical predictions namely the logarithm of the expected number of cliques  $\log \langle Y_N \rangle$  while the numerical points are related to numerical calculation of the entropy  $S_{BP}$ . For small  $N$ , the asymptotic regime  $N' \rightarrow \infty$  is well captured by the BP computation, giving reliability of the cavity method. Nevertheless, by increasing the size of the clique  $N$ , we found some problem in the convergence of the symmetric algorithm.

We also obtain the clique number reported in figure 3.5. We explicitly find the maximum clique in network by a reinforcement method, looking for larger clique until the algorithm does not find any solutions with reinforcement parameter<sup>3</sup>  $\gamma_0 = 0.9999$ . This algorithm shows good property of convergence and find similar results as decimation scheme but need for less amount of time. The numerical value are well within the theoretical bounds showing good performance of the method.

<sup>2</sup>in the regime  $0 < \delta Y_N < 2$

<sup>3</sup>The reinforcement description is introduced in section 1.3 where we introduce the quantity  $\gamma_t$  by means of a real external parameter  $\gamma_0$





## Chapter 4

# Biological Applications

The algorithm we have presented in the previous chapter has been tested in a well known academic problem, the maximum clique. From now on we provide, as a proof-of-concept, two biological applications where our method can be used. The first one refers to the sensory-response system while the second aims at analyzing large biological networks of interactions among biomolecules. There are many other problems where our algorithm may be applied, the protein structural alignment being one of them.

The cell, in order to survive, has to perform several actions in particular they have to respond to external stimuli. The mechanism through which the cell responds to environment conditions is called sensory-response system. The most common example of sensory-response system in bacteria is the two-components systems (TCS) which is found on the interaction specificity between two different families of proteins[91, 51]. The TCS has been deeply analyzed in the last decade at molecular level [97, 35]. Despite the main advance in this field, however, there are many open questions regarding the regulation within individual pathways. The system-specific mechanism through which a stimulus is associated to adaptive response is still unclear and matter of research [82, 17].

The set of all these connections within the cell defines large networks where nodes represent biological components and edges the physical or chemical interactions among them. From one side, we can be interested in understanding the basic mechanism behind each of them on a general ground, and from the other side we may be focused to analyze the network as a whole in order to explain more complex function the cell is able to perform[67, 9]. From this perspective the topology of the networks is relevant in clarifying its functionality and is crucial in understanding the key role played by specific nodes. This analysis is of particular relevance in systems with a large

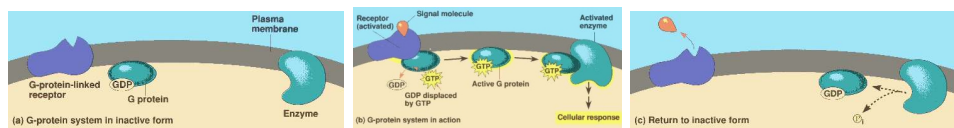


Figure 4.1: We summarize the main moments in the sensory-response systems of one of the main mechanism for signaling in organisms. A family of proteins involved in transmitting chemical signals outside the cell, and causing changes inside the cell is called G-protein. This family of proteins is composed by an extracellular part called G-protein-linked receptor bounds to the cell membrane and the intracellular G-protein inside the cell. As soon as a signal molecule binds to the G-protein-linked receptor, the receptor inside the cell activates a G protein by causing a conformational change. G protein complexes bind to phosphate groups (GDP/GTP) that turns on the molecule when is in its three phosphate state. This change results in the action of another protein usually an enzymes altering its activity. After the active part, when the signal is no more present the GDP group of the receptor dephosphorylates and turns back to its inactive state.

number of degrees of freedom for which a large amount of data is available and –at the same time– a set of clear theoretical models is still lacking. Taking inspiration from this approach, we show how the algorithm performs in the searching of specific pattern of nodes in real directed networks.

## 4.1 Finding protein partners from multi-species sequence data

The study of sensory-response systems has defined the basics of how many organisms detect and respond with sensitivity to changes in their chemical or physical environments. Such studies have recently focused on events that occur at the cellular and molecular levels, elucidating the mechanisms of detecting extracellular signals and transducing such signals into the appropriate intracellular events. The two-component signal transduction (TCS) is the most prominent example of sensory-response system in bacteria[97, 35]. The signal is transduced from a *histidine sensor kinase* (SK) to a *response regulator* (RR), which in its phosphorylated form becomes in most cases an activated transcription factor. The SK, which is regulated by environmental stimuli, autophosphorylates at a histidine residue (HK-His) using ATP present in the cell, creating a high-energy phosphoryl group (P). The P

is then subsequently transferred to an aspartate (ASP) residue in the RR protein. Phosphorylation induces a conformational change in the regulatory domain that results in the activation of an associated domain effecting the response. This interaction forms the central part of signal transduction mechanism.

In figure 4.1, it is summarized three different chemical moments.

- Autophosphorylation of the histidine residues
- Phosphotransfer from HK to RR
- Dephosphorylation of the aspartate residue of the RR

The signal-binding site is outside the cell while the histidine kinase sensor site is inside the cell. When signal molecules binds outside the cell to the specific HK, the histidine receptor is activated, as shown is the center of the figure, when a P group binds to the HK-His (autophosphorylation). Then it displaces (phosphotransfer) to the RR protein, usually a transcription factor, altering its activity and starting the response. This activity is usually temporary because RR hydrolyzes the phosphate P deactivating the RR.

Despite the fundamental importance of protein-protein interactions in most biological processes, identifying interaction partners between SK and RR is experimentally and computationally a major problem. Each bacterium contains  $\mathcal{O}(10)$  interacting SK/RR pairs forming different TCS pathways. The necessity to trigger the correct answer for each specific extracellular signal forbids crosstalk between pathways. So, even if all different SK and respectively RR in one species are structurally and functionally similar, only specific samples of these two *protein families* interact.

Our question here is, if GA algorithm can help in recognizing interaction partners. We start from a large collection of SK sequences extracted from hundreds of bacterial genomes, and a second large collection of RR sequences coming from the same bacteria, and we aim at extracting interacting SK/RR pairs, exploiting sequence similarities of proteins inside each family. Proteins inside each family are homologous: they show strong structural and functional similarity, but also a considerable amount of sequence variability between species. To maintain function, this variability cannot be random: imagine, e.g., two interacting proteins, and a random mutation occurring in the interaction surface of one of them. It is likely to have a deleterious effect on the affinity between the two proteins, but it might be compensated for by a mutation in the other protein. This mechanism introduces a co-evolutionary coupling of interacting proteins, and therefore the similarity networks of two interacting protein families are expected to be

similar.

So that the basic idea is simple: two SK with very similar amino-acid sequences will (due to their probably recent common evolutionary origin) interact with two similar RR. Globally spoken, an alignment of two similarity networks - one for the SK family, one for the RR family - might be able to pair a large fraction of all those SK and RR which actually belong to common TCS pathways [84].

Our data set consists of two multiple-sequence alignments (with gaps) for 2546 SK and 2546 RR proteins from 231 genomes [96]. They are selected such that, due to the frequent coding of an entire TCS in one operon, the correct mapping is known, and can be used *a posteriori* to verify our GA results. Similarity networks for each protein family are constructed as  $k$ NN graphs: Each protein is linked to the  $k$  most similar proteins, where similarity is measured via the Hamming distance  $d_{ij}$  between the aligned aminoacid sequences of two proteins  $i$  and  $j$ . The link weight is given as

$$s_{ij} = \exp[-d_{ij}^2/d_k^2], \quad (4.1.1)$$

with  $d_k$  being the average distance between each protein and its  $k$ th neighbor. One might use more sophisticated distance measures (e.g. alignment scores), but due to the proof-of-concept character of this application we have chosen the simplest possible measure. To identify interaction partners, we must align only proteins inside the same species, formally this is implemented by imposing  $h_{i\pi} = -\infty$  for all  $i$  and  $\pi$  belonging to different species. Finally, we have also introduced various amounts of information about real interaction partners, by randomly introducing positive similarities between a number of actual interaction partners (training set). Summarizing we have three different choices for the external field

$$h_{i\pi} = \begin{cases} -\infty & \pi \notin \mathcal{M}(i) \\ 0 & i \in \mathcal{T} \\ \omega_i & \text{else} \end{cases} \quad (4.1.2)$$

where  $\mathcal{M}(i)$  is the set of all possible images of  $i$  namely all the RR's belonging to the same species of  $i$  whereas  $\mathcal{T}$  defines the set of selected SK proteins forced to be mapped to the real partner. All the other couples have a field proportional to a small random noise with fixed average value  $\bar{\omega}_i \ll \min_{ij} s_{ij}$  that helps the convergence of the algorithm.

The results are summarized in the following table for different  $k$  and training-set sizes  $T = |\mathcal{T}|$ . Error bars result from an average over different random training sets. The values display the fraction of correctly aligned protein pairs in between all proteins not being in the training set.

T	3NN, $k = 3$	6NN, $k = 6$	9NN, $k = 9$
2000	$88.7 \pm 1.7$	$89.8 \pm 1.9$	$90.5 \pm 1.7$
1000	$76.2 \pm 1.3$	$78.7 \pm 1.0$	$79.6 \pm 0.8$
500	$67.4 \pm 1.9$	$73.1 \pm 1.4$	$75.0 \pm 1.0$
0	48.1	58.9	64.7

We note that even without training set,  $T = 0$ , almost 65% of all proteins are correctly matched (for  $k = 9$ ). This number has to be compared to a random matching, where only  $231/2546 \sim 9\%$  correct matchings would be expected. The introduction of a training set improves strongly the performance, for a training set of 2000 protein pairs, about 90% of the remaining 546 proteins are correctly aligned. These results beautifully demonstrate that the original idea to exploit sequence similarity of proteins across species is actually providing information about who is interacting with whom. Work is in progress in applications which require to incorporate refined biological priors into the algorithm.

## 4.2 Motifs in biological network

Cells have to carry out numerous functions to survive, ranging from replication and energy conversion to molecule transport and signalling –used in cellular communication. Many of these functions require complicated cascades of reactions between proteins, DNA, metabolites, that have been revealed by means of new experimental techniques like i.e. mass spectroscopy [34], genome-wide chromatin immunoprecipitation [37], yeast two-hybrid assays [94].

We will focus on protein like i.e. transcription factors (TF) and gene (G), by modelling the set of interconnections between them as abstract graphs or networks, in which nodes represent bio-molecules and edges describe the physical and chemical reaction among them (see figure for a brief explanation of the main interaction between TF and genes 4.2). This network represent a dynamical system and are designed with strong separation of timescales: the input signal takes sub-second to change the TF activities. Then binding to DNA sites requires several seconds and finally to see protein product can take minutes or hours.

The connection between biomolecules can be either undirected if the relation is symmetric or directed, if one molecule regulates the other but the contrary is not true. Although the details of the physical connections in in-

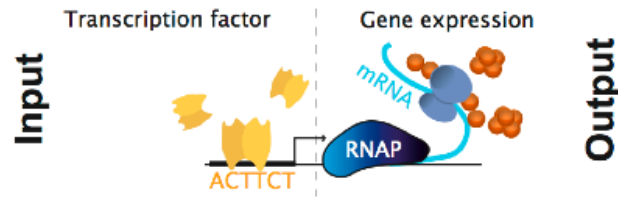


Figure 4.2: The cartoon explains the interaction between TF and genes. Each gene is usually preceded by a regulatory DNA region called the promoter, where the activator transcription factor binds due to the high affinity with specific sequence sites. This binding increases, or decreases in several cases, the probability the RNA polymerase (RNAP) start syntetizing the mRNA and produce the gene product.

teractions is clearly lost, this large amount of information can be exploited to describe the functionality of the cell at a coarser scale and the key role played by each biological component in the whole system.

Recent works indicate that biological networks show recurrent small patterns, pointed out as basic modules of molecular information processing, called *network motifs*. These are collections of nodes interacting in a specific manner, that occur much more frequently than what would be expected by chance [67, 9], suggesting that their presence might be responsible of some function performed by the network. The basic idea behind this approach is that the main functions performed by living organism can be best described by interactions between modules rather than between single elements. The importance of these motifs as information-processing modules has been justified theoretically [86, 54] and verified experimentally [43, 55].

Since transcription network is highly selected by evolution –a regulatory interaction can be modified just by few point mutations on the genome– the overrepresentation of interaction patterns can be the outcome of a stronger selection pressure because of some beneficial effects they perform on the organism.

Some examples are shown in figure 4.3 in which we show two different kind of motifs that perform specific function in the network. First of all the feed forward loop that is highly represented in all the studied networks. This circuit is composed by two transcription factors,  $X$  and  $Y$ , and a target gene  $Z$ . The FFL has two parallel paths: a direct path from transcription factor  $X$  to the gene  $Z$  and an undirected path through transcription factor  $Y$ . It is responsible to a slow response to activation signal and a fast response to repression signal as it is described in [2]. The second example of known

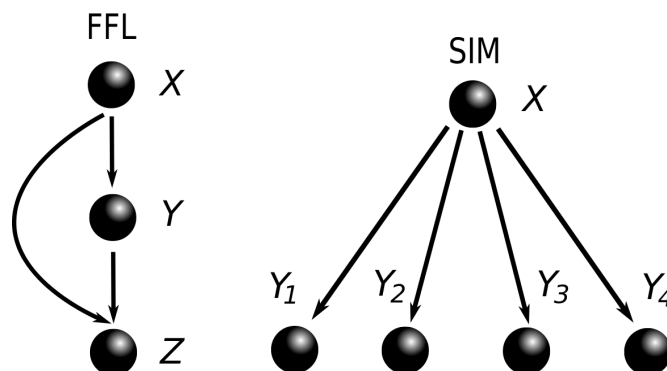


Figure 4.3: Two motif examples: feed forward loop (left) and single input module (right). The feed forward loop (FFL), depending on the sign of the interaction could be coherent or incoherent with different possible functionality. In the coherent case the loop provides a pulse filtration activating a response only to persistent signal. On the contrary the incoherent loop is a pulse generator and a response accelerator. In the section Methods we describe how to take into consideration the sign on the edge. The second motif is called single-input module (SIM) and occurs when a single molecule regulates a set of genes with no additional regulation. This is useful when a set of genes must work in a synchronized manner.

functional motif is the single-input module, in which one factor  $X$  regulates a group of target genes. The main function of this pattern is to coordinate expression of several genes  $Y_1, Y_2$  etc. with shared function. In particular it can generate temporal expression programme in which each gene has a defined order of activation[2].

Moreover network motifs provide an important tool for understanding the modularity and the large-scale structure of the network. Thus identifying these patterns and describing their dynamical function could be useful to understand the design principles underlying the mechanism that sustain cellular functions.

Unfortunately, this analysis is hampered by the limited size of motifs handled by current methods. For example, in [67] exact counting have only been reported up to size 4 motifs and in [46] a certain type of motif generalizations up to 6 nodes. Efficient algorithms are known for specific classes of subgraphs, such as cycles and cliques reviewed in [11], while several approximating methods have been developed, most of them based on sampling techniques, for finding Hamiltonian cycles and spanning trees [40, 29, 57]. Depending on the graph being analyzed, the run-time of most of these algo-

rithms grows very fast with the size of the subgraph thus making impossible to detect the presence of larger network motifs.

This limitation leaves several important problem unsolved and new insight could be gained by exploring larger subgraphs. As a matter of fact finding motif of bigger size is useful to better describe the topology of the network because smaller subgraphs can change their functionality if put in interaction with other neighboring nodes in a specific manner [46, 6]. From the algorithmic viewpoint, this task remains a major challenge mainly because of the computational complexity of the algorithm. In this sense it is worth noticing the works going in this direction of Alon et al [45] in which they introduce a sampling algorithm for finding  $N$ -nodes subgraphs. Kellis et al [33] proposed a new method that assesses the significance of a single query subgraph and then apply it to all the possible  $N$  size subgraphs using a symmetry breaking technique.

First of all we describe how specializing the GA algorithm. Then we discuss the results and the run-time of the algorithm and finally we introduce some possible generalizations.

#### 4.2.1 Algorithm

Two basic problems have to be faced in order to identify motif in large networks. The first is that the number of different subgraphs increase exponentially with the size of the subgraphs i.e. there are 13 different 3-nodes subgraphs, 199 4-nodes subgraphs, 9364 5-nodes subgraphs and so on. Moreover, the subgraph isomorphism problem is NP-complete[33].

The problem is defined as follows: we are interesting in finding a directed subgraph  $g(N, E)$  of  $N$  nodes and  $E$  arcs into a new directed graph  $G(N', E')$  of normally much larger order  $N' > N$ . A directed graph is a graph whose edges have a unique direction, meaning that the relationship between vertices is not symmetric. They are contained in the adjacency matrix  $A$ , whose entries  $a_{ij}$  are equal one if there is an edge going from one node  $i$  to  $j$  or zero otherwise. Of course, in case of directed graphs, this matrix is not symmetric. After having introduced the mapping  $\pi$  we define the energy function as

$$\mathcal{H} = \frac{1}{N} \sum_{i,j \in V(g)} (1 - \delta_{a_{ij}, a'_{\pi_i \pi_j}}), \quad (4.2.1)$$

where  $A$  and  $A'$  are respectively the adjacency matrices of  $g(N, E)$  and  $G(N', E')$ . It is straightforward to verify that 0 energy is related to the isomorphic sub-matching between the subgraph  $g$  and the graph  $G$ , while



on the other hand configurations with edge mismatches would correspond to higher energy values. This means that finding the ground state of  $\mathcal{H}$  is equivalent to identify subgraph  $g(N, E)$  in the graph  $G(N', E')$ . Moreover, as usual, we directly might enforce the injectivity, so that the constraint factor reads

$$f_{ij}(\pi_i \pi_j) = (1 - \delta_{\pi_i \pi_j}) \left[ (1 - e^{-\beta'}) e_{ij} e_{ji} + e^{-\beta'} \right], \quad (4.2.2)$$

where  $e_{ij} = \delta_{a_{ij}, a'_{\pi_i, \pi_j}}$  and  $\beta' = \beta/N$ .

We obtain two coupled equations, called belief propagation (BP) equations, for the cavity marginal probability and the messages  $\phi_{j \rightarrow i}$ 's

$$p_{k \rightarrow j}(\pi_j) = \prod_{i \in V(g)/j} m_{i \rightarrow j}(\pi_j) \quad (4.2.3)$$

$$m_{i \rightarrow j}(\pi_j) = \sum_{\pi_i: f_{ij} \neq 0} p_{i \rightarrow j}(\pi_i), \quad (4.2.4)$$

In term of the beliefs we can compute the entropy so to obtain the BP approximated value of the logarithm of the number of subgraphs

$$S_{BP}(\beta) = - \sum_{ij\pi\pi'} b_{ij}(\pi, \pi') \log b_{ij}(\pi, \pi') + \sum_{i\pi} (N-2) b_i(\pi) \log b_i(\pi) \quad (4.2.5)$$

As a trivial limit, we choose the subgraph  $g(N, E)$  to be composed of two node and a single directed edge. The entropy results to be equivalent to  $S = \log E'$  and is exactly computed by BP simulation. In this case, in fact, creating a cavity, the correlation vanishes and BP assumptions are correct.

In order to obtain the ground state configuration we perform the  $\beta \rightarrow \infty$  limit, obtaining the BP equations (4.2.3)(4.2.4) further reduce to

$$p_{k \rightarrow j}(\pi_j) = \prod_{i \in K/j} \phi_{i \rightarrow j}(\pi_j) \quad (4.2.6)$$

$$m_{i \rightarrow j}(\pi_j) = \sum_{\substack{\pi_i: a_{ij} = a'_{\pi_i, \pi_j} \\ a_{ji} = a'_{\pi_j, \pi_i}}} p_{i \rightarrow j}(\pi_i). \quad (4.2.7)$$

We apply this method in transcription regulation networks. These networks give a coarse-grained description of interaction between transcription factor proteins that regulate the expressions of several genes or other transcription factors. The transcription of a general is the process by which the specific protein RNA polymerase, produce a mRNA that corresponds to a particular gene sequence. This mRNA codes the information for the production of a new protein called gene product in the process 4.2.

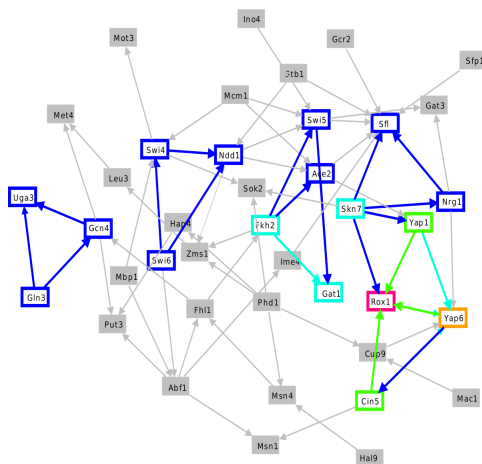


Figure 4.4: We show part of the transcription regulation network of *S. Cerevisiae* and all the possible feed forward loops contained in it. The selected edges of FFL are emphasized with bold line. This plot has been done using the program MAVISTO[85]. It is possible to recognize nodes and edges involved in more than one FFL with different colour: dark blue, blue, green and orange, if they are respectively part of two, three or four FFL's.

Here we study the transcriptional regulation networks of two different organisms *E. Coli* and *S. Cerevisiae* (see figure 4.4). While the first is described in [86] the latter is based on the YPD database [19]. Some recent works point out that these networks can be split into small patterns made up of several nodes connected in a given way. BP algorithm is able to identify subgraphs of a given shape embedded in these genetic networks and gives a good estimation on the number of times it appears in them computing the entropy (4.2.5). In order to quantify the correctness of our algorithm we define the relative error for each query subgraph and see what is the fraction of samples that are correctly predicted with BP algorithm. Since the number of subgraphs is an integer, we compute the relative error for the entropy estimation as

$$\Delta S = \frac{|S_{BP} - \log(\mathcal{N})|}{\log(\mathcal{N} + 1)} \quad (4.2.8)$$

where  $\mathcal{N}$  is the exact number of times a subgraph is embedded in the real network. This measure quantify correctly the number of estimated solution of BP algorithm and shows the reliability of the method proposed.

In this sense, this is a step-forward because allow us to look for subgraphs of larger size being much less expensive from the computational point of view. Numerical limit of exact algorithm comes out just from size of order

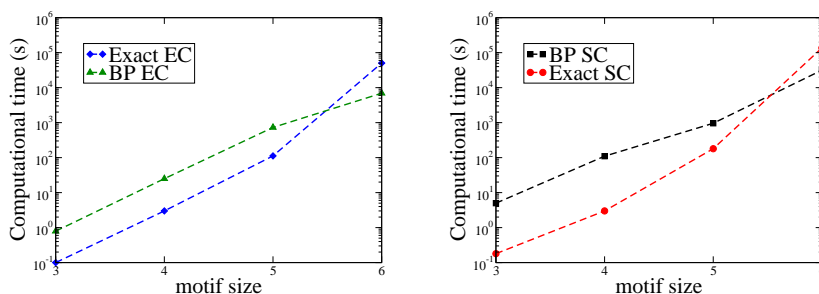


Figure 4.5: We show the computational complexity of the BP algorithm as a function of the run-time of exact enumeration. We compare this result with respect to exact enumeration method provide by Alon et al. in [67]. In both cases we gain one order of magnitude in the computation of 6-nodes subgraphs showing the limitation of exact method scaling behavior with respect to our technique.

$\geq 5$  therefore the properties of real networks are much more easy to access with this procedure. We show in figure 4.5 the run-time of our algorithm to find  $N$ -nodes subgraphs as a function of the size  $N$  in both the networks providing evidence of the different scaling behavior of BP method and exact enumeration.

## S. *Cerevisiae* network

Firstly we test the performance of the algorithm using the *S. Cerevisiae* transcription network. The network is constructed from data collected in YPD database [19] and is available on web page [1].

We show in figure 4.7 the entropy  $S_{BP}$  as a function of the logarithm of exact result obtained using the method proposed by Alon et al [67]. BP algorithm converges fast for almost the totality of samples present in the network estimating the correct entropy with respect to the expected values as shown in figure 4.7. The straight line outlines the perfect matching between the predicted entropy and the exact count. Despite the crude approximation, our numerical data are well distributed along this line demonstrating the good estimation of our method. Computing the relative error of our predicted value we obtain, as shown in figure 4.6 that almost 80% of the queries subgraph are predicted with an accuracy of 20% or less.

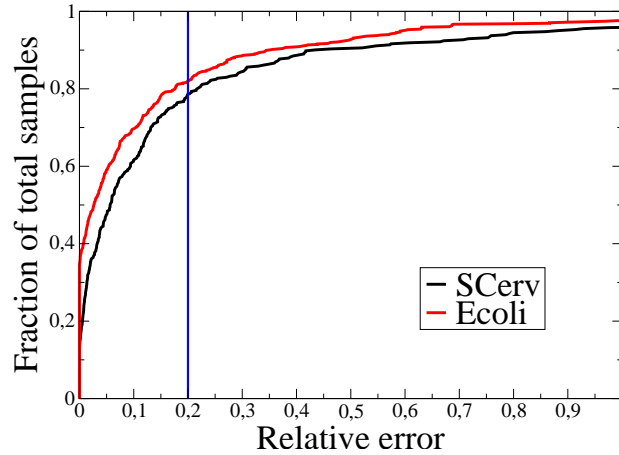


Figure 4.6: In figure we plot the density function of the relative error defined in equation (4.2.8) for both network *S. Cerevisiae* and *E. Coli*. It is clear that in both the network the 80% of the subgraphs are predicted with an accuracy of at least 20%. This provide a clear sign of the reliability of the method proposed for the estimation of the number of subgraph occurrence.

### E. Coli network

Secondly we use the *E. Coli* transcription regulation network. This network is base on RegulonDB database enhanced by several transcription factors finding in literature as described in [86]. It consists on 116 transcription factors and 419 operons involved in 577 interactions and available on [1]. We look for motif of size  $\leq 6$  and compare the results of our approximate algorithm with the correct one.

We could see as in the previous case a good performance of the BP algorithm that is able to estimate the number of occurring subgraphs. We report in figure 4.8 the entropy  $S_{BP}$  as a function of the logarithm of the exact number obtained with MFINDER [67] pointing out the perfect agreement with the red straight line. The number of samples for which the algorithm does not converge is a negligible fraction of the total and we obtain in the remaining cases that the 80% of the totality queries are correctly predicted with an accuracy of the 20% as shown in 4.6.

### 4.2.2 Discussion

We presented a novel approach to the discovery and counting of network motifs based on message passing technique. This algorithm can be generalized

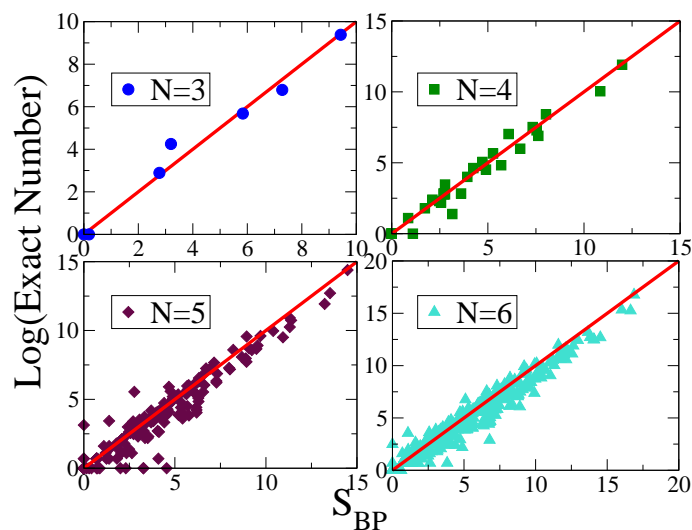


Figure 4.7: In the four panels we plot the logarithm of a given subgraph occurrence in the genetic network of *S. Cerevisiae*, as a function of entropy computed using BP algorithm. To each point is associated one directed subgraph given the number of nodes  $N$ . The entropy in the Bethe approximation regime correctly estimate the number of times the subgraphs is contained in the network. The straight line shows the best possible solution in which the two number are equal and it is possible to see the points accumulate in that region.

in many ways. In particular we might be interested in adding some information on the type of interaction between two biomolecules, for example the strength or reliability, or the sign of the interconnection. This is the case of transcription regulation networks where each transcription factor could activate or repress a response regulator, need for the introduction of two types of edges: + and -. In this networks identifying subgraphs with specific configuration of signs could be more indicative than identifying only pattern of nodes with specific interconnection. In transcription regulation network an important feature is captured by *frustrated closed subgraphs*. This is a set of subgraphs that have at least one close path with an odd number of negative edges like i.e. the incoherent feed forward loop that have completely different properties than the coherent one. It is possible to demonstrate these subgraphs might be liable to non-monotonic behavior [89] of the networks. This properties is supposed to be related to the stability towards external variability and could be used in order to understand deeply the functions carry out by the network itself.

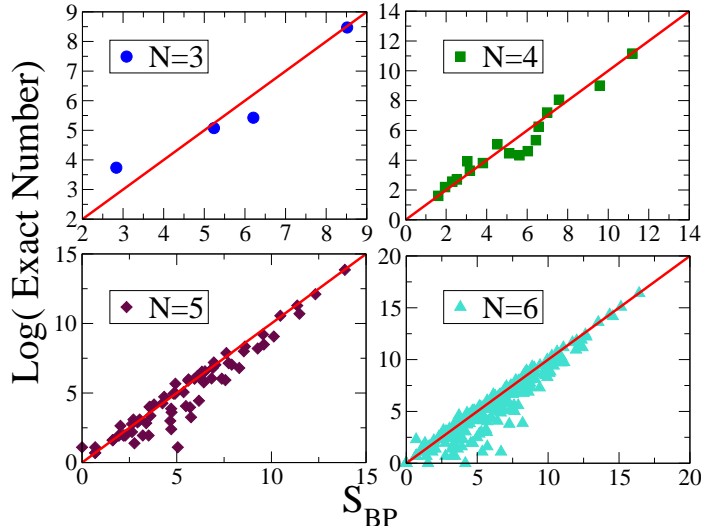


Figure 4.8: In the four panels we plot the logarithm of a given subgraph occurrence in the genetic network of *E. Coli*, as a function of BP entropy. To each point is associated one directed subgraph  $g(N, E)$  with a fixed number of nodes  $N$ . The entropy in the Bethe approximation regime correctly estimate the number of times the subgraphs is contained in the network. The straight line show the best possible solution in which the two number are equal and it is possible to see a good agreement between our method and the exact one.

The BP algorithm could analyze this kind of networks by employing a special mapping to a non-signed motif identification problem explained below. Starting a network  $G(N', E')$  with signs on edges  $c : E' \rightarrow \{-1, 1\}$ , we will define a new network  $G_s(N'_s, E'_s)$ . First, duplicate the nodes of  $G(N', E')$ , defining the set  $N'_s = V(G) \cup [N' + 1, \dots, 2N']$  and then apply the following rule to draw new edges: if there is a positive edge going from nodes  $i$  to  $j$  in  $G$ , add to  $E'_s$  two edges: one from  $i + N' \rightarrow j + N'$  and the other from  $i \rightarrow j$ . Otherwise if the edge from  $i$  to  $j$  in  $V(G)$  is negative, add the following two edges instead: one from  $i \rightarrow j + N'$  and from  $i + N' \rightarrow j$ . We could visualize this procedure as sketched in figure 4.9, adding one more plane. All the nodes  $i < N'$  are drawn in the upper plane while the other  $i > N'$  are contained in the lower plane. In that representation negative edges pass from one plane to the other while positive ones lie on the same plane. It is straightforward to see that frustrated loops live between the two sheets because passes from one plane to the other an even number of times. In figure 4.9 we show a frustrated loop in the network  $G$  with a blue shadow

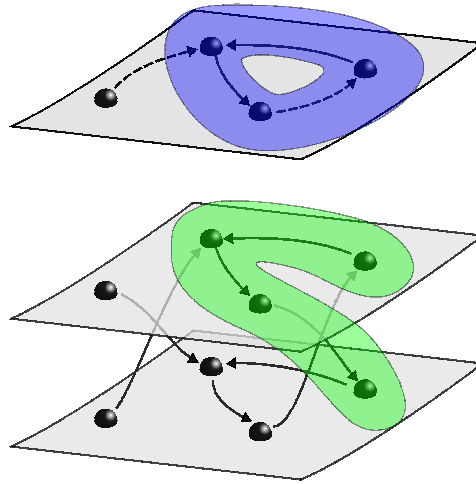


Figure 4.9: In the right panel the cartoon sketches an example of frustrated subgraph like a loop using the prescription defined in the article. After drawing a *twofold* network we construct an edge between nodes in the same plane ( $i \rightarrow j$  and  $i + N \rightarrow j + N$ ), if there is a positive edge - solid line - in the original network as showed on the first plane. Otherwise, if the original edge is negative - dashed line - we connect the duplicated nodes belonging to different plane ( $i \rightarrow j + N$  and  $i + N \rightarrow j$ ). A frustrated loop, pointed out in the cartoon with a blue shading, is equivalent to a path going from one node  $i$  to its duplicate  $i + N$  (green shading).

and its image on the new network  $G_s$  is outlined in green.





# Conclusions and Perspectives

In this thesis we have developed novel statistical physics inspired algorithms to efficiently solve computationally hard problems such as those arising from subgraph identification. These classes of problems find many interesting application in the field of computational biology to the large body of datasets which have recently become available.

From a theoretical perspective the problem of searching for graphs of a given shape is particularly challenging. This issue is enlightened in the cavity solution of the traveling salesman problem that we have sketched in chapter one. Using the standard argument á la De Gennes we showed how the global constraint is lost and the BP equations of the TSP become equivalent to that of the 2-matching problem, which aims at finding the optimal set of disconnected cycles.

The failure of this method suggests the need to enforce explicitly the global constraints and represents theoretical motivation for this work. Firstly we discussed the particular case where the peculiarity of the *connectivity* constraint can be rephrased in term of local quantities, by using an appropriate representation. The novelty of the method is based on the introduction of  $N$  new parameters  $d_i$ , interpreted as distances from a root node. This provided an algorithmic scheme that has been applied to high dimensional data clustering, giving new and interesting results.

The natural further step has been to generalize the method in order to deal with any possible shape of sub-graphs. This has been done introducing a global constraint that increases the computational costs but can be definitely used for the graph alignment problem (GA). This very general problem of aligning graphs such that they show maximum overlap, has many interesting special cases: maximum clique identification (MCI), traveling salesman problem (TSP) and subgraph isomorphism problem (SI). The MCI is a theoretical application where the algorithm shows very good performance and agreement with theoretical predictions.

However, it is worth to notice the long standing problem of TSP seems to

be inaccessible with all the different representations, showing that problem is intrinsically difficult and leading to more complicated scenario.

As a proof-of-concept we have used GA algorithm to predict interaction specificity within the cell sensory-response system. The idea was to exploit the tendency of two types of interacting proteins to have correlated evolutionary history. Results show good performance of the method proposed that is able to correctly predict almost the 65%-90% of the interactions depending on the biological information exploited.

The last application took its inspiration from the problem of identifying patterns of nodes overrepresented in a given network [67]. Most of the recent literature in this field is concentrated in identifying building blocks in the set of interactions devoted to perform specific functions. This is motivated by the fact that biological networks show modularity, simplicity and robustness to component tolerances. Inspired by this picture, the introduction of *motifs*, or recurrent pattern of nodes, tries to explain the mechanism behind the network functionality. Nevertheless, this basic idea need for the definition of *a priori* reference models, leading to ambiguous answers [4].

The method we proposed has many other possible application that can be analyzed, one of them being the protein structural overlap. It is devoted to the problem of aligning 3D structures of two or more different proteins, and defining a similarity score among them [66, 36]. Looking for 3D structural overlap between family of proteins is a major problem firstly because it is not easy to define a well defined objective function to minimize. Nevertheless, also measuring the statistical relevance of a given proposed alignment is another big open question. This application is currently being tested and will be the subject of a forthcoming publication.

# List of Publications

The results exposed in this thesis have been published in

- S. Bradde, A. Braunstein, H. Mahmoudi, F. Tria, M. Weigt and R. Zecchina  
Aligning graphs and finding substructures by message passing.  
*Europhys. Lett.*, **89**, 37009 (2010)
- M. Bailly-Bechet, S. Bradde, A. Braunstein, A. Flaxman, L. Foini and R. Zecchina  
Clustering with shallow trees  
*J. Stat. Mech.* P12010 (2009).

Other works published during the PhD:

- S. Bradde and G. Bianconi  
Percolation transition and distribution of connected components in generalized random network ensembles.  
*J. Phys. A: Math. and Theor.*, **42**, 195007 (2009).
- S. Bradde and G. Bianconi  
Percolation transition in correlated hypergraphs.  
*J. Stat. Mech.* P07028 (2009).
- S. Bradde, F. Caccioli, L. Dall'Asta and G. Bianconi  
Critical fluctuations in spatial complex networks  
*Phys. Rev. Lett.*, **104**, 218701 (2010).



# Aligning graphs and finding substructures by a cavity approach

S. BRADDE<sup>1,2</sup>, A. BRAUNSTEIN<sup>2</sup>, H. MAHMOUDI<sup>3</sup>, F. TRIA<sup>3</sup>, M. WEIGT<sup>3</sup> and R. ZECCHINA<sup>2(a)</sup>

<sup>1</sup> *International School for Advanced Studies (SISSA) - Via Beirut 2/4, I-34014 Trieste, Italy, EU*

<sup>2</sup> *Politecnico di Torino - Corso Duca degli Abruzzi 24, I-10129 Torino, Italy, EU*

<sup>3</sup> *Institute for Scientific Interchange - Viale Settimio Severo 65, I-10133 Torino, Italy, EU*

received 2 October 2009; accepted in final form 20 January 2010

published online 22 February 2010

PACS 75.10.Nr – Spin-glass and other random models

**Abstract** – We introduce a new distributed algorithm for aligning graphs or finding substructures within a given graph. It is based on the cavity method and is used to study the maximum-clique and the graph-alignment problems in random graphs. The algorithm allows to analyze large graphs and may find applications in fields such as computational biology. As a proof of concept we use our algorithm to align the similarity graphs of two interacting protein families involved in bacterial signal transduction, and to predict actually interacting protein partners between these families.

Copyright © EPLA, 2010

Over the last decade, the use of graphs for the description of relations between components of complex systems has become increasingly popular [1]. However, most part of the current literature concentrates on (computationally accessible) local characteristics like node degrees, whereas the full exploitation of more global properties of large networks remains frequently elusive due to their inherent algorithmic complexity. Most often studying global properties requires solving NP-hard problems or even harder problems if some form of uncertainty or lack of information is included in the definition of the problem. In both cases heuristic algorithms need to be developed. Specific examples, covered by this article, include the comparison of two different networks, the so-called *graph-alignment problem* (GA) [2–4], and the *sub-graph isomorphism* (SGI), as a particular case of which we consider the widely studied maximum-clique problem [5–7].

Recently, there has been a lot of interest in distributed algorithms to deal with optimization problems over networks. In the context of statistical physics a new generation of algorithms has been developed (*e.g.* [8,9]) that have shown promising performance on several applications (for a review, see [10]). These techniques are based on the so-called cavity method and are known as message-passing (MP) algorithms. They are fully distributed and easy to run on parallel machines. A recent result in this framework is an algorithm for finding a connected sub-graph of a given graph which optimizes a given factorized cost function [11].

Here we aim at making a step forward by introducing new techniques for SGI and GA. We develop two alternative MP strategies and test their performance on three sample problems. The first two are well-defined theoretical benchmarks, where our results can be compared to rigorous bounds: i) the maximum-clique problem in random graphs for the SGI problem; and ii) the alignment of two random graphs of controlled similarity. The third sample problem is thought as a proof-of-concept application in computational biology: We study iii) the alignment of the similarity networks of two interacting protein-domain families involved in bacterial signal transduction, to identify actual signaling pathways. This case, involving large networks of  $> 2500$  nodes, exploits co-evolutionary processes between interacting proteins to identify interaction partners [12].

**The model.** – Both problems, SGI and GA, can be put into the common framework of matching two graphs of possibly different size. Let  $G = (V, E, w)$  and  $G' = (V', E', w')$  be two weighted graphs with nodes  $V, V'$ , edges  $E, E'$  and edge weights  $w, w'$ . In the applications shown in this letter, weights are non-negative, but this is not a necessary condition for the applicability of the message-passing algorithms. In the case of unweighted graphs, we assume  $w$  and  $w'$  to describe the adjacency matrices, *i.e.* weights are 1 if an edge is present between two vertices, and zero else. Furthermore we denote the node number by  $N = |V|$  ( $N' = |V'|$ ), and the edge number by  $M = |E|$  ( $M' = |E'|$ ). Neighbors of a node  $i$  are assembled in  $\partial i$ . To facilitate notation, primed quantities

<sup>(a)</sup>E-mail: riccardo.zecchina@polito.it



## Clustering with shallow trees

M Bailly-Bechet<sup>1</sup>, S Bradde<sup>2,3</sup>, A Braunstein<sup>4</sup>,  
A Flaxman<sup>5</sup>, L Foini<sup>2,3</sup> and R Zecchina<sup>4</sup>

<sup>1</sup> Université Lyon 1, CNRS UMR 5558, Laboratoire de Biométrie et Biologie Évolutive, Villeurbanne, France

<sup>2</sup> SISSA, via Beirut 2/4, Trieste, Italy

<sup>3</sup> INFN Sezione di Trieste, Italy

<sup>4</sup> Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino, Italy

<sup>5</sup> IHME, University of Washington, Seattle, WA, USA

E-mail: [mbailly@biomserv.univ-lyon1.fr](mailto:mbailly@biomserv.univ-lyon1.fr), [bradde@sissa.it](mailto:bradde@sissa.it),  
[alfredo.braunstein@polito.it](mailto:alfredo.braunstein@polito.it), [abie@u.washington.edu](mailto:abie@u.washington.edu), [laura.foini@sissa.it](mailto:laura.foini@sissa.it) and  
[riccardo.zecchina@polito.it](mailto:riccardo.zecchina@polito.it)

Received 6 October 2009

Accepted 25 November 2009

Published 21 December 2009

Online at [stacks.iop.org/JSTAT/2009/P12010](http://stacks.iop.org/JSTAT/2009/P12010)

doi:[10.1088/1742-5468/2009/12/P12010](https://doi.org/10.1088/1742-5468/2009/12/P12010)

**Abstract.** We propose a new method for obtaining hierarchical clustering based on the optimization of a cost function over trees of limited depth, and we derive a message-passing method that allows one to use it efficiently. The method and the associated algorithm can be interpreted as a natural interpolation between two well-known approaches, namely that of single linkage and the recently presented affinity propagation. We analyse using this general scheme three biological/medical structured data sets (human population based on genetic information, proteins based on sequences and verbal autopsies) and show that the interpolation technique provides new insight.

**Keywords:** cavity and replica method, message-passing algorithms

**ArXiv ePrint:** [0910.0767](https://arxiv.org/abs/0910.0767)





# Percolation transition and distribution of connected components in generalized random network ensembles

Serena Bradde<sup>1,2</sup> and Ginestra Bianconi<sup>3</sup>

<sup>1</sup> International School for Advanced Studies, via Beirut 2/4, 34014, Trieste, Italy

<sup>2</sup> INFN, Via Valerio 2, Trieste, Italy

<sup>3</sup> The Abdus Salam International Center for Theoretical Physics, Strada Costiera 11, 34014, Trieste, Italy

E-mail: [bradde@sissa.it](mailto:bradde@sissa.it) and [gbiancon@ictp.it](mailto:gbiancon@ictp.it)

Received 21 January 2009, in final form 27 February 2009

Published 23 April 2009

Online at [stacks.iop.org/JPhysA/42/195007](http://stacks.iop.org/JPhysA/42/195007)

## Abstract

In this work, we study the percolation transition and large deviation properties of generalized canonical network ensembles. This new type of random networks might have a very rich complex structure, including high heterogeneous degree sequences, non-trivial community structure or specific spatial dependence of the link probability for networks embedded in a metric space. We find the cluster distribution of the networks in these ensembles by mapping the problem to a fully connected Potts model with heterogeneous couplings. We show that the nature of the Potts model phase transition, linked to the birth of a giant component, has a crossover from second to first order when the number of critical colors  $q_c = 2$  in all the networks under study. These results shed light on the properties of dynamical processes defined on these network ensembles.

PACS numbers: 00.00, 20.00, 42.10

(Some figures in this article are in colour only in the electronic version)

## 1. Introduction

Recently the study of critical phenomena in complex networks has attracted a great deal of interest [1]. One of the main critical phenomena occurring in networks is the percolation transition which is a continuous structural phase transition that can be characterized by critical indices as a statistical mechanics second-order phase transition. This phase transition determines the robustness properties of complex networks [2–5] and the critical temperature of the Ising [6–8] and XY models [9, 10] on complex networks. Moreover, the onset of a percolating cluster determines a transition in between a phase in which small loops are suppressed and a phase in which the expectation value of small loops is positive in the limit of large network sizes [11].



# The percolation transition in correlated hypergraphs

Serena Bradde<sup>1,2</sup> and Ginestra Bianconi<sup>3</sup>

<sup>1</sup> SISSA, via Beirut 2/4, 34014, Trieste, Italy

<sup>2</sup> INFN, Via Valerio 2, Trieste, Italy

<sup>3</sup> The Abdus Salam International Center for Theoretical Physics,  
Strada Costiera 11, 34014, Trieste, Italy  
E-mail: [bradde@sissa.it](mailto:bradde@sissa.it) and [gbiancon@ictp.it](mailto:gbiancon@ictp.it)

Received 12 May 2009

Accepted 17 June 2009

Published 17 July 2009

Online at [stacks.iop.org/JSTAT/2009/P07028](http://stacks.iop.org/JSTAT/2009/P07028)

[doi:10.1088/1742-5468/2009/07/P07028](https://doi.org/10.1088/1742-5468/2009/07/P07028)

**Abstract.** Correlations are known to play a crucial role in determining the structure of complex networks. Here we study how their presence affects the computation of the percolation threshold in random hypergraphs. In order to mimic the correlation in real networks, we build hypergraphs from generalized hidden variable ensembles and we study the percolation transition by mapping this problem to the fully connected Potts model with heterogeneous couplings.

**Keywords:** random graphs, networks, critical phenomena of socio-economic systems, socio-economic networks



## Critical Fluctuations in Spatial Complex Networks

Serena Bradde,<sup>1</sup> Fabio Caccioli,<sup>1</sup> Luca Dall'Asta,<sup>2</sup> and Ginestra Bianconi<sup>3</sup>

<sup>1</sup>*International School for Advanced Studies, via Beirut 2/4, 34014, Trieste, Italy*

<sup>2</sup>*The Abdus Salam International Center for Theoretical Physics, Strada Costiera 11, 34014 Trieste, Italy*

<sup>3</sup>*Department of Physics, Northeastern University, Boston, Massachusetts 02115 USA*

(Received 3 December 2009; published 26 May 2010)

An anomalous mean-field solution is known to capture the nontrivial phase diagram of the Ising model in annealed complex networks. Nevertheless, the critical fluctuations in random complex networks remain mean field. Here we show that a breakdown of this scenario can be obtained when complex networks are embedded in geometrical spaces. Through the analysis of the Ising model on annealed spatial networks, we reveal, in particular, the spectral properties of networks responsible for critical fluctuations and we generalize the Ginsburg criterion to complex topologies.

DOI: 10.1103/PhysRevLett.104.218701

PACS numbers: 89.75.Hc, 64.60.aq

A great deal of attention has been given recently to the effects that different topological properties may induce on the behavior of equilibrium and nonequilibrium processes defined on networks and to the possible implications for the study of several social, biological, and technological networks [1,2]. Heterogeneous degree distributions, small world and spectral properties, in particular, have been recognized as being responsible for novel types of phase transitions and universality classes [1–4]. For instance, scale-free networks present a complex critical behavior for the Ising model, percolation, and spreading processes that explicitly depends on the exponent of the power law in the degree distributions [1–3]. On the other hand, the existence of nontrivial spectral properties is crucial for the stability of synchronization processes and  $O(n)$  models [4].

Despite the large amount of interest in the subject, much smaller attention has been devoted to critical phenomena on complex networks embedded in a metric space [5–9], though some important problems related to navigability, efficiency, and search optimization in spatial networks have already been discussed in the literature [10–13]. In fact, spatial embedding is a very relevant aspect of infrastructure and technological networks, including airport networks, the Internet, and power-grid networks. Moreover, a pivotal role in shaping the topology of social networks is played by hidden metric structures in some underlying abstract space, such as that of the social distance between individuals [8,9].

The aim of this Letter is to investigate the role of spatial embedding in relation with the critical behavior of phase transitions in complex networks. It is well known that in regular lattices, space dimensionality governs the critical behavior of equilibrium and nonequilibrium systems. In particular, below the upper critical dimension, critical fluctuations that are not captured by the mean-field approach set in. Similarly, for complex networks embedded in a low dimensional space we can expect that, as the link proba-

bility becomes short ranged, the effect of the underlying space might change the critical behavior leading to a breakdown of the validity of (heterogeneous) mean-field arguments. This should be relevant to understand real phenomena in spatial networks, such as the spreading of viruses [6], the emergence of congested phases in the packet-based traffic on technological networks [14], and cascading failure phenomena in power-grid networks [15].

As a prototypical example of the complex behavior induced by spatial embedding, in this Letter we consider the Ising model on annealed scale-free networks. On a scale-free network with a degree distribution  $P(k) \sim k^{-\gamma_{SF}}$ , the critical temperature of the Ising model diverges for  $\gamma_{SF} < 3$ . The critical exponents, computed by means of the annealed network approximation [16] or by assuming a quenched randomness [17,18], deviate from the mean-field ones as long as  $\gamma_{SF} < 5$ , with the exception of  $\gamma, \gamma'$  describing the divergence of the magnetic susceptibility  $\chi$  close to the critical temperature  $T_c$  ( $\chi \sim |T - T_c|^{-\gamma, \gamma'}$ ). In fact,  $\gamma, \gamma'$  always remain fixed to their mean-field value  $\gamma = \gamma' = 1$ . For these reasons we refer to the critical behavior of random scale-free networks as the *heterogeneous mean-field* solution. We derive here a *Ginsburg criterion* [19] for spatial complex networks determining the condition under which critical fluctuations become larger than the ones predicted within a mean-field approach. In particular, we will show that the critical behavior is always mean field, whenever the matrix  $\mathbf{p} = \{p_{ij}\}_{i,j=1,\dots,N}$ , fixing the probabilities of existence of each link  $(i, j)$  has a finite spectral gap  $\Delta$  between the maximal eigenvalue  $\Lambda$  and the second maximal one  $\lambda_2$ . On the contrary, when the spectral gap  $\Delta \rightarrow 0$  in the thermodynamic limit, the critical behavior depends on the behavior of the tail of the spectrum of  $\mathbf{p}$ . We will demonstrate by theoretical and numerical results that the behavior of such a tail is well captured by an exponent  $\delta_S$ , related to the effective dimension  $d_{\text{eff}}$  of the network through the relation  $\delta_S = (d_{\text{eff}} - 2)/2$ . We find that for



# Bibliography

- [1] U. Alon. <http://www.weizmann.ac.il/mcb/urialon/>.
- [2] U. Alon. Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, 8(6):450.
- [3] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215:403, 1990.
- [4] Y. Artzy-Randrup, S. J. Fleishman, N. Ben-Tal, and L. Stone. Comment on network motifs: Simple buildingblocks of complex networks and superfamilies of evolved and designed networks. *Science*, 305:1107c, 2004.
- [5] A. Barla, G. Jurman, S. Riccadonna, S. Merler, M. Chierici, and C. Furlanello. Machine learning methods for predictive proteomics. *Brief Bioinform*, 9(2):119, 2008.
- [6] K. Baskerville and M. Paczuski. Subgraph ensembles and motif discovery using an alternative heuristic for graph isomorphism. *Phys. Rev. E*, 74(5):051903, Nov 2006.
- [7] M. Bayati, C. Borgs, A. Braunstein, J. Chayes, A. Ramezanzpour, and R. Zecchina. Statistical Mechanics of Steiner Trees. *Phys. Rev. Lett.*, 101:37208, 2008.
- [8] M. Bayati, A. Braunstein, and R. Zecchina. A rigorous analysis of the cavity equations for the minimum spanning tree. *Journal of Mathematical Physics*, 49(12):125206, 2008.
- [9] J. Berg and M. Lässig. Local graph alignment and motif search in biological networks. *Proc Natl Acad Sci U S A*, 101(41):14689–14694, October 2004.
- [10] H. A. Bethe. Statistical theory of superlattices. *Proc. R. Soc. A*, 150:552, 1935.

- 
- [11] G. Bezem and J. v. Leeuwen. Enumeration in graphs. Technical Report RUU-CS-87-07, Department of Information and Computing Sciences, Utrecht University, 1987.
- [12] B. B. Bollobas. *Random graphs*. Cambridge, University Press, 2001, 2nd ed. edition, 1985.
- [13] R. Boppana and M. Halldorsson. Approximating maximum independent sets by excluding subgraphs. *BIT*, 32(2):180–196, 1992.
- [14] S. Bradde, A. Braunstein, H. Mahmoudi, F. Tria, M. Weigt, and R. Zecchina. Aligning graphs and finding substructures by a cavity approach. *Europhysics Letters*, 89:37009, Feb. 2010.
- [15] A. Braunstein, M. Mézard, M. Weigt, and R. Zecchina. *Constraint satisfaction by survey propagation*, volume 9 of *Computational Complexity and Statistical Physics*, page 424. Oxford University Press, 2005.
- [16] A. Braunstein and R. Zecchina. Learning by message-passing in networks of discrete synapses. *Phys. Rev. Lett.*, 96:030201, 2006.
- [17] L. Burger and E. van Nimwegen. Accurate prediction of protein-protein interactions from sequence alignments using a bayesian method. *Mol. Sys. Biol.*, 4:165, 2007.
- [18] S. A. Cook. The complexity of theorem-proving procedures. In *STOC '71: Proceedings of the third annual ACM symposium on Theory of computing*, page 151, New York, NY, USA, 1971. ACM Press.
- [19] M. C. Costanzo, M. E. Crawford, J. E. Hirschman, J. E. Kranz, P. Olsen, L. S. Robertson, M. S. Skrzypek, B. R. Braun, K. L. Hopkins, P. Kondu, C. Lengieza, J. E. Lew-Smith, M. Tillberg, and J. I. Garrels. Ypdtm, pombepdtm and wormpdtm: model organism volumes of the bioknowledgetm library, an integrated resource for protein information. *Nucleic Acids Research*, 29(1):75–79, 2001.
- [20] T. M. Cover and J. A. Thomas. *Elements of information theory*. Wiley-Interscience, New York, NY, USA, 1991.
- [21] H. de Jong. Modeling and simulation of genetic regulatory systems: A literature review. *Journal of Computational Biology*, 9(1):67–103, 2002.



- [22] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes. Critical phenomena in complex networks. *Rev. Mod. Phys.*, 80(4):1275, 2008.
- [23] E. R. Dougherty, J. Barrera, M. Brun, S. Kim, R. M. Cesar, Y. Chen, M. Bittner, and J. M. Trent. Inference from clustering with application to gene-expression microarrays. *J Comput Biol*, 9(1):105, 2002.
- [24] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences of the United States of America*, 95(25):14863–14868, 1998.
- [25] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA*, 95(25):14863, 1998.
- [26] P. Erdős and A. Rényi. On random graphs. *Publicationes Mathematicae Debrecen*, 6:290, 1959.
- [27] S. Fields and O. Stanley. A novel genetic system to detect protein-protein interactions. *Nature*, 340:245.
- [28] B. J. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315:972, 2007.
- [29] A. Frieze and R. Kannan. Quick approximation to matrices and applications, 1999.
- [30] K. Gary and L. Ying. Verbal autopsy methods with multiple causes of death. *Statistical Science*, 23(1):78, 2008.
- [31] P. G. de Gennes. *Scaling concepts in polymer physics / Pierre-Gilles de Gennes*. Cornell University Press, Ithaca, N.Y. :, 1979.
- [32] E. N. Gilbert. Random graphs. *Ann. Math. Stat.*, 30:1141, 1959.
- [33] J. Grochow and M. Kellis. Network motif discovery using subgraph enumeration and symmetry-breaking. pages 92–106. 2007.
- [34] Y. Ho, A. Gruhler, A. Heilbut, G. D. Bader, L. Moore, S. L. Adams, A. Millar, P. Taylor, K. Bennett, K. Boutilier, L. Yang, C. Wolting, I. Donaldson, S. Schandorff, J. Shewnarane, M. Vo, J. Taggart, M. Goudreault, B. Muskant, C. Alfarano, D. Dewar, Z. Lin, K. Michalickova, A. R. Willems, H. Sassi, P. A. Nielsen, K. J. Rasmussen, J. R. Andersen, L. E. Johansen, L. H. Hansen, H. Jespersen,

- A. Podtelejnikov, E. Nielsen, J. Crawford, V. Poulsen, B. D. Sørensen, J. Matthiesen, R. C. Hendrickson, F. Gleeson, T. Pawson, M. F. Moran, D. Durocher, M. Mann, C. W. Hogue, D. Figeys, and M. Tyers. Systematic identification of protein complexes in *saccharomyces cerevisiae* by mass spectrometry. *Nature*, 415(6868):180–183, January 2002.
- [35] J. A. Hoch. Two-component and phosphorelay signal transduction. *Current Opinion in Microbiology*, 3(2):165 – 170, 2000.
- [36] L. Holm and C. Sander. Protein structure comparison by alignment of distance matrices. *Journal of Molecular Biology*, 233(1):123 – 138, 1993.
- [37] V. R. Iyer, C. E. Horak, C. S. Scafe, D. Botstein, M. Snyder, and P. O. Brown. Genomic binding sites of the yeast cell-cycle transcription factors *sbf* and *mbf*. *Nature*, 409(6819):533–538, January 2001.
- [38] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264, 1999.
- [39] M. Jerrum. Large cliques elude the metropolis process. *Random Structures and Algorithms*, 3,4:347, 1992.
- [40] M. Jerrum, A. Sinclair, and M. C. Algorithms. The markov chain monte carlo method: An approach to approximate counting and integration. pages 482–520. PWS Publishing, 1996.
- [41] D. S. Johnson, L. A. McGeoch, and E. E. Rothberg. Asymptotic experimental analysis for the held-karp traveling salesman bound. In *SODA '96: Proceedings of the seventh annual ACM-SIAM symposium on Discrete algorithms*, page 341, Philadelphia, PA, USA, 1996. Society for Industrial and Applied Mathematics.
- [42] K. Joseph B. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proc. Am. Math. Soc.*, 7(1):48, 1956.
- [43] S. Kalir, J. McClure, K. Pabbaraju, C. Southward, M. Ronen, S. Leibler, M. G. Surette, and U. Alon. Ordering genes in a flagella pathway by analysis of expression kinetics from living bacteria. *Science*, 292(5524):2080–2083, June 2001.
- [44] R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, page 85. Plenum Press, 1972.

- [45] N. Kashtan, S. Itzkovitz, R. Milo, and U. Alon. Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. *Bioinformatics*, 20(11):1746–1758, 2004.
- [46] N. Kashtan, S. Itzkovitz, R. Milo, and U. Alon. Topological generalizations of network motifs. *Phys. Rev. E*, 70(3):031909, Sep 2004.
- [47] S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22(3):437 – 467, 1969.
- [48] R. Kikuchi. A theory of cooperative phenomena. *Phys. Rev.*, 81(6):988, 1951.
- [49] W. Krauth and M. Mézard. The cavity method and the travelling-salesman problem. *Eur. Phys. Lett.*, 8:213, 1989.
- [50] F. Krzakala, A. Montanari, F. Ricci-Tersenghi, G. Semerjian, and L. Zdeborová. Gibbs states and the set of solutions of random constraint satisfaction problems. *Proc. Natl. Acad. Sci.*, 104(25):10318–10323, 2007.
- [51] M. T. Laub and M. Goulian. Specificity in two-component signal transduction pathways. *Annual Review of Genetics*, 41(1):121–145, 2007.
- [52] M. Leone, S. Sumedha, and M. Weigt. Clustering by soft-constraint affinity propagation: applications to gene-expression data. *Bioinformatics*, 23:2708, 2007.
- [53] L. Zdeborová. PhD thesis, 2009.
- [54] S. Mangan and U. Alon. Structure and function of the feed-forward loop network motif. *Proc Natl Acad Sci U S A*, 100(21):11980–11985, October 2003.
- [55] S. Mangan, A. Zaslaver, and U. Alon. The coherent feedforward loop serves as a sign-sensitive delay element in transcription networks. *J Mol Biol*, 334(2):197–204, November 2003.
- [56] E. Marinari and R. Monasson. Circuits in random graphs: from local trees to global loops. *Journal of Statistical Mechanics: Theory and Experiment*, 2004(09):P09004, 2004.

- 
- [57] E. Marinari, G. Semerjian, and V. Van Kerrebroeck. Finding long cycles in graphs. *Phys. Rev. E*, 75(6):066708, Jun 2007.
- [58] M. Mézard and G. Parisi. Replicas and optimization. *J. Phys. (Paris) Lett.*, 46:L-771, 1985.
- [59] M. Mézard and G. Parisi. Mean-field equations for the matching and the travelling salesman problems. *Eur. Phys. Lett.*, 2:913, 1986.
- [60] M. Mézard and G. Parisi. A replica analysis of the travelling salesman problem. *J. Phys. (Paris) Lett.*, 47:1285, 1986.
- [61] M. Mézard and G. Parisi. Mean-field theory of randomly frustrated systems with finite connectivity. *Eur. Phys. Lett.*, 3(10):1067, 1987.
- [62] M. Mézard and G. Parisi. The bethe lattice spin glass revisited. *Eur. Phys. J. B*, 20:217, 2000.
- [63] M. Mézard and G. Parisi. The bethe lattice spin glass revisited. *Eur. Phys. J. B*, 20:217, 2001.
- [64] M. Mézard, G. Parisi, and R. Zecchina. Analytic and Algorithmic Solution of Random Satisfiability Problems. *Science*, 297:812, 2002.
- [65] M. Mézard and R. Zecchina. Random k-satisfiability: from an analytic solution to a new efficient algorithm. *Phys. Rev. E*, 66:056126, 2002.
- [66] C. Micheletti and H. Orland. MISTRAL: a tool for energy-based multiple structural alignment of proteins. *Bioinformatics*, 25(20):2663–2669, 2009.
- [67] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824, October 2002.
- [68] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman, and L. Troyansky. Computational complexity from characteristic phase transitions. *Nature*, 400:133, 1999.
- [69] A. Montanari and A. Dembo. Ising models on locally tree-like graphs. *Ann. Appl. Probab.*, 2:565, 2010.
- [70] A. Montanari, R.-T. F., and G. Semerjian. Solving constraint satisfaction problems through belief propagation-guided decimation. *CoRR*, abs/0709.1667, 2007.

- [71] A. Montanari and M. Mézard. *Information, Physics, and Computation*. OUP Oxford, 2009.
- [72] C. J. L. Murray, A. D. Lopez, D. M. Feehan, S. T. Peter, and G. Yang. Validation of the symptom pattern method for analyzing verbal autopsy data. *PLoS Med*, 4(11):e327, 2007.
- [73] A. Murzin, S. Brenner, T. Hubbard, and C. Chothia. Scop: A structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, 247:536, 1995.
- [74] H. Orland. ... *J. Phys. (Paris) Lett.*, 46:L-763, 1985.
- [75] A. Paccanaro, J. A. Casbon, and M. Saqi. Spectral clustering of protein sequences. *Nucleic Acid Research*, 34:1571, 2006.
- [76] C. M. Papadimitriou. *Computational complexity*. Addison-Wesley, Reading, Massachusetts, 1994.
- [77] G. Parisi. *Statistical Field Theory (Frontiers in Physics)*. Addison Wesley.
- [78] G. Parisi. Some remarks on the survey decimation algorithm for k-satisfiability. *CoRR*, cs.CC/0301015, 2003.
- [79] J. Pearl. *Probabilistic reasoning in intelligent systems : networks of plausible inference*. Morgan Kaufmann, 1997.
- [80] A. G. Percus and O. C. Martin. The stochastic traveling salesman problem: Finite size scaling and the cavity prediction. *J. of Stat. Phys.*, 94(5):739, 1999.
- [81] S. Peri and et al. Development of Human Protein Reference Database as an Initial Platform for Approaching Systems Biology in Humans. *Genome Research*, 13(10):2363–2371, 2003.
- [82] A. Procaccini, B. L., H. Szurmant, T. Hwa, and M. Weigt. *in preparation*, 4:165, 2010.
- [83] G. R. G. Low-density parity check codes. *IEEE Trans. Inform. Theory*, 8:21, 1962.
- [84] A. K. Ramani and E. M. Marcotte. *J. Mol. Biol.*, 327:273, 2003.
- [85] F. Schreiber and H. Schwöbbermeyer. Mavisto: a tool for the exploration of network motifs. *Bioinformatics*, 21(17):3572–3574, 2005.

- [86] S. S. Shen-Orr, R. Milo, S. Mangan, and U. Alon. Network motifs in the transcriptional regulation network of escherichia coli. *Nat Genet*, 31(1):64–68, May 2002.
- [87] P. Smolen, D. A. Baxter, and J. H. Byrne. Modeling transcriptional control in gene networks—methods, recent results, and future directions. *Bulletin of Mathematical Biology*, 62(2):247 – 292, 2000.
- [88] M. J. Solomon and A. Varshavsky. Formaldehyde-mediated DNA-protein crosslinking: a probe for in vivo chromatin structures. *Proceedings of the National Academy of Sciences of the United States of America*, 82(19):6470–6474, 1985.
- [89] E. Sontag. Monotone and near-monotone biochemical networks. *Systems and Synthetic Biology*, 1(2):59–87, April 2007.
- [90] N. Sourlas. Statistical mechanics and the travelling salesman problem. *Eur. Phys. Lett.*, 2(12):919, 1986.
- [91] A. M. Stock, V. L. Robinson, and P. N. Goudreau. Two-component signal transduction. *Ann. Rev. Biochem.*, 69(1):183–215, 2000.
- [92] D. J. Thouless, A. P. W., and R. G. Palmer. Solution of ‘solvable model of a spin glass’. *Philosophical Magazine*, 35:593, 1977.
- [93] A. H. Y. Tong, M. Evangelista, A. B. Parsons, H. Xu, G. D. Bader, N. Page, M. Robinson, S. Raghibizadeh, C. W. V. Hogue, H. Bussey, B. Andrews, M. Tyers, and C. Boone. Systematic Genetic Analysis with Ordered Arrays of Yeast Deletion Mutants. *Science*, 294(5550):2364–2368, 2001.
- [94] P. Uetz, L. Giot, G. Cagney, T. A. Mansfield, R. S. Judson, J. R. Knight, D. Lockshon, V. Narayan, M. Srinivasan, P. Pochart, A. Qureshi-Emili, Y. Li, B. Godwin, D. Conover, T. Kalbfleisch, G. Vijayadamodar, M. Yang, M. Johnston, S. Fields, and J. M. Rothberg. A comprehensive analysis of protein-protein interactions in saccharomyces cerevisiae. *Nature*, 403(6770):623–627, February 2000.
- [95] J. Venter, K. Remington, J. Heidelberg, A. Halpern, and D. Rusch. Environmental genome shotgun sequencing of the sargasso sea. *Science*, 304:66, 2004.
- [96] M. Weigt, R. White, H. Szurmant, J. Hoch, and T. Hwa. *Proc. Natl. Acad. Sci.*, 106:67, 2009.

- 
- [97] A. H. West and A. M. Stock. Histidine kinases and response regulator proteins in two-component signaling systems. *Trends in Biochemical Sciences*, 26(6):369, 2001.
- [98] K. M. Wong, M. A. Suchard, and J. P. Huelsenbeck. Alignment Uncertainty and Genomic Analysis. *Science*, 319(5862):473–476, 2008.
- [99] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Generalized belief propagation. In *In NIPS 13*, volume 13, page 689, 2001.
- [100] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Constructing free energy approximations and generalized belief propagation algorithms. *IEEE Trans. Inform. Theory*, 51:2282, 2005.

