Exact Determination of all Ground States of Random Field Systems in Polynomial Time

A. K. Hartmann and K. D. Usadel
Theoretische Tieftemperaturphysik
Gerhard Mercator Universität
47048 Duisburg, Germany
(November 3, 1998)

An algorithm is developed which allows calculating all ground states of ferromagnetic and unfrustrated antiferromagnetic Ising systems with arbitrary site dependent fields by transforming the system into an equivalent network and calculating the maximal flow. By a trial and error scheme a minimum cut is constructed which corresponds to a spin configuration. In this way each ground state is calculated with a finite probability. The algorithm is applied to site diluted antiferromagnets in external magnetic fields. It is found that in this case its time complexity is approximately quadratic in the lattice size. As an application we calculate the distribution of overlaps between groundstates of the site diluted antiferromagnet in a strong magnetic field and we analyse the fractal structure of these groundstates.
key words: ground states; degeneration; exact calculation; minimum cut; maximum flow; diluted antiferromagnet; distribution of overlaps; fractal structure.

I. INTRODUCTION

Ground state properties of complex systems with a phase space containing a large number of valleys are very difficult to calculate. An example are spin glasses where the exchange interactions between pairs of spins have different signs leading to frustrations [1]. Conventional methods like simulated annealing [2] suffer from the problem that the system gets trapped in a metastable state which it can not leave in finite computation time and are therefore not very helpful. The direct construction of exact ground states can be formulated as an optimization problem which is NP complete for frustrated systems [3]. However, it has been realized earlier that for systems with unfrustrated exchange interactions ground states can be found in polynomially bounded computing time by mapping the systems onto a certain network with an associated network flow which has to be optimized [4]. This method has been applied succesfully to the random field Ising ferromagnet by Ogielski [5] who calculated for each random field configuation one ground state. In diluted systems a large ground state degeneration is expected and one is therefore interested in calculating as many ground states as possible. It is the purpose of the present paper to generalize the up to now used algorithms in such a way that any of the degenerate ground states can be obtained with a finite probability. The construction which is described in the following is possible for any Ising type hamiltonian which with local gauge transformations can be mapped onto a hamiltonian with ferromagnetic exchange interactions and with arbitrary local fields. For concreteness we describe the application of this algorithm to a diluted antiferromagnet in a field (DAFF) which is a prominent experimental realization of a random field system [6].

The algorithm is based on the theory of graphs and especially on algorithms calculating maximum flows in networks. These concepts can be found in textbooks [7–9]. The general scheme for the transformation of a Hamiltonian into a network can be found in [4]. To keep the paper selfcontained and to provide the reader not familiar with these mathematical concepts with necessary background we feel that it is useful to give a condensed review of this subject. The new algorithm for calculating all ground states of a random field system is described in detail and its time complexity is investigated.

The outline of the paper is as follows. First we describe how an Ising system with site dependend fields is transformed into an equivalent network in order to calculate exactly one of its ground states. Next we demonstrate, how each degenerated ground state of such a system can be detected with a non-vanishing probability and we test the time complexity of this new algorithm. In the final part the distribution function of the overlaps for different ground states of the DAFF in a strong magnetic field is calculated and an analysis of the fractal structure of the domains is performed.

II. TRANSFORMATION OF THE ISING SYSTEM INTO A NETWORK

The Hamiltonian of the system is given by

$$H = \sum_{\langle i,j \rangle} J_{ij}\epsilon_i\epsilon_j\sigma_i\sigma_j - \sum_i B_i\epsilon_i\sigma_i. \tag{1}$$

The geometry is simple-cubic. The magnetic field B_i is local so that it is possible to model random field systems. The interaction is antiferromagnetic ($J_{ij} > 0$) and restricted to nearest neighbours. Each bond $\langle i,j \rangle$ is only counted once. We assume that the lattice sites are numbered from 1 to n. Each site is occupied ($\epsilon_i = 1$) with a probability p with an Ising spin ($\sigma_i = \pm 1$).
The lattice is transformed into a *network*, i.e. a directed graph $G = (V, E)$ with capacities $c(i,j) \geq 0$ assigned to the edges $(i,j) \in E$ of this graph in the following way [4].
The vertices $i \in V$ of the network are the sites of the lattice plus two extra sites, a source 0 and a sink $n+1$. For each bond of the system a directed edge is constructed. In general the direction of the edges is arbitrary. We make use of this freedom and assign nonvanishing capacities to edges (i,j) with $i < j$ in order to avoid cycles in the graph. This choice is necessary for the new algorithm presented later. For each site of the lattice there is also an edge from the source to that site or from that site to the sink. There is also a (phantom) edge $(0, n+1)$. The lattice is divided into two penetrating sublattices A, B with $t(i) = 1$ for $i \in A$ and $t(i) = -1$ for $i \in B$. With $N(i)$ we denote the set of nearest neighbours belonging to the site i. Then the capacities are set to the following values:

- Let
$$c(i,j) := \begin{cases} 4J_{ij}\epsilon_i\epsilon_j & \text{if} \quad j \in N(i) \wedge i < j \\ 0 & \text{otherwise} \end{cases} \tag{2}$$
for all $i, j = 1, \ldots, n$.
- Let $c(i,0)$, $c(n+1,i)$ and $c(i,i) = 0$ for all $i = 0, 1, \ldots, n+1$.
- Let $w(i) := \epsilon_i[2\sum_{j \in N(i)} J_{ij}\epsilon_j - 2B_i t(i)] - \sum_{j \in N(i)} c(i,j)$ for all $i = 1, \ldots, n$.
- If $w(i) \geq 0$ then let $c(i, n+1) := w(i)$ and $c(0,i) := 0$,
else let $c(i, n+1) := 0$ and $c(0,i) := -w(i)$.
- Let $c(0, n+1) := -\sum_{\langle i,j \rangle} J_{ij}\epsilon_i\epsilon_j + \sum_{i=1}^{n} B_i\epsilon_i t(i) - \sum_{j=1}^{n} c(0,j)$.

All capacities exept that of the edge $(0, n+1)$ are not negative, independend of the sign of the local fields B_i. Vanishing capacities represent non-existent edges and vice versa.
A *cut* separating 0 and $n+1$ is any partition $(S, \overline{S})$ of the vertices ($S \cup \overline{S} = V$ and $S \cap \overline{S} = \emptyset$) which fulfills $0 \in S$ and $n+1 \in \overline{S}$. The capacity of a cut is the sum of the capacities of edges leading from a vertex $i \in S$ to a vertex $j \in \overline{S}$:

$$C(S, \overline{S}) = \sum_{i \in S, j \in \overline{S}} c(i,j). \tag{3}$$

A cut can be represented by a vector $X = (1, x_1, \ldots, x_n, 0)$ with $x_i = 1$ if $i \in S$ and $x_i = 0$ otherwise [10]. By using this vector one can write the capacity of a cut as

$$c(S, \overline{S}) = c(X) := \sum_{i,j=0}^{n+1} c(i,j)x_i(1 - x_j) \tag{4}$$

which is a quadratic form in X. Substituting $x_0 = 1$, $x_{n+1} = 0$, $\sigma_i = (2x_i - 1)t(i)$, using only edges (i,j) with $i < j$ and comparing with the Hamiltonian (1) results in the scheme (2). So the minimum of the Hamiltonian is equal to the minimum of the cut–capacity (3), the capacity of the so called minimum cut [4].
To determine the capacity of the minumum cut the *max flow–min cut theorem* [11] is used: The maximum flow through the network is equal to the capacity of the minimum cut. The maximum flow can be calculated with the *Ford–Fulkerson Algorithm* (FFA) [11]. It is explained in textbooks, see for instance [7,8]. It uses flow values $f(i,j)$ assigned to each edge. At vertices flow conservation holds. Initially all flows are set to zero. The flow through the network is then stepwise augmented starting from the sink. This process is repeated up to the point where no more increment is possible.[1] The egde $(0, n+1)$ is not regarded in the algorithm, because its capacity is usally negative. The value of $c(0, n+1)$ is added afterwards to the result, because this edge connects always the sets S and $\overline{S}$. It is recommended to use only integer (or rational) values for the capacities to avoid convergency problems.

[1] We did choose the flow augmenting paths at random in order to avoid multiple selection of 'unlucky' paths (see [12]).

III. CALCULATION OF ALL GROUND STATES

The output of the Ford–Fulkerson algorithm is a set of flows $\{f(i,j)\}$. An easy way to determine *one* minimum cut is the following construction [9]: Define S as the set of those vertices i which are connected to the source 0 by a path whose edges fulfill the condition $f(i,j) < c(i,j)$ and let $\overline{S} := V \setminus S$. Then the partition $(S, \overline{S})$ is a minimum cut.
In systems with degenerated ground states many minimum cuts exist. With this approach one can determine only one minimum cut.
To develop an algorithm which gives all ground states, it is important to notice that in a case of a minimum cut the edges connecting the subsets S and $\overline{S}$ do fulfill $f(i,j) = c(i,j)$, i.e. they are *saturated*. Note that there exist also edges which are saturated but do not connect S and $\overline{S}$ in any minimum cut (see appendix A).
This leads to a simple algorithm which removes edges iteratively in order to cut the network into two pieces:

1. Calculate the flow through the network with the FFA.
2. Remove an untested edge with $f(k,l) = c(k,l)$.
3. Recalculate the flow through the network.
4. If the recalculated flow is lower than the old flow by the amount of the capacity of the removed edge, then goto 1,
   else reinsert the edge and goto 2.
5. Let $S := \{i \in V | i \text{ is connected to } 0\}$ and $\overline{S} = V \setminus S$.

It is important, that the test edges are chosen at random, so that each ground state is calculated with a non–vanishing probability. At the beginning we put all test edges with $f(k,l) = c(k,l)$ in an array. Each edge is chosen with equal probability. The last edge in the array replaces the edge with is selected for testing. Because the flow is decreased in the run of the algorithm, each selected edge is proved whether it still fulfills $f(k,l) = c(k,l)$. If this is not true then the edge is no longer a candidate for removal.
The disadvantage of this algorithm is that for every removal of an edge the whole flow through the network is recalculated. Therefore the algorithm is very time consuming.
An alternative approch is not to recalculate the flow, but to 'subtract' the flow through the removed edge out of the network. Afterwards one tries to increase the flow with the FFA. If this is possible, then the edge is reinserted. Because in any path from the source 0 to the sink $n+1$ there is only one edge to be removed in order to construct a minimum cut, one can split the flow which is to be subtracted among all ancestors and descandants of the edge. So none of them is saturated. This reduces the number of candidats which can be removed.
Because there are only edges (i,j) with $i < j$, the subtraction of the flow can be done in one linear sweep through the network. In the following subtraction algorithm the edge (k,l) is to be removed. The values $rb(i), ra(i)$ are the amounts of flow which have to be subtracted 'before' and 'after' a vertex i:

1. Let $rb(i) := 0$ and $ra(i) := 0$ for $i = 0, \ldots, n+1$.
2. Let $rb(k) := f(k,l)$, $ra(l) := f(k,l)$ and $f(k,l) := 0$.
3. For $i = k, \ldots, 1$ do steps 4 to 6:
4. Distribute the flow $rb(i)$ among the ancestors j of i: for all $j \in V$ with $(j,i) \in E$ let the parts $p(j) := rb(i) f(j,i) / \sum_m f(m,i)$.
5. Decrease the flow for all $j \in V$ with $(j,i) \in E$: $f(j,i) := f(j,i) - p(j)$.
6. Add the parts for all $j \in V$ with $(j,i) \in E$: $rb(j) := rb(j) + p(j)$.
7. For $i = l, \ldots, n$ do steps 8 to 10:
8. Distribute the flow $ra(i)$ among the descendants j of i: for all $j \in V$ with $(i,j) \in E$ let the parts $p(j) := ra(i) f(i,j) / \sum_m f(i,m)$.
9. Decrease the flow for all $j \in V$ with $(i,j) \in E$: $f(i,j) := f(i,j) - p(j)$.
10. Add the parts for all $j \in V$ with $(i,j) \in E$: $ra(j) := ra(j) + p(j)$.

In appendix B an example demonstrates how the substraction of the flow works.
In the case of an antiferromagnetic system one has to be careful about the choice of the function $t(i)$, because for some 'critical' values of the magnetic field it has an influence on the ground states that can be detected. In appendix C an example is shown where not all ground states can be constructed. The solution to this problem is as follows: For every isolated cluster of occupied sites it has to be chosen at random whether in the function $t(i)$ the value -1 is assigned to sublattice A or B. Alternatively one also can do this selction for each cluster in such a way that its contribution to the capacity $c(0, n+1)$ is as negative as possible. So every ground state can be detected with non–vanishing probability.
A very important quantity for the application of an algorithm is its time complexity because obviously it determines its usefulness in applications. To determine the time complexity of this algorithm we calculated ground states of 2 and 3-dimensional diluted antiferromagnets (p=0.7, periodic boundary conditions) with homogeneous couplings and fields at $B/J = 2.05$. In these calculations each test edge had the same likelihood. As hardware an IBM workstation (RS/6000 mod. 580) with AIX operating system was used. The algorithm was implemented in the language C using the option $-O$ for compiling. The CPU–times for calculating a ground state configuration were obtained and for a fixed lattice size n this time was averaged over 200 groundstates for different vacancy configurations. This averaged time $t(n)$ is shown as a function of lattice size n in a logarithmic plot in figure 1. The bars represent the 95%–confidence intervals.
The resulting time complexity is approximately quadratic in the lattice size:

$$\text{2–dim:} \quad t(n) \in O(n^{2.08 \pm 0.04}) \tag{5}$$

$$\text{3–dim:} \quad t(n) \in O(n^{2.07 \pm 0.04}) \tag{6}$$

With this new algorithm every groundstate can be detected with nonvanishing probability. One should notice, however, that the likelihoods for different ground states are not the same, even if the magnetic field is not 'critical'. If one selects the test edges with probabilities proportional to their capacities, the distribution of the ground states somewhat equilibrates, but this adds an extra factor of $\log n$ to the time complexity.

IV. DISTRIBUTION OF OVERLAPS AND STRUCTURE OF DOMAINS

In this section the ground state algorithm is applied to a calculation of the overlaps of ground state spin configurations and of the structure of domains for diluted antiferromagnets in a strong magnetic field.
In [13] a phase diagram of a diluted antiferromagnet in three dimensions with 50 % dilution and constant exchange interactions $J_{ij} = J = 1$ in an external field $B_i = B$ was determined by Monte Carlo simulation. The phase diagram is divided into three sections: A low temperature and low field antiferromagnetically (AFM) ordered region, a paramagnetic (PM) one and a so called domain state (SG) which is characterized by strong irreversibilities and the appearence of frozen antiferromagnetic domains showing fractal behaviour. In [14] the line separating the SG and the PM state is found to exhibit a shape resembling the so called AT line in the case of spin glasses [1].
To contribute to an understanding of the nature of the domain state we calculated ground states for system sizes $n = 12^3, 16^3, 20^3$, and 24^3. We randomly selected 200 vacancy configurations (120 configurations for $n = 24^3$) and calculated for each configuration 10 ground states. We used a large value of the magnetic field, $B/J = 3.025$, which is well in the domain state. At this value of the magnetic field the antiferromagnetic order parameter, the staggered magnetization, is practically zero (less than 1 % of its saturation value) so that long range antiferromagnetic order definitely can be ruled out.
To characterize the domain state we used a well known spin glass order parameter: the distribution of overlaps between these randomly selected ground states [15]. Some care is necessary, however, to properly define this overlap in the present case in order to achieve good statistics. First, due to the rather strong dilution the sites occupied by spins form clusters of different sizes ranging from single sites to the largest (percolating) cluster. Here, as usual, two sites are said to belong to the same cluster if they are both occupied by spins and are nearest neighbors on the lattice. The ground state properties of a large system are determined by the spins on the percolating cluster, isolated clusters only contribute to a background without structure. Therefore, in calculating the overlaps we only consider the spins on the percolating cluster. Second, for large magnetic fields some spins are parallel to the external field in all ground states. Using for a three dimensional system a field value larger than $3 * J$ all spins with less than four neighbours are always pointing in the direction of the external field. They contribute in a trivial way to the overlap. But since their number varies for different spin configurations better statistics is achieved if they are left out. Thus we calculate

$$q^{\alpha\beta} := \frac{1}{N_c} \sum_i \sigma_i^\alpha \sigma_i^\beta \tag{7}$$

where $\{\sigma_i^\alpha\}$ and $\{\sigma_i^\beta\}$ are any two of the ten calculated ground states of the system for fixed vacancy configuration and the sum runs over those N_c spins of the percolating cluster which have more than B/J neighbours.
These overlaps are calculated for a large number of vacancy configurations and the corresponding probability distribution $P_L(q)$ is obtained by counting the numbers of overlaps within intervals with magnitude Δq (0.01 for $L = 12, 16$ and 0.005 for $L = 20, 24$) and following normalization. The result is shown in figure 2.
For increasing lattice sizes n this distribution gets steeper, in the $n \to \infty$ limit one expects from these results that it approaches a delta distribution. Fitting the different distributions to Gaussians and extrapolating to $n \to \infty$ we obtain for the center of the distribution $q_0 \approx 0.87$.
Our calculations show that for large systems the probability of getting the same ground state goes to zero rapidly. Thus a very large groundstate degeneration is obvious. Due to the strong external field these groundstates have a preferred spin direction resulting in a distribution function not symmetric in q but well localized around q_0.
In [16] the distribution of overlaps was determined for the SK spin glass model by Monte Carlo simulation. Below the AT line in zero magnetic field all values $q \in [-1, 1]$ have a non-vanishing probability. Above the AT line the probability function in an external magnetic field shows a similar finite size behaviour as in the present case. Whether both cases have a common origin remains unclear, more work is needed to decide whether the domain state resembles a spin glass state as it was proposed in [13].
In [17] it was shown that the antiferromagnetic domains formed in the DAFF at low temperatures have an interesting fractale structure. The results were obtained within a Monte Carlo simulation cooling the system in an external magnetic field. The frozen domains were analysed and the fractal properties of a variety of physical quantities were obtained. The same analysis is applied to the exact groundstates of the DAFF at $B = 3.025$. In particular, we analysed all domains in all the calculated groundstates and averaged the relevant quantities over all domains of equal volume, see [17] for further details. Interestingly, the fractal dimensions for all quantities obtained are pratically the same as those obtained from the Monte Carlo data. As an example in figure 3 the magnetization–volume relation is shown in a logarithmic plot and compared with the results of [17]. The fractal dimension $d_m \approx 0.94$ (defined through the relation $M \sim V^{d_m}$) coincides while the proportionality constants are slightly different which is attributed to the slightly different fields used in both approaches.
In conclusion we would like to stress that the new algorithm described in the present paper has a large variety of interesting physical applications. It cannot only contribute to an understanding of ground state properties of a large class of random field and random bond systems but also to low temperature properties of these systems by performing Monte Carlo simulations starting from the exact groundstates. In this way the phase space relevant for low temperatures is probed. It is expected that this approach leads to new results in a temperature region which is only difficult to access with conventional methods. Work in this direction is in progress.

ACKNOWLEDGMENTS

We thank U. Nowak for fruitful discussions and a critical reading of the manuscript. This work was supported by the Deutsche Forschungsgemeinschaft through Sonderforschungsbereich 166.

APPENDIX A: A FOUR SPIN SYSTEM

The following network is constructed out of a 2-dimensional four–spin system with open boundary conditions and antiferromagnetic couplings $J_{12} = J_{14} = J_{23} = J_{34} = 1$. The magnetic field is zero. Figure 4 shows a possible result of the FFA. The pairs of numbers are $(c(i,j),\ f(i,j))$. Not shown are edges with capacity $c = 0$ and the phantom edge $(0, 5)$.
Only the edges $(0, 1)$ or $(4, 5)$ can be removed out of the graph in order to construct a minimum cut. The edge $(1, 4)$ cannot be removed, because the flow can be redirected through the edges $(1, 2)$, $(2, 3)$ and $(3, 4)$. The two resulting spin–configurations are purely antiferromagnetic, the second one being the reversion of the first.

APPENDIX B: SUBSTRACTION OF THE FLOW

As an example for the substraction of flow out of a network we show in figure 5 a part of a network in which the edge $(9, 10)$ is to be removed.

In figure 6 the network is displayed after the removal of this edge $(9,10)$. The flow is decreased about the magnitude 6. Notice how the substracted flow is partitioned between the ancestors and descendants of the vertices 9 respectivley 10. The flow into the vertices 3, 8 and out of 16 and 11 is treated in the same way, but not shown.

APPENDIX C: HIDDEN GROUND STATES

In figure 7 a network is shown which is equivalent to a linear system consisting of two pairs of spins seperated by a vacancy. The boundary conditions are open. All couplings and all fields have the value $J = B = 1$. The function $t(i)$ is chosen as $t(1) = t(3) = t(5) = 1$ and $t(2) = t(4) = -1$. The capacity of the phantom edge is $c(0,6) = -6$.
In this situation it is possible to cut the network by removing the edges $(0,1)$, $(1,2)$ or $(2,6)$ which are representing the ground states $\uparrow\downarrow x \uparrow\downarrow$, $\uparrow\uparrow x \uparrow\downarrow$ and $\downarrow\uparrow x \uparrow\downarrow$. The other ground states with the last two spins reversed or both pointing upwards are only detectable if one chooses $t(1) = 1, t(2) = -1, t(4) = 1, t(5) = -1$.

---

[1] For reviews see K. Binder and A. P. Young, Rev. Mod. Phys. **58**, 801 (1986) ; D. P. Belanger and A. P. Young, JMMM **100**, 272 (1991)
[2] S. Kirkpatrick, C. D. Gelatt Jr. and M. P. Vecchi, Science **220**, 671 (1983)
[3] F. Barahona, J. Phys. **A 15**, 3241 (1982)
[4] J. P. Picard, H. D. Ratliff, Networks **5**, 357 (1975)
[5] A. T. Ogielski, Phys. Rev. Lett. **57**, 1251 (1986)
[6] S. Fishman and A. Aharony, J.Phys. **C 12**, L729 (1979)
[7] M. N. S. Swamy and K. Thulasiraman, Graphs, Networks, and Algorithms, John Wiley & Sons, New York 1981
[8] J. D. Claiborne, Mathematical preliminaries for computer networking, John Wiley & Sons, New York 1990
[9] W. Knödel, Graphentheoretische Methoden und ihre Anwendung, Springer, Berlin 1969
[10] P. L. Hammer, Operations Research **13**, 388 (1965)
[11] L. R . Ford, D. R. Fulkerson, Canadian J. Math. **8**, 399 (1956)
[12] R. E. Tarjan, Data Structures and Network Algorithms, Society for Industrial and Applied Mathematics, Philadelphia, 1983
[13] K. D. Usadel and U. Nowak, J.Magn.Magn. **104–107**, 179 (1992)
[14] U. Nowak and K. D. Usadel, Phys.Rev. **B 44**, 7426 (1991)
[15] G. Parisi, J.Phys. **A 13**, 1101 (1980)
[16] A. P. Young, J.Appl.Phys. **57**, 3361 (1985)
[17] U. Nowak and K. D. Usadel, Phys.Rev. **46**, 8329 (1992)

APPENDIX: FIGURE CAPTIONS

Figure 1 Time complexity of the algorithm for DAFF (p=0.7), $T = 0$, $d = 2, 3$, $B/J = 2.05$, lattice sizes $n = 4^3$ to $n = 18^3$ (for $d = 2$ analogous).

Figure 2 Distribution of overlaps on the percolating cluster using the spins with more than 4 neighbours for the DAFF ($p = 0.5$) at $T = 0$, $B/J_B = 3.025$, with $n = 24^3, 20^3, 16^3, 12^3$. The lines are guides for the eyes only.

Figure 3 $\ln M$ vs $\ln V$. All quantities are dimensionless and explained in the text. Upper curve: results obtained from exact groundstates, $B = 3.025$. Lower curve: Monte Carlo results, $B = 3$.

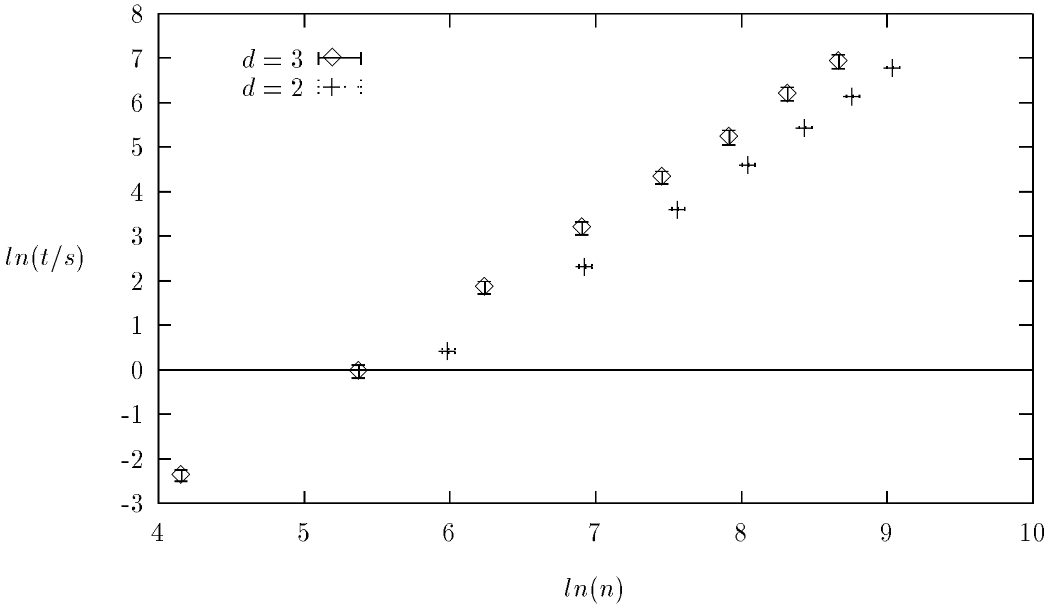
Figure 4 A four spin graph.

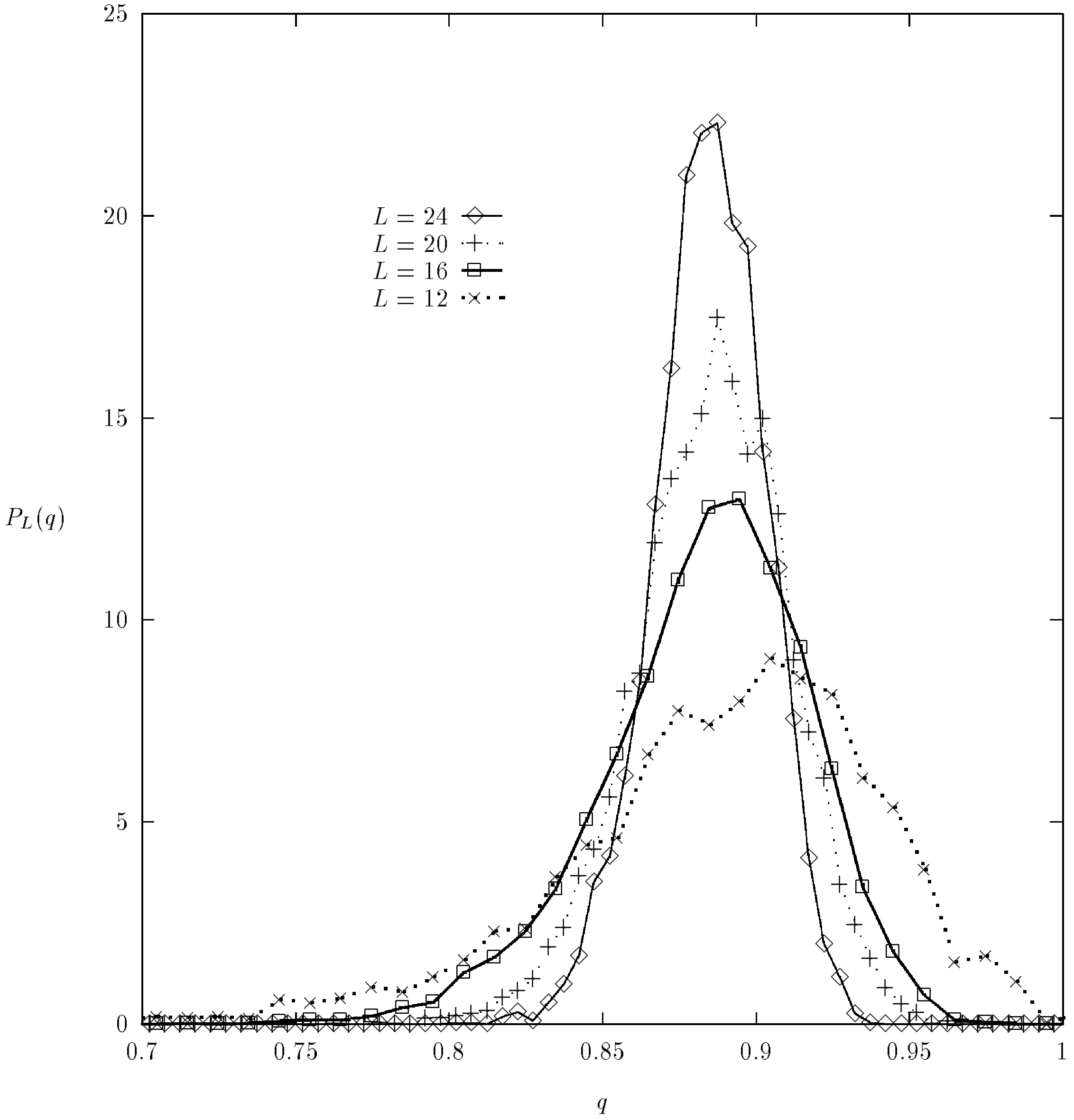
Figure 5 Part of an example–network.

Figure 6 Network after substracting the flow $f(9, 10)$.

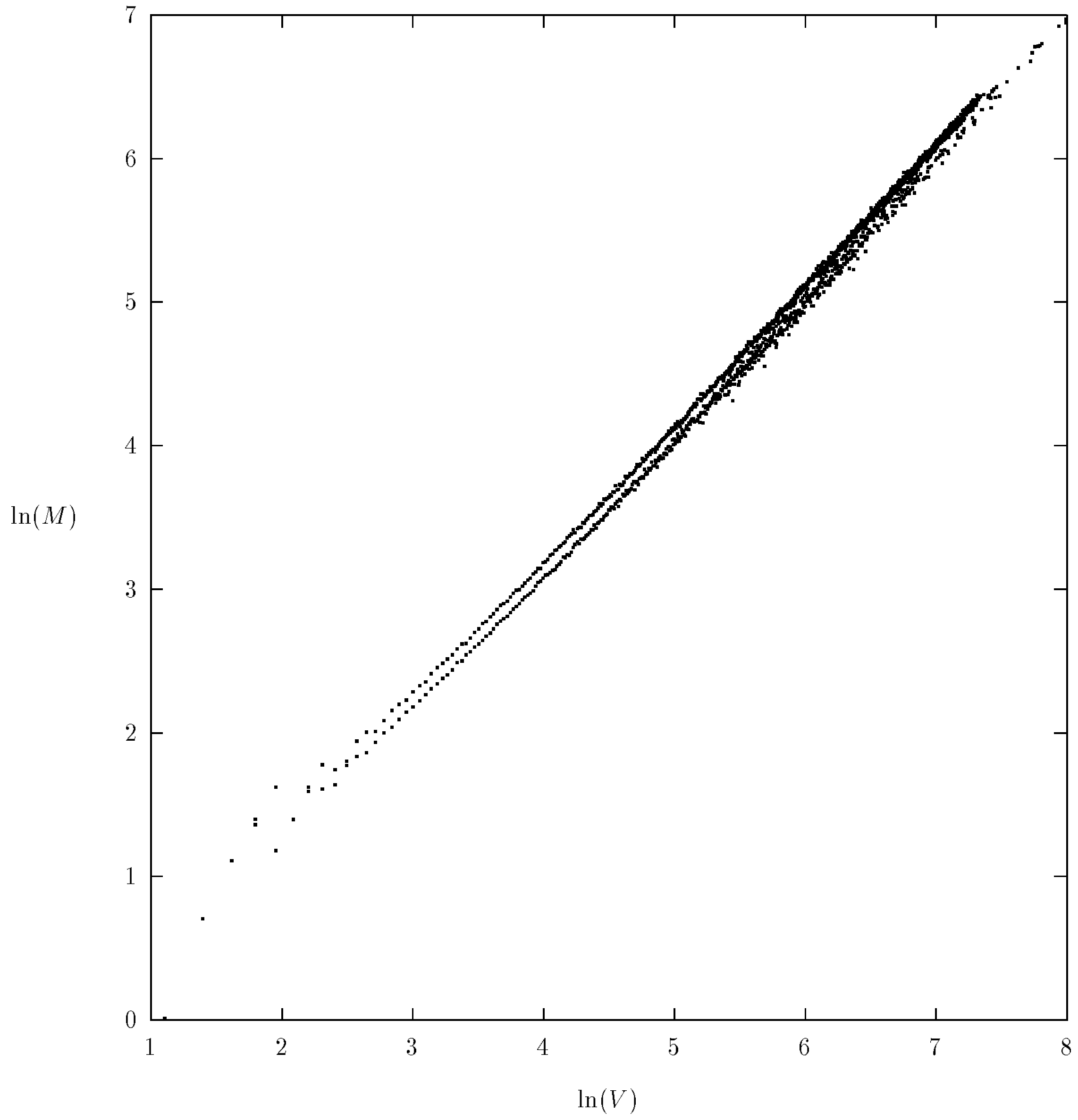
Figure 7 Diluted antiferromagnetic chain with not all ground states detectable because of the choice of the function $t(i)$.

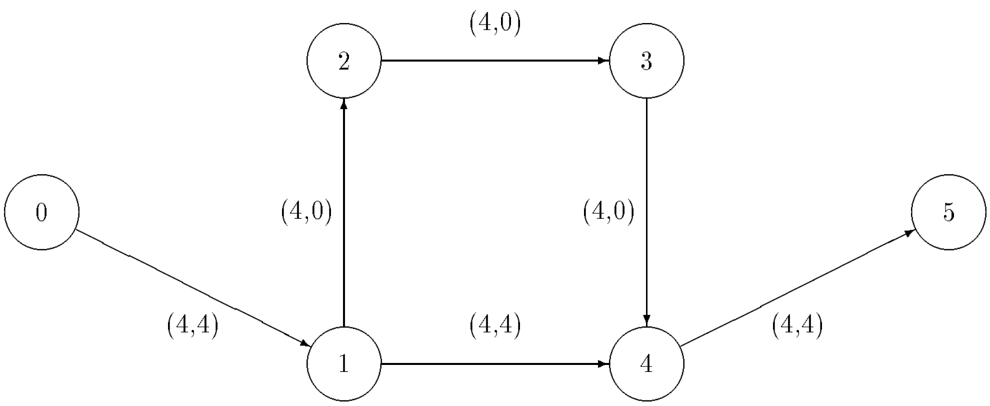
APPENDIX: FIGURES

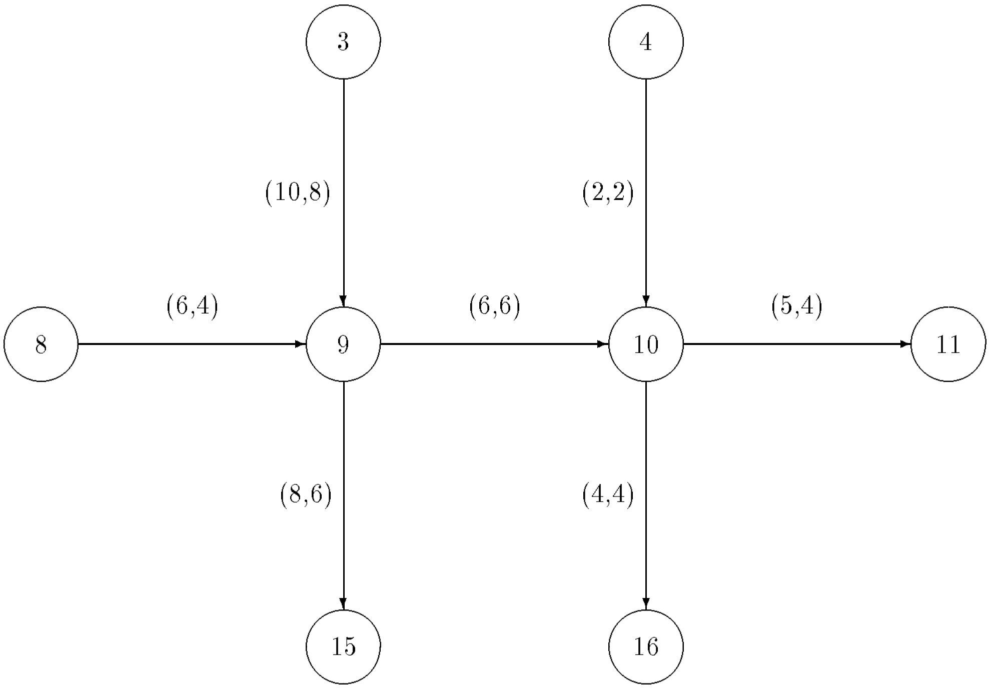


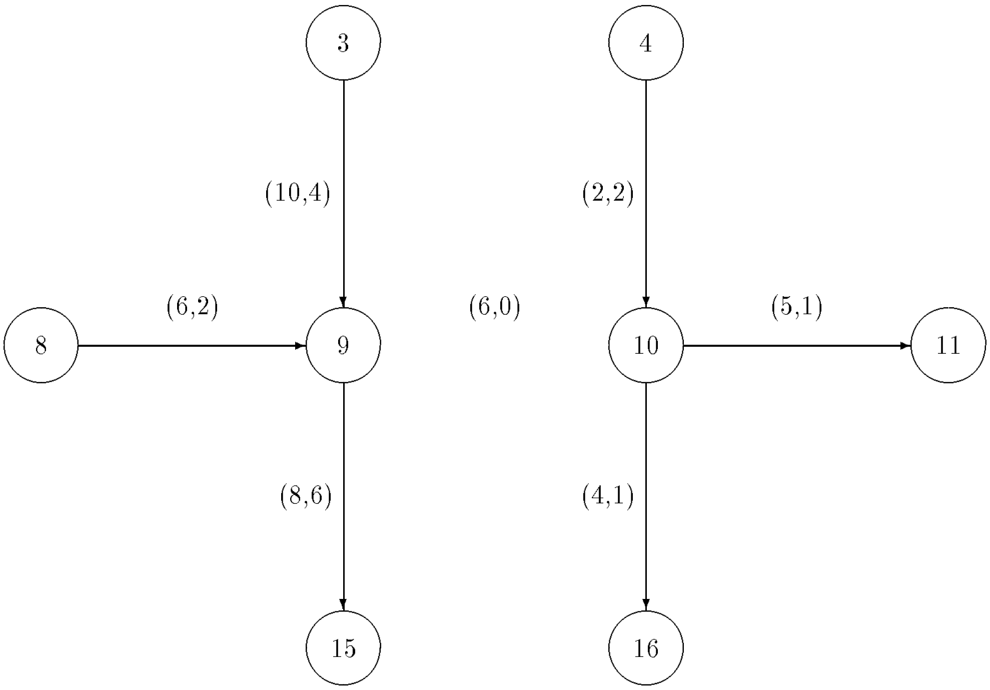


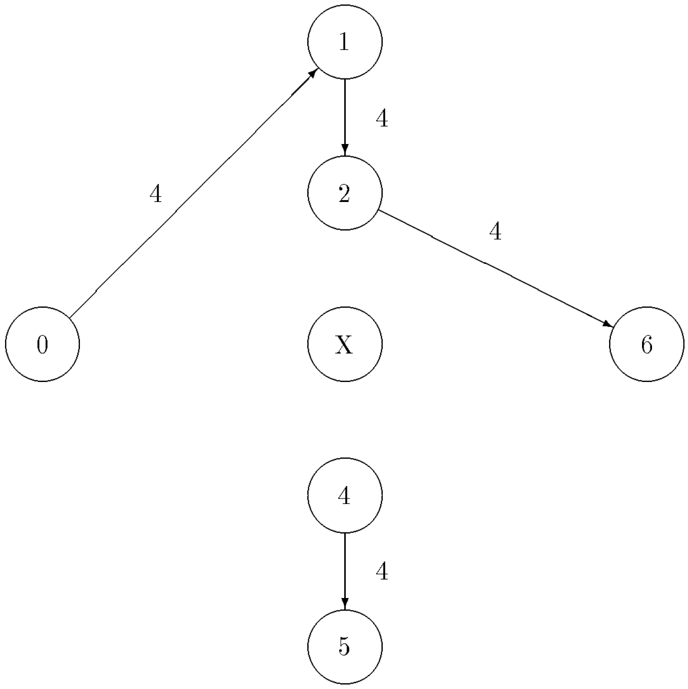
25
20
15
10
5
0
0.7
0.75
0.8
0.85
0.9
0.95
1
$P_L(q)$
q
$L = 24$
$L = 20$
$L = 16$
$L = 12$


ln(M)
ln(V)
0
1
2
3
4
5
6
7
8


0
1
2
3
4
5
(4,4)
(4,0)
(4,0)
(4,0)
(4,4)
(4,4)


3
4
(10,8)
(2,2)
(6,4)
(6,6)
(5,4)
8
9
10
11
(8,6)
(4,4)
15
16


3
4
(10,4)
(2,2)
(6,2)
(6,0)
(5,1)
8
9
10
11
(8,6)
(4,1)
15
16


1
4
4
2
4
0
X
6
4
4
5