

METHODS OF COMPUTER SIMULATIONS OF PHASE TRANSITIONS IN THE ASHKIN-TELLER MODEL

G. MUSIAŁ¹, L. DĘBSKI¹, G. KAMIENIARZ¹, J. ROGIERS²

¹*Institute of Physics, A. Mickiewicz University, ul. Umultowska 85, 61-614 Poznań, Poland*

²*Institute for Theoretical Physics, K. U. Leuven, Celestijnenlaan 200D, B-3001 Leuven, Belgium*

Abstract: The complementary Monte Carlo and series expansions methods of computer simulations have been described to investigate the critical behaviour of the Ashkin-Teller model in three dimensions. In the first method the invariance of the ratio of the square of the second moment of the order parameter to its fourth moment in the critical region has been exploited and some critical points on the phase boundaries have been calculated in the regions where the continuous transitions are expected. The continuity of the order parameter on the critical lines is verified by a finite size scaling analysis. Large-scale simulations have been performed on SGI Power Challenge XL and L supercomputers using the 64-bit random number generator. The numerically generated series expansions method is described for which the effective algorithm for generation of graphs starting from polygons and based on collapsing the unlinked vertices, is introduced. The new feature of our algorithm is that for each graph we introduce new links between unlinked vertices and we decorate bonds with new vertices, so that more complex graphs in an early stage of the graph generation procedure are obtained. The resulting set of graphs enables the application of the series expansions method and achievement of the precision of allocation of points on the phase diagram comparable to the precision of the Monte Carlo method, i.e. at least 3 decimal digits.

1. INTRODUCTION

Ashkin and Teller [1] have introduced a model of four-component system by generalization of Ising model, which can be interpreted as a two-component system. Fan has shown [2] that this model can be expressed in terms of Ising spins, with two spins s_i and σ_i at each lattice site (i.e. s_i and σ_i are variables that can take values $+1$ or -1). We will refer to it as standard Ashkin-Teller (SAT) model. This model can be interpreted as two superimposed Ising models. One of them is described in spin variables s_i and the other in variables σ_i and in both models there are exclusively two-spin interactions of a constant magnitude J_2 between the nearest neighbors only. Simultaneously, these two different models are coupled by four-spin interaction of a constant magnitude J_4 , also only between couples of spins residing at the nearest neighboring lattice sites. Thus, the Hamiltonian of this model is

$$-\beta\mathcal{H} = H = \sum_{\langle i,j \rangle} \{K_2 (s_i s_j + \sigma_i \sigma_j) + K_4 s_i \sigma_i s_j \sigma_j\}, \quad (1)$$

where $\beta = (k_B T)^{-1}$, $\langle i, j \rangle$ denotes summation over nearest neighboring lattice sites and $K_i = -J_i/k_B T$, with $i = 2$ or 4 .

If the third-order terms are included [3, 4], we obtain the extended Ashkin-Teller (EAT) model, which is described by the Hamiltonian

$$H = \sum_{\langle i,j \rangle} \{K_2 (s_i s_j + \sigma_i \sigma_j) + K_4 s_i s_j \sigma_i \sigma_j + K (s_i + s_j)(\sigma_i \sigma_j + 1)\}. \quad (2)$$

The SAT model corresponds to $K = 0$.

The properties of the EAT model were investigated within the molecular-field approximation (MFA) [3], as well as in two dimensions (2D) also within the molecular field renormalization group (MFRG) method [4] and the Monte Carlo (MC) method [5]. The SAT model in 2D and 3D was examined [6, 7] within the Monte Carlo methods and the short series analysis, yielding approximate phase diagrams.

The MFRG results [4] and the Monte Carlo simulations [5] do not confirm the existence of the tricritical points in the phase diagram of the EAT model which were present in the diagram calculated within MFA. Thus there is the problem of continuous transitions for the SAT model in 3D. The latter model seems to exhibit the tricritical point in the antiferromagnetic region [6] but both, the simulations and the series analysis supporting this picture, are too short to give a definite answer.

The aim of this paper is to prepare two complementary methods applicable to the 3D SAT model which ensure sufficient accuracy of the calculation of positions of the phase transition points on the (K_4, K_2) diagram. These are the Monte Carlo (MC) and the series expansions (SE) methods. The first method is widely used and allows controlling the precision of computations but simultaneously it is of stochastic character, whereas the latter, although giving the results of an asymptotic character, is one of the most reliable ones.

The simulations of MC type based on the invariance of the ratio of the square of the second moment of the order parameter to its fourth moment gave precise location of the phase transition point and critical exponents in the 3D Ising model [8, 9]. Using this fact we propose these simulations for the 3D SAT model in the critical region. We also present some preliminary results for phase boundaries between the paramagnetic phase and the ferromagnetic and the antiferromagnetic phases in the 3D SAT model.

In SE, the classification of the contributions to the terms of different order naturally leads to a description in terms of linear graphs whose bonds correspond to the strength of the interactions. For a review we refer the reader to the papers by Nickel [10] and Domb [11]. A linear graph is a collection of k points, called vertices, and i bonds, so that each pair of vertices is connected by at most one bond. The order of a graph is the number of its bonds, whereas the order of a vertex is the number of bonds meeting at this vertex. These graphs are fundamental in SE but are also used in a wide class of domains from fundamental mathematics to sciences and further to linguistics. One of the purposes of our paper is to obtain a complete list of graphs. This is a challenging and especially complicated problem for the 3D SAT model, to which also the graphs with odd vertex order contribute, in contrast to the situation in the Ising model.

Traditionally two standard methods of generation of graphs are used in the literature on the SE method. The first method, that starts from polygons as the first graphs, produces new graphs by collapsing vertices, whereas the second method, originally formulated for star graphs (a star graph is a connected graph with no vertex of the property that its removal together with its in-

cident bonds transforms the graph in a disconnected one) also starts from polygons and leads to new graphs by connecting vertices [12] (see also [10,11]). These methods were used successfully for the Ising model in which only vertices of even order appear. For the AT-model the first method allowed us to complete graphs up to 12-th order only, as it is discussed in section 4. Thus we had to look for a more efficient algorithm, since we intended to obtain series up to 14-th order.

We present here an effective algorithm for generation of closed connected linear graphs in any dimension which is the basic tool in SE and enables completing graphs up to the order 14 (and even greater). It is an extension of earlier algorithms [13] based on collapsing and connecting the unlinked vertices starting from polygons. The graphs form the basis for the numerical generation of SE for spin-lattice systems. The appearance of odd order vertices and consequently a great number of graphs, complicates also the next steps of series preparation, i.e. colouring and putting on the lattice. Thus, also the use of the numerically generated SE to calculate the free energy for the SAT model is explained and discussed in details. Estimation of the asymptotic behaviour of the SE of the free energy (and similarly other thermodynamic functions) gives the desired information about the location of a phase transition point.

2. DESCRIPTION OF THE MONTE CARLO METHOD

Our simulations of the Ashkin-Teller model have been performed on SGI Power Challenge XL supercomputers using the 64-bit random number generator. We have generated equilibrium configurations of the finite size samples of spins for fixed values of the model parameters. Periodic boundary conditions were imposed and thermalization of the initial configurations was applied.

One possible strategy for sampling the configurations of a system is to choose configurations completely at random. However there is a serious problem with this approach. If we include with equal weights the configurations which bring negligible contributions to the quantities as the partition function, at the same time these configurations bring a great unphysical contribution to the variance of the internal energy, which leads to serious errors at low temperatures. We can alleviate this problem by the so-called importance sampling in which the program takes into account the largest possible number of the configurations which bring great contribution to the quantities we are trying to calculate.

Gibbs distribution was sampled using the Metropolis algorithm [14], We started with some initial configuration α of spins, and a new configuration α' of the system was generated from α by the repetitive application of the importance sampling procedure when flipping successive spins on the lattice. The spins may be selected at random, or each of the spins in the sample may be reversed in turn (which is the case here). We either reverse the spin or not, according to some initially chosen transition probability, and when each spin of the sample has been visited once (on the average or consecutively) we carried out one Monte Carlo step (MCS) per spin. Either of these procedures ensures the satisfaction of the accessibility criterion, which states that it must be possible to evolve the system from a given starting point to any of its other configurations by applying the evolution rule a sufficiently large number of times.

In order to decide whether to accept a single spin-flip or not, we compared the energies of the new and old configurations. If the energy change $E_{\alpha'} - E_{\alpha}$ was negative, then the new configuration was automatically accepted; if, however, it was positive, the new configuration was accepted with a probability $e^{-\beta(E_{\alpha'} - E_{\alpha})}$. Physically it means that both configurations are in equilibrium and none of them arises at the expense of the other. Using this method, we generated configurations which allowed us to calculate physical quantities in a direct way.

The phase transition point T_c was determined [5, 8] from the analysis of the fourth order cumulant

$$Q_L = \frac{\langle M^2 \rangle_L^2}{\langle M^4 \rangle_L}, \quad (3)$$

where $\langle M^n \rangle$ denotes the n -th power of the order parameter averaged over an assembly of independent samples of the size $L \times L$ or $L \times L \times L$ in 2D and 3D, respectively. For $T > T_c$ and $L \gg \xi$ where ξ denotes the correlation length, Q_L tends towards 1/3 which corresponds to a Gaussian distribution. For $T < T_c$ and $L \gg \xi$, Q_L tends to 1. For $L \ll \xi$, Q_L varies only weakly with temperature and linear dimension, and stays very close to the value 0.856 [8] in two dimensions and to 0.6233 [8, 9] in three dimensions - the values achieved at the critical point in the limit $L \rightarrow \infty$. This number is characteristic of the corresponding Ising universality class. This behaviour of the cumulant is useful for determination of T_c . One may plot Q_L versus T for various L 's and estimate T_c from the intersection point of these curves (see [5] and the papers cited therein).

In order to confirm that the transition is of the Ising type we can use the finite size scaling relation [15] between magnetization m , temperature T and size of the system L :

$$m(T, 1/L) = L^{-\beta/\nu} f[(T - T_c)L^{1/\nu}], \quad (4)$$

where f is the scaling function which depends on the boundary conditions. If we expand this equation to the first order for T close to T_c we have

$$L^{\beta/\nu} m(T - T_c, 1/L) = a_1 + a_2 (T - T_c) L^{1/\nu}, \quad (5)$$

where the amplitudes a_1 and a_2 are temperature and size independent.

3. THE SERIES EXPANSIONS METHOD

One way to obtain the information on behaviour of physical quantities is to express them in the high- or low-temperature series expansions, and to estimate the asymptotic behaviour from these series. The longer the series the more accurate the description.

As an illustration of the calculations of thermodynamic functions, we explain how to calculate the reduced free energy of the SAT model

$$f = -\frac{F}{Nk_B T} = \frac{1}{N} \ln Z, \quad (6)$$

using the SE numerical technique. Here F is the free energy and N is the number of sites in the

lattice. The most important quantity which allows us to calculate the free energy (and the other thermodynamic functions) is the partition function Z

$$Z = \text{Tr} e^{-\beta\mathcal{H}} = \text{Tr} \prod_{\langle i,j \rangle} e^{K_1 \sigma_i \sigma_j + K_2 s_i s_j + K_4 \sigma_i s_i \sigma_j s_j}, \quad (7)$$

where we will later put $K_1 = K_2$. For further calculations we perform the following transformation

$$e^{K_1 \sigma_i \sigma_j + K_2 s_i s_j + K_4 \sigma_i s_i \sigma_j s_j} = a + b \sigma_i \sigma_j + c s_i s_j + d \sigma_i s_i \sigma_j s_j \quad (8)$$

and we use the fact that $\sigma_i, s_i = \pm 1$, which allows us to write down all nonequivalent equations (there are 4 such equations) and to obtain the expressions for the constants a, b, c, d in terms of the couplings K_1, K_2, K_4 , e.g.

$$a = \text{ch } K_1 \text{ ch } K_2 \text{ ch } K_4 + \text{sh } K_1 \text{ sh } K_2 \text{ sh } K_4. \quad (9)$$

Thus, we have

$$Z = \text{Tr} a^{Nz/2} \prod_{\langle i,j \rangle} (1 + v_1 \sigma_i \sigma_j + v_2 s_i s_j + v_4 \sigma_i s_i \sigma_j s_j), \quad (10)$$

where $v_1 = b/a, v_2 = c/a, v_4 = d/a, z$ is the coordination number of the lattice.

After multiplication of all terms in the product in Eq. (10) we obtain a sum of other terms that contain some powers of spins, v_1, v_2 and v_4 . The trace can be calculated separately for every term and then only these terms in which spins appear in even powers give nonzero contribution to the sum. This means that spins contribute a factor 1 in every nonvanishing term. We can order the terms with increasing p (p is the sum of powers at v_1, v_2 and v_4) and truncate the sum rejecting terms with $p > p_{\max}$, which should be chosen high enough to obtain the desired precision of computation.

Such an ordering of terms leads naturally to a description in terms of linear graphs in which bonds correspond to these interactions between spins which are specified in the Hamiltonian. The fact that spins appear only with even powers means that only closed graphs give nonzero contribution to the free energy.

Using graph representation we can write the partition function (10) in the form

$$Z = a^{Nz/2} 2^{2N} \left(1 + \sum_{\mathbf{g}} N^{i(\mathbf{g})} l(\mathbf{g}) v_1^{j(\mathbf{g})} v_2^{n(\mathbf{g})} v_4^{m(\mathbf{g})} \right), \quad (11)$$

where the summation runs over all nonisomorphic graphs \mathbf{g} in increasing order, i, j, n, m are integer numbers (their sense is explained in the next paragraph) and l is the lattice constant (i.e. a number of nonequivalent puttings around 1 site of the lattice divided by a symmetry factor of the graph) of a graph \mathbf{g} . The value of the number $i(\mathbf{g})$ of nonequivalent puttings of one graph on different sites of the lattice depends on the geometry and the size of the lattice. As discussed above, below Eq. (10), we sum over all graphs with orders lower or equal to p_{\max} . In the section 4 we propose the effective algorithm of generation of graphs.

In the next step graphs are coloured, i.e. a specific spin-spin interaction together with a coupling constant is assigned to each bond in the graph, which is connected with the choice of a specific model. This assignment has to be made in every possible way to simulate each possible

configuration of the system in SE. A coloured graph g contains j bonds representing σ - σ interaction and each interaction introduces the factor v_1 to the term that is represented by this graph in Eq. (11). Similarly in this graph we have n bonds which represent s - s interactions and the factors v_2 . The remaining m bonds in this graph represent σ - s interactions introducing the factors v_4 to the term under consideration. Thus the order p of the graph is $p = j + n + m$, as mentioned above. Here also graphs with odd number of bonds joined to a vertex are possible to get coloured, which means that we have many more graphs when compared to those used for the Ising model.

Following Eqs (6) and (11) and expanding \ln we get the final expression for computation of the reduced free energy

$$f = 2 \ln 2 + \frac{z}{2} \ln a + \sum_g l(g) v_1^{j(g)} v_2^{n(g)} v_4^{m(g)}. \quad (12)$$

When we put $K_1 = K_2$, as it should be for SAT model, we obtain $b = c$ and of course $v_1 = v_2$.

Table I. The scheme of the algorithm of graph generation

INPUT:	N_{max} – the maximum order of graphs to generate; A – the maximum order of a vertex in a graph.
OUTPUT:	array GRAPHS with labels and numbers of vertices of all generated graphs.
Step 1	Generate polygons from order 3 up to N_{max} , calculate their labels and store them and the number of their vertices as the first graphs in the array GRAPHS. Set $N_{gr} = N_{max} - 2$. (N_{gr} is the number of hitherto generated graphs) Set $K = 1$. (K is the number of the graph considered from the array GRAPHS)
Step 2	While $K \leq N_{gr}$ do Steps 3-4.
Step 3	Subroutine NewGraphs (K , array GRAPHS, N_{gr} , N_{max}).
Step 4	Set $K = K + 1$. (consider the next graph from the array GRAPHS)
Step 5	Output (the array GRAPHS). STOP.

4. THE ALGORITHM FOR GENERATION OF CLOSED CONNECTED LINEAR GRAPHS

We propose here the algorithm which generates all possible closed connected linear graphs of the i -th order, which for $i \leq N_{comp}$ are completed, when the order of a vertex is limited to a certain valued, i.e. the number of bonds joined at a vertex is lower or equal to A . No limiting case is achieved by putting $A = N_{max}$, where N_{max} is the maximum order of the graphs to be generated. A and N_{max} are parameters of the algorithm as described in Table I. If we label the vertices of a graph as $1, 2, \dots, k$, it can be characterized by its adjacency matrix. This is the $(k \times k)$ -matrix with matrix elements $M_{ij} = M_{ji} = 1$ if the vertices are connected by a bond, and $M_{ij} = M_{ji} = 0$, if not, and with diagonal elements equal to zero.

We generate graphs starting from polygons which are the simplest graphs, and write their labels to the graph list. Call these the starting graphs. Next we successively consider each graph

from the list to produce new graphs, which also are added to this list as new graphs. The scheme of the algorithm is illustrated in Table I, whereas the subroutine NewGraphs that generates new graphs from the already obtained one and stored in array GRAPHS, is shown in Table II. In the first step of graph processing (Table II) we choose all possible pairs of vertices that are not directly connected. This is achieved by checking if the (i,j) element of the graph adjacency matrix is 0. Then the subroutine splits into two independent paths. For every pair of not directly connected vertices: (a) we collapse a vertex with a higher label (which then disappears from the graph labelling) onto one with a smaller label (Step 3 in Table II), forming a possible new graph (we will refer to this as the collapsing procedure); (b) we connect these vertices with additional bonds (Steps 4-6 in Table II), forming other possible new graphs (we will refer to this as the connection procedure). This means that we have to generate graphs up to some higher order N_{max} to find all possible graphs up to the required order N_{compl} .

Table II. The scheme of the subroutine NewGraphs (K , array GRAPHS, N_{gr} , N_{max}) for generation of new graphs from K -th graph in array GRAPHS

For each pair of vertices (i, j) with $i > j$ in K -th graph do:	
Step 1	Take the first pair of vertices (i, j) in K -th graph.
Step 2	If $M_{ij} = 0$ then do Steps 3-4, else go to Step 7. (M is the adjacency matrix of K -th graph)
Step 3	Collapse the i -th vertex (which disappears from the graph labeling) onto the j -th one in the K -th graph. If there is a vertex of order 1 or of order greater than A in the new graph, then go to Step 4, else calculate the label of this graph and check whether it exists in the array GRAPHS; if it exists then go to Step 4, else set $N_{gr} = N_{gr} + 1$, store its label and number of vertices as N_{gr} -th graph in the array GRAPHS, go to Step 4.
Step 4	Calculate N_b . (N_b is the order of the K -th graph) Set $D = N_{max} - N_b - 1$. If $D < 0$ then go to Step 7, else in K -th graph connect the vertices i and j with the additional bond; calculate the label of the new graph and check whether it exists in the array GRAPHS; if it does not exist then set $N_{gr} = N_{gr} + 1$, store its label and number of vertices as N_{gr} -th graph in the array GRAPHS. If $D = 0$ then go to Step 7, else set $l = 1$; while $l \leq D$ do Steps 5-6.
Step 5	In K -th graph connect vertices i and j with the additional bond and introduce l further vertices on this bond; calculate the label of this graph and check whether it exists in the array GRAPHS; if it exists go to Step 6, else set $N_{gr} = N_{gr} + 1$, store its label and number of vertices as N_{gr} -th graph in the array GRAPHS, go to Step 6.
Step 6	Set $l = l + 1$.
Step 7	Take the next pair (i, j) of vertices with $i > j$ in K -th graph, if it exists then go to Step 2, else return.

In order to identify the graphs it is important to give just a single label to isomorphic graphs. This is realized using a canonical labelling [11, 12, 16]. The label of the graph is the vector composed of the overdiagonal elements of matrix M read by rows starting from the left to right and from the top to bottom. The vector, consisting of bits 1 and 0, forms a binary number. By the graph label or graph key we mean the maximum binary number we can obtain by relabelling the vertices, e.g. by considering the $k!$ permutations of the vertex labellings. This is a unique way for labelling the graphs. All isomorphic graphs get the same label or key. Graphs with different keys are topologically different.

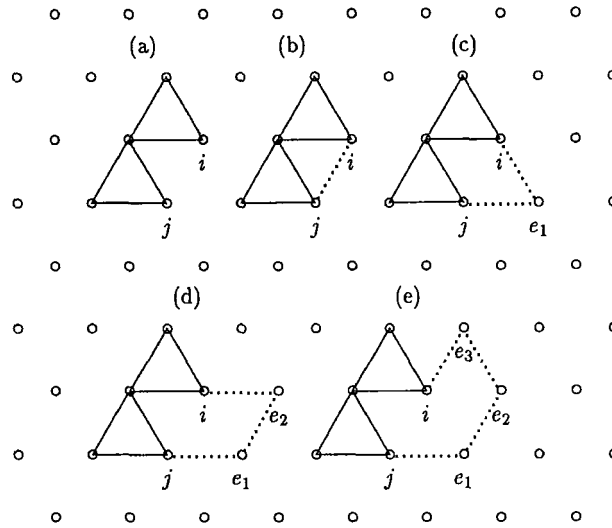


Fig. 1. Illustration of the extension of the connection procedure performed in Steps 5 and 6 on the scheme in Table II. For simplicity the exemplary graphs are put on the triangle lattice.

Using the key we check if the newly generated graph, obtained by the collapsing or connection method, is present in our graph list (array GRAPHS in the algorithm). If not, the new graph is added at the end of the list of graphs. The algorithm stops when all graphs from the list have been submitted to the collapsing and connection procedures and we do not obtain a new graph from the last one in the list.

Our idea is not only to bring together these two standard procedures of graph generation, additionally we propose in procedure (b) to connect vertices not only with a simple line, but with a line decorated with additional vertices. Thus path (b) in fact splits into many paths. In the first one (Step 4 in Table II) we introduce a simple diagonal in the graph and we obtain the first new graph. In the next path (Steps 5-6 in Table II) we introduce one additional vertex on this diagonal and another new graph arises. Then we subsequently put two and more vertices on the diagonal obtaining additional new graphs. This procedure stops when the number of vertices in new graph reaches N_{max} .

Table III. The numbers of generated closed connected linear graphs of higher orders in terms of the values of N_{max} - the maximum allowed order of graphs, when applying collapsing procedure only. For boldfaced numbers all graphs of a given order are completed. The maximum number of bonds joined at a vertex is 6 (as for a simple cubic lattice). The numbers in italic are not verified which means that they contain a few percent of double graphs

N_{max}	The order of generated graphs									
	8	9	10	11	12	13	14	15	16	17
10	22	29	19							
11	22	54	64	37						
12	22	60	142	162	85					
13	22	60	163	382	418	181				
14	22	60	164	486	1137	1112	452			
15	22	60	164	492	1548	3435	3115	<i>1211</i>		
16	22	60	164	493	1607	5222	10859	<i>10061</i>	3396	
17	22	60	164	493	1613	5504	18347	<i>38653</i>	<i>31078</i>	<i>9765</i>

Table IV. The numbers of generated closed connected linear graphs of higher orders in terms of the values of N_{max} - the maximum allowed order of graphs, when applying our procedure. For boldfaced numbers all graphs of a given order are completed. The maximum number of bonds joined at a vertex is 6 (as for a simple cubic lattice). The completeness of data with * should be confirmed by calculation with $N_{max} = 18$. The numbers in italic are not verified which means that they contain a few percent of double graphs

N_{max}	The order of generated graphs									
	9	10	11	12	13	14	15	16	17	
10	58	152								
11	59	160	445							
12	60	162	480	1499						
13	60	164	490	1583	5289					
14	60	164	493	1607	5449	18365				
15	60	164	493	1610	5515	19996	<i>77954</i>			
16	60	164	493	1614	5542	20259	<i>85796</i>	<i>334789</i>		
17	60	164	493	1614	5542	20267*	<i>86459</i>	<i>349062</i>	<i>1362236</i>	

This extension of the standard connection procedure is illustrated in Fig. 1. The original graph (K -th graph selected in Step 2 of the algorithm, Table I) is (a). Its bonds are drawn with the solid line; (i, j) is the pair of vertices considered in Step 2 of Table II. The next graph (b) is this in which one extra bond (the dotted line) is included to connect the vertices i and j (Step 4 in Table II). Moreover, the next three graphs (c-e) illustrate the effect of our extension of the connection procedure in Steps 5 and 6 from Table II, when $N_{max} = 10$. 1, 2 and 3 extra vertices (denoted as e_1, e_2, e_3 in Fig. 1) are introduced on the dotted line connecting the vertices i and j . One should remember that these new graphs will also be subjected to the connection procedure.

To find a complete list of all graphs up to some order N_{compl} is a problem. The basic difficulty is to obtain graphs with a very complicated structure. When we introduce additional vertices on

the diagonal connecting two not directly connected vertices in the graph considered we quickly generate new complicated graphs, since these graphs will also undergo this procedure. Thus we reach N_{compl} at a lower value of N_{max} .

It is obvious that N_{max} is the critical parameter in the process of the generation of graphs. The realization of a high N_{max} is limited not only by the amount of operational memory required (writing on and reading from hard disks consumes a prohibitively long time), but first of all by the computing time needed for comparing the keys. Because the number of graphs increases rapidly when N_{max} increases, this comparison time becomes prohibitively long.

We implemented our algorithm in FORTRAN 77 to be able to use variables of integer type with the length of 1 byte. The basic element in the algorithm is the list of labels of the already generated graphs which takes far more than 90% of the memory used by the program. In our implementation it is an array GRAPHS which is just of the integer*1 type.

For the largest graphs we needed 136 digits to represent a graph label. For our purposes it is the most convenient form of a graph label because it allows one to colour the bonds of a graph numerically (colouring of bonds is assigning a specific spin-spin interaction together with a coupling constant to every bond in the graph as it is explained in section 3). As mentioned above, in the SAT model there are three different colours of bonds in graphs representing three types of interactions that appear in this model. The most convenient way to represent a colour for a given bond is to set the non-zero elements of the adjacency matrix equal to 1, 2 or 3, and this notation is used for the graph label.

It is also possible to put the key of a graph in a binary form. Then one needs only 3 eight-bytes variables to keep a graph label in memory. This reduces the memory needs by a factor of 5.6. It was not necessary to apply it to this problem because we did not use the whole accessible memory in our computations. For the run with $N_{max} = 17$ our program used only about 150 MB of RAM, but for computation with a greater value of N_{max} one should switch to this way of storing the labels of generated graphs. We might also expect a reduction of time consumed for the comparison of graphs.

The special problem was also implementation of canonical labelling of the newly generated graphs. Considering the $n!$ possibilities of numbering n vertices in a graph was too time consuming. We have constructed our own procedure that starts from giving number 1 to a vertex of the maximum order and all possible vertices are taken into account. Using the procedure with this condition all generated graphs were verified for elimination of double graphs up to 14-th order in data presented in Tables III-V, but even the use of this subroutine to calculate the labels of new graphs when generating them takes too much computing time. We had to construct another procedure of less complexity (i.e. with a lower number of operations) using some topological properties of graphs (for the review see [17] and the papers cited therein). We obtained a reasonable time of graph generation but as a result of this simplification the final list contains a few percent of double graphs which were eliminated as we explained above.

Table V. The numbers of generated closed connected linear graphs of higher orders in terms of the values of N_{max} - the maximum allowed order of graphs, when applying connection procedure only. The maximum number of bonds joined at a vertex is 6 (as for a simple cubic lattice). The numbers in italic are not verified which means that they contain a few percent of double graphs

N_{max}	The order of generated graphs									
	8	9	10	11	12	13	14	15	16	17
10	11	26	65							
11	11	26	65	180						
12	11	26	65	180	560					
13	11	26	65	180	560	1871				
14	11	26	65	180	560	1871	6732			
15	11	26	65	180	560	1871	6732	26485		
16	11	26	65	180	560	1871	6732	26485	107221	
17	11	26	65	180	560	1871	6732	26588	107274	452841

5. RESULTS AND CONCLUSIONS

5.1. The Monte Carlo simulations

To present the MC method, we have performed our simulations for the SAT model on the 3D simple cubic lattice and for the regions of the parameter space (K_4, K_2) where the phase transitions are expected to be continuous, obtaining some preliminary results. We have used (so) as an order parameter M . In practice, as suggested by Binder [15], we calculated root mean square (RMS) value of the order parameter to avoid trouble with a restricted ensemble.

To improve the accuracy we have used the following finite size scaling law (see [15])

$$M_{RMS}(T = T_c) \propto L^{-\beta/\nu}, \quad (13)$$

which follows from Eq. (5). This relation suggests that $L^{\beta/\nu} M$ should approach a straight line independent of L , if we have chosen the proper values for T_c and the critical exponents. In addition, in the region of continuous phase transitions the Ashkin-Teller model belongs to the same universality class as Ising model for which the critical exponents are known. Making use of this fact we have put $\beta = 0.6301$ and $\nu = 0.3267$ [8],

An example of such an analysis for the SAT model is shown in Fig. 2, where we have plotted the L -dependence of $L^{\beta/\nu} M_{RMS}$ with $K_4 = -0.34$ i.e. in the antiferromagnetic region. For this region the sublattice magnetization (calculated by storing the data from one of the two equivalent sublattices) is taken into account as the order parameter M_{RMS} (as mentioned above). For a properly chosen value of K_2 , here it is $K_2 = 0.334$, and for $L \geq 14$ the data oscillate around a constant value within the experimental error, which means that the two above values of the coupling constants are the coordinates of the critical point (K_4^c, K_2^c) on the phase diagram. Moreover, it confirms the continuous character of the phase transition.

For the 3D SAT model we have investigated the ferromagnetic and antiferromagnetic regions of the parameters (K_4, K_2) with continuous transitions to the paramagnetic phase, using systems with linear size $L \leq 26$. The corresponding estimates of the critical couplings K_2^c as a function of

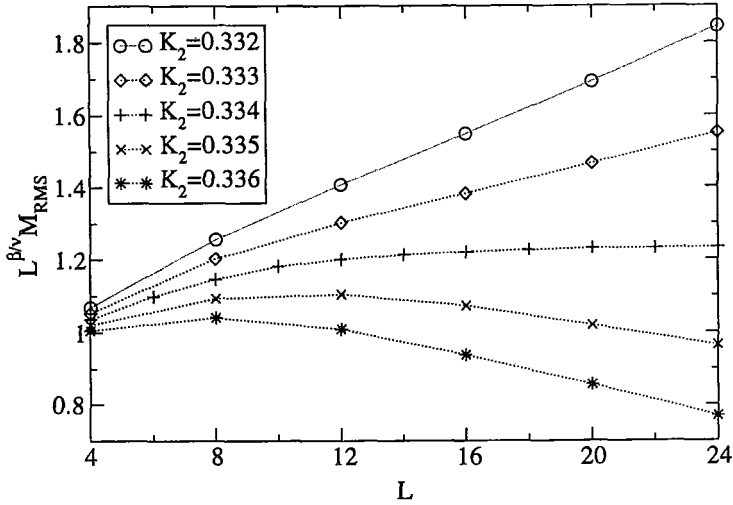


Fig. 2. Size-scaled values of the order parameter $L^{\beta/\nu} M_{RMS}$ calculated for $K_4 = -0.34$ and for different values of linear size L of the cluster. The values of the Ising critical exponents are assumed

K_4^c are presented in Table VI. Our results for K_2^c , (denoted as K_2^f, K_2^{af} for the ferro- and antiferromagnetic region, respectively) are given in the second and the fifth column whereas the previous estimates K_2^D of Ditzian et al. [6] are displayed in the third and the sixth column.

In the Ising limit $K_2 = 0$ we have estimated the critical coupling at $K_4^c = 0.2215(4)$ which agrees within the error bars with that found by Talapov et al. [9], using the Cluster Processor and newly constructed special-purpose computer.

Table VI. The ferromagnetic and antiferromagnetic critical couplings for the continuous phase transitions in the 3D SAT model

K_4^c	K_2^f	K_2^D	K_4^c	K_2^{af}	K_2^D
0.220	0.040	0.05	-0.230	0.090	0.05
0.217	0.066	0.06	-0.242	0.141	0.12
0.213	0.087	0.078	-0.266	0.208	0.21
0.210	0.0996	0.10	-0.313	0.296	
0.203	0.118	0.11	-0.340	0.334	
0.195	0.133	0.13	-0.360	0.359	0.36

Our simulations have been performed on SGI Power Challenge supercomputers with the use of 32- and 64-bit random number generators for the system sizes L up to 26 for 3D SAT model. CPU time used for the simulations varied from a few minutes for the smallest L values up to 48.6 days for the SAT model with $L = 26$. A typical duration of our simulation was between 10 and 60 hours of CPU time.

In the simulations we always apply thermalization of the initial configurations and it is enough to take 10^5 MCS for this purpose. Then we calculate partial averages of the cumulant Q usually

after each $10 \times 5 \times 10^5$ MCS. The factor 10 reflects the fact that for the calculation of a partial average of Q we take results only every 10-th MCS to avoid correlations between sampled configurations of spins in the system. Final averages as well as errors were calculated mainly from five to six partial averages. Experimental error never exceeded 1% of the value of the

cumulant Q , while for the critical values of the parameters, it was less than 10^{-3} when a finite size scaling analysis was performed; if not, the error was less than 6×10^{-3} .

In one single MCS our computer program visited lattice sites one after the other and for each site two independent decisions were taken, namely, whether to accept the flip of spin s and for spin o . In the antiferromagnetic phase the system is composed of two sublattices and magnetization of one of them was taken as an order parameter. Each of the two sublattices was divided into four identical subsublattices for 3D SAT model. The program visited all sites of one subsublattice then continuing over another one.

5. 2. The series expansion method

Application of the SE method to the 3D SAT model with a precision comparable to that of MC method is limited by the disposal of the closed linear graphs completed up to the required order, as presented in section 3. Algorithms presented in literature allow completion of these graphs up to the 12 order in real time, which is the consequence of the fact that also graphs with odd order vertices contribute to the SAT model. Thus, we had to look for a more efficient algorithm, since we intended to complete graphs up to the 14-th order. We only have to complete closed connected linear graphs, because all multicomponent ones can be obtained as simple combinations of the connected ones.

The numbers of generated closed connected linear graphs of higher orders in terms of the values of N_{max} - the maximum allowed order of graphs when applying standard procedure (which is called the collapsing procedure here) and our procedure are presented in Tables III and IV, respectively. Our procedure allows the user to complete graphs up to a much higher order than the standard procedure at the same value of parameter N_{max} , which obviously is the critical parameter of the cost of calculations. At $N_{max} = 15$ the difference is only 1 order, but the effectiveness of our procedure is clearly seen at a higher N_{max} , because our procedure is partially a combination of the collapsing and connection procedures. For the sake of comparison we present also the results for the latter in Table V. They are rather poor when compared to the results shown in Tables III and IV, due to the fact that some graphs cannot be generated in this way.

We also compared the results of our algorithm presented in Table IV to those of Balińska and co-workers [17]. They call these graphs the connected f -graphs with $f=6$ and the minimum vertex order 2. We collected in Table VII the numbers of these graphs for the particular numbers of bonds and numbers of vertices. We found no differences up to the order 11, up to which the graphs in [17] were completed from our point of view, i.e. when ordering them by the number of bonds in the graphs. The boldfaced numbers have not been known in the literature yet and are consistent with the asymptotic estimates of Balińska [18] and with those of Bender, Canfield and McKay [19] (and the papers cited therein).

Table VII. The numbers of all connected f -graphs with $f=6$ and the minimum vertex order 2 (i.e. closed graphs - no end vertices). N_b denotes the number of bonds

N_b	The number of vertices												Sum
	3	4	5	6	7	8	9	10	11	12	13	14	
3	1	0	0	0	0	0	0	0	0	0	0	0	1
4	0	1	0	0	0	0	0	0	0	0	0	0	1
5	0	1	1	0	0	0	0	0	0	0	0	0	2
6	0	1	3	1	0	0	0	0	0	0	0	0	5
7	0	0	3	5	1	0	0	0	0	0	0	0	9
8	0	0	2	11	8	1	0	0	0	0	0	0	22
9	0	0	1	15	31	12	1	0	0	0	0	0	60
10	0	0	1	12	63	71	16	1	0	0	0	0	164
11	0	0	0	8	89	230	144	21	1	0	0	0	493
12	0	0	0	5	97	507	703	274	27	1	0	0	1614
13	0	0	0	2	82	825	2256	1869	474	33	1	0	5542
14	0	0	0	1	59	1072	5399	8445	4469	785	36	1	20267

One can propose richer forms than a line decorated with vertices to connect the vertices - Steps 5 and 6 in the scheme presented in Table II. For instance, the simplest graphs can also be used for vertices connection, i.e. when graph (c) in Fig. 1 has an extra dotted bond between the vertices i and j , or when graph (e) has an additional dotted bond between the vertices e_1 and e_3 then, the i and j vertices are connected via a triangle. There are two reasons against such an extension of the algorithm: a) these forms of graphs will appear in the later steps of the algorithm, b) it introduces a great number of comparisons of the graph keys. Our procedure is a compromise between the degree of complication of the algorithm and the operational memory needs. The size of memory needs may be reduced by the use of a binary representation of graph keys, as we indicated in section 4. There is the only one critical parameter - time of computation. Thus, we have to minimise the number of calculations and comparisons between graph keys in the algorithm. Our algorithm is the right step in the direction towards reducing the algorithm's complexity (i.e. the number of operations to obtain the required value of N_{compl}).

After the set of closed connected linear graphs has been completed, we have to generate and add to it all multicomponent graphs up to the required order N_{compl} . We complete the multicomponent graphs regarding each possible combination of the connected ones and calculating the canonical label of each obtained graph. Each new graph is compared to each previously obtained graphs before storing to avoid double graphs.

The resulting set of graphs should be arranged according to increasing orders of the graphs. In the next two steps of the numerically generated SE method these graphs are coloured and put on the lattice. These steps and the application of the method to the 3D SAT model are explained in section 3. Now the series expansion is completed and is of the form shown by Eqs (11) and (12). Taking particular values of the model parameters we can estimate the asymptotic behaviour from these series in the standard way [11] (and the papers cited therein) to obtain the information on behaviour of the model in the critical region. The longer the series the more accurate the

description. Using our procedure of graphs generation one can complete graphs even up to the order $N_{max} = 15$ in a real time, which should be compared to $N_{max} = 10$ used in [6] (the 11-th order was not completed).

5. 3. Concluding remarks

In the 3D SAT model our preliminary MC results are more accurate than those of Ditzian et al. [6] and suggest that also the run of the line of the continuous phase transitions in the antiferromagnetic part of the phase diagram is different from that of these authors. It suggests that this line may be extrapolated similarly to that found previously [20]. As to the tricritical point in the 3D SAT model, its existence has not been absolutely confirmed yet. The simulations in the vicinity of non-Ising universality class have to be performed extremely carefully. The estimates given in Table VI are calculated from the corresponding 6 partial averages, each consisting of several millions of MCS per spin.

The resulting set of graphs is the basic element for the application of the numerically generated SE, which are necessary to complete the analysis of the 3D version of many spin-lattice systems like the SAT model with a sufficient accuracy. Here we expect the precision of allocation of points on the phase diagram comparable to the precision of the Monte Carlo method, i.e. at least 3 decimal digits. The SE analysis of the 3D SAT model is in preparation. It is worth noting that the procedure of generation of graphs which is proposed here is independent of the choice of the model and the lattice geometry, thus this result is of general character.

Using our experience in studying the 2D EAT model [4, 5], the 3D Ising model [8], the methods and the algorithms presented here, we plan to consider the 3D SAT model, whose phase diagram is still not fully resolved and rises many interesting points as discussed above.

Acknowledgements

The authors wish to thank Dr. K. Balińska and M. Kupczyk M. Sc. Eng. for valuable discussions and private communication concerning comparison of the results. Thanks are also due to the Poznań Supercomputing and Networking Center for the access to SGI Power Challenge XL and L supercomputers, on which simulations have been performed. Partial financial support from the State Committee for Scientific Research within the grant 8T11F 015 18 is also acknowledged.

References

- [1] J. Ashkin, E. Teller, *Phys. Rev.*, **64**, 178 (1943).
- [2] C. Fan, *Phys. Lett.*, **39A**, 136 (1972).
- [3] P. Pawlicki, J. Rogiers, *Physica*, **A214**, 277 (1995).
- [4] P. Pawlicki, G. Musiał, G. Kamieniarz, J. Rogiers, *Physica*, **A242**, 281 (1997).
- [5] P. Pawlicki, G. Kamieniarz, L. Dębski, *Physica*, **A242**, 290 (1997).
- [6] R. V. Ditzian, J. R. Banavar, G. S. Grest, L. P. Kadanoff, *Phys. Rev.*, **B22**, 2542 (1980).
- [7] P. Arnold, Y. Zhang, *Nuclear Phys.*, **B501**, 803 (1997).
- [8] G. Kamieniarz, H. W. J. Blöte, *J. Phys. A: Math. Gen.*, **26**, 201 (1993).
- [9] H. W. J. Blöte, E. Luijten, J. R. Heringa, *J. Phys. A: Math. Gen.*, **28**, 6289 (1995).
- [10] A. L. Talapov, H. W. J. Blöte, *J. Phys. A: Math. Gen.*, **29**, 5727 (1996).

- [11] B. G. Nickel, J. J. Rehr, *J. Stat. Phys.*, **61**, 1 (1990).
- [12] C. Domb, *Graph theory and embeddings*, in: *Phase Transitions and Critical Phenomena*, ed. by C. Domb and M. S. Green, Academic, New York 1974, Vol. 3, p. 1.
- [13] J. L. Martin, *Computer techniques for evaluating lattice constants*, *ibidem*, p. 97.
- [14] B. R. Heap, *J. Math. Phys.*, **7**, 1582 (1966).
- [15] G. Musiał, *Comp. Meth. Sei. Techn.*, **4**, 75 (1998).
- [16] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, E. Teller, *J. Chem. Phys.*, **21**, 1087 (1953).
- [17] K. Binder and D. W. Heerman, *Monte Carlo Simulation in Statistical Physics*, Springer Series in Solid State Physics, Springer-Verlag, Berlin, Heidelberg 1988.
- [18] J. Nagle, *J. Math. Phys.*, **7**, 1588 (1966).
- [19] K. T. Balińska, J. Miądowicz, *Graph Theory of New York*, XXIX, 33 (1995);
<http://www.man.poznan.pl/~balinska/>
- [20] K. T. Balińska, M. Kupczyk, private communication.
- [21] E. A. Bender, E. R. Canfield, B. D. McKay, *J. Combinat. Theory*, **A80**, 124 (1997).
- [22] R. J. Baxter, *Exactly Solvable Models in Statistical Mechanics*, Acad. Press, London 1982.