Cluster–Exact Approximation of Spin Glass Groundstates

Alexander K. Hartmann *
Institut für Theoretische Physik
Ruprecht–Karls–Universität Heidelberg
Philosophenweg 19
69120 Heidelberg, Germany
(July 2, 1998)

PACS: 75.10.H, 75.10.N, 02.10

We present a fast ($\sim O(N^3)$) algorithm which calculates groundstates of Ising spin glasses approximately. It works by randomly selecting clusters of spins which exhibit no frustrations. The spins which were not selected, contribute to the local fields of the selected spins. For the spin–cluster a groundstate is exactly calculated by using graphtheoretical methods. The other spins remain unchanged. This procedure is repeated many times resulting in a state of low energy. The total time complexity of this scheme is approximately cubic. To demonstrate how deep in energy this algorithm can get, we applied it to the $\pm J$ model. We estimate that the groundstate energy density of the infinite system is -1.400 ± 0.005 (2d) and -1.766 ± 0.002 (3d). Finally the distribution of overlaps for selected systems is calculated in order to characterize the algorithm.
Key words: Spin Glasses, Edwards–Anderson Model, Groundstates, Spin Clusters, Graph Theory, Minimum Cut, Maximum Flow.

I. INTRODUCTION

The combination of frustration and randomness makes it difficult to find groundstates of spin glasses [1] using numerical simulations. In the past years many methods [2] have been proposed including the multicanonical ensemble [3], genetic algorithms [4], a scheme, which uses storing of spin configurations [5], and an exact algorithm exhibiting exponential timecomplexity [6]. In this paper we present a method which combines clustering of spins with exact calculation of groundstates of polynomial solvable problems in order to calculate approximately groundstates of Edwards–Anderson (EA) spin glasses.
The Hamiltonian of the EA model is given by

$$H = -\sum_{<ij>} J_{ij}\sigma_i\sigma_j - \sum_i B_i\sigma_i \tag{1}$$

where the sum goes over nearest neighbours of spins $\sigma_i = \pm 1$. The exchange interactions J_{ij} are selected at random according a probability distribution. The external field B_i can be site–dependent.
For our computer experiments we used simple cubic lattices of $N = L^d$ spins (L = linear lenght, d = dimension) with periodic boundary conditions in no external field ($B_i = 0$). The interactions are $J_{ij} = \pm 1$ with equal probability using the constraint $\sum_{<ij>} J_{ij} = 0$.

II. ALGORITHM

The algorithm for the approximation of groundstates in frustrated Ising systems works by taking a spin configuration and calculating another one, which has a lower or equal energy. This procedure is iterated many times.

---

*e-mail: hartmann@tphys.uni-heidelberg.de

The idea of the scheme for lowering the energy is to choose a cluster of spins, which exhibits no frustrations. The interaction of the cluster with the spins at its boundary is included into the local fields of the cluster–spins. The groundstate of the cluster is exactly calculated by using concepts of graph theory [7,?,?]: An equivalent network is constructed [10] and the maximum flow is calculated with the Ford–Fulkerson algorithm [11]. Then the network is transformed to a smaller one [12] to speed up the following calculations. Finally a minimum cut is constructed [13]. The new configuration consists of the unchanged spins, which are not in the cluster, and the groundstate of the cluster. By definition this procedure cannot increase the energy.
For describing the spin–cluster the Hamiltonian can be rewritten in the form

$$H_c = -\sum_{<ij>} J_{ij} t_i t_j \sigma_i^c \sigma_j^c - \sum_i B_i^c t_i \sigma_i^c + C \tag{2}$$

The values of $t_i = 0, \pm 1$ describe the cluster and are used to handle antiferromagnetic interactions. If spin σ_i does not belong to the cluster: $t_i = 0$. If $t_i \neq 0$ and $t_j \neq 0$ ($\sigma_i^c t_i := \sigma_i$) their signs have to be choosen so that $J_{ij}^c := J_{ij} t_i t_j > 0$, because only Hamiltonians with positive exchange interactions can be transformed into an equivalent network. The local fields B_i^c include the external field B_i and the interactions with the neighbours of spin σ_i, which are not in the cluster:

$$B_i^c = B_i + \sum_{<j>} J_{ij}(1 - |t_j|)\sigma_j \tag{3}$$

The constant C summarizes the interactions between the non–cluster spins and can be dropped. The construction of the unfrustrated cluster works in the following way: a spin is randomly selected as seed. Iteratively neighbouring spins are added, if no frustration occours. This test is done by looking for a factor t_i which transforms all bonds between the spin and allready existing cluster–spins to positive values. If a cluster cannot be extended, one more seed–spin is selected. The following algorithm contains the details. The local variables δ_i are used to indicate the spins, which are already detected as neighbours of the cluster. The set A contains the neighbours of spin σ_i, which are already in a cluster, whereas B contains the neighbours, which are added to the boundary of the cluster. The boundary is stored on a stack S.

```
algorithm create_cluster({J_ij})
begin
    for all i do
    begin
        initialise t_i ← 0;
        initialise δ_i ← 0;
    end;
    while there are unmarked spins with δ_i = 0 do
    begin
        select one index i_0 from the spins with δ_i = 0;
        push i_0 on an empty Stack S;
        δ_{i_0} ← 1;
        while S is not empty do
        begin
            pop one randomly choosen index i from S;
            A ← {j|j is neighbour of i and t_j ≠ 0};
            if A = ∅ then t_i ← +1;
            else if ∀ j ∈ A : J_ij t_j has the same sign α then
                t_i ← α;
            if t_i ≠ 0 then
            begin
                B ← {j|j is neighbour of i and δ_j = 0};
                for all j ∈ B do
                begin
                    push j on S;
                    δ_j ← 1;
                end;
            end;
            else t_i ← 0;
        end;
```

end;
return $(\{t_i\})$;
end;

A run for calculating a groundstate consists of choosing randomly an initial configuration or choosing all spins pointing up, and calculating new configurations until the energy can not further be lowered. In our experiments we found as a good stopping criterium that the energy did not change for the last n_g steps. We used $n_g = N/2$ (2d,3d) as rule of thumb, because in some very long runs we observed, that the longest period between two jumps in energy never exceeded this value and scaled almost lineary with system size.
Because the minimum energy reached in a run, depends on the starting configuration and on the clusters which were constructed, we performed several runs for each realization of the random variables J_{ij}. We used that one, which ended on the lowest energy level. We found 3 runs per realization sufficent, because by increasing the number of runs, the average groundstate energy was only lowered about 0.05 percent.
Since in each step the spin–cluster is randomly built and by using the algorithm [13] all of the degenerated cluster–groundstates have a positive probability to be calculated, usually in each step a new spin configuration is constructed, even if the energy remains constant. So it is possible to explore large areas of the configuration space.
In figure 1 the average energy density $e_L(t) = E_L(t)/N$ of 96 realizations is shown as a function of the step number for lattice sizes $L = 4, 6, 16$ of the 2d system. One can see that in the beginning the algorithm approaches very fast low values. Later the decrease in energy is very small.
It is possible to use other procedures in selecting the spins of the cluster. We tried three other methods (we call the first presented algorithm *method A*):

- method B Not only neighbours of cluster–spins were added to the stack S, but neighbours of all spins, which were tested, if they can be added to the cluster.
- method C The spins which were tested, if they can be added to the cluster, were totaly selected at random from the spins, which were yet untested.
- method D Methods A and B are applied alternately.

We tested all four methods by calculating groundstates for eight realizations of 16^2 systems (3 runs each). In table I the average groundstate energy density e^0 is displayed. Also we constructed for the 2d and 3d case 100 respectively 10 spin-clusters for 100 realizations of system sizes $L = 4, 6, \ldots, 20$. The average size fractions n_c/N of the spin–clusters are shown in the last two rows.

method	A	B	C	D
$-e^0$	1.425	1.425	1.415	1.427
n_c/N (2d)	0.708	0.696	0.644	–
n_c/N (3d)	0.578	0.568	0.541	–

TABLE I. Comparison of four methods used to construct spin-clusters

We observed that for some realizations method A and for others method B resulted in lower energies, so it is plausible that the combination of both methods gave the best results. So we used method D for all further experiments. Maybe better methods for constructing the cluster would result in a faster convergence and/or lower energies.

III. RESULTS

To estimate the groundstate energies of the 2d and 3d $\pm J$ spin glass we performed calculations for lattice sizes $L = 4, \ldots, 30$ (2d) and $L = 4, \ldots, 14$ (3d) with 96 ($4^2, \ldots, 20^2, 4^3, \ldots, 10^3$), respectively 32 ($30^2, 12^3, 14^3$) realizations of the random variables J_{ij}.
In figure 2 the groundstate energy density $e^0(L)$ of the 2d systems is plotted as function of system size. To estimate the groundstate energy density of the infinite system we performed a finite-size scaling analysis. A fit to the function $f_L^0(N) = f_\infty^0 + c/N$ [3] results in $e^0 = -1.400 \pm 0.005$. This value is consistent with earlier results from a genetic algorithm -1.400 ± 0.005 [4], multicanonical simulations -1.394 ± 0.007 [3] pure MC -1.407 ± 0.008 [14] and transfer matrix calculations -1.4024 ± 0.0012 [15].
The results for the 3d systems are shown in figure 3. The fit gives a groundstate energy density of $e^0 = -1.766 \pm 0.002$. The results of other authors are 1.7863 ± 0.0028 [3] and 1.765 ± 0.01 [4]
As seen in figure 1 the time for approaching the groundstate increases rapidly with the system size. The average number of steps needed to calculate a groundstate are shown in figure 4 as a function of the number of spins N. For the 2d system (lower curve) a fit to a function $t(N) = a + bN^c$ results in a timecomplexity of $O(N^{1.17 \pm 0.06})$. The result for the 3d system is similiar: $O(N^{1.27 \pm 0.08})$. Because the time to calculate one groundstate of the cluster increases quadraticly with the number of spins [13] and the size of the cluster is linear in the system size, this totally results in an approximate cubic timecomplexity of the algorithm. Performing one run for example for a 10^3 system consisting of 1000 configurations needed about 1.5 hours on a 80 Mhz PowerPC processor system.
In order to characterize the way the algorithm approaches the groundstates we selected one realization of a 2d ($L = 16$) system and used configurations of different energy densities $e_s \in [-1.414, 1.0]$ as starting–configurations. For each configuration 8 different runs for constructing groundstates were performed. Only for the runs, which exhibited the groundstate energy density value of $e^0 = -1.414$, the last 100 configurations were stored. The overlap

$$q := \sum_{i=1}^{N} \sigma_i^\alpha \sigma_i^\beta \tag{4}$$

was calculated between all pairs (α, β) of groundstates belonging to a starting-configuration . We obtained the corresponding probability distributions $P_L(q)$ by counting the numbers of overlaps within intervals of length $\Delta q = 0.05$. The result is displayed in figure 5 for the starting energies $e_s = -1.414, -1.07, 0.117$. For these three staring energies 800, 300 respectively 400 of the resulting states exhibited the lowest energy density and were included into the calculation. The distribution is not symmetric, because the different groundstates are not independent. One can see that the algorithm approaches local minima of the energy landscape. The higher the starting energy, the broader is the distribution, i.e. the more minima are reachable. If a system is caught in a local minimum, only a restricted area of configuration space can be explored by the algorithm, which is indicated by a narrow distribution for low starting energies. But these minima are very rich of states, because in each step a different configuration is generated which is indicated by $P_L(1) = 0$. One should mention, that for other realizations we got also different looking distributions. There are systems where $P_L(q)$ is always narrow or always broad, independent of the starting energy. The accessible area is usually smaller for lower groundstate energies than for higher lying groundstates.
The distribution function displayed in figure 5 is not symmetric, because the different groundstates are not generated independly. In order to characterize not the algorithm but the groundstate structure, we made for each of 20 realizations of 16^2 systems 80 independent runs for calculating groundstates, dropped the states which exhibited not the lowest energy and computed once more the overlaps. We weighted the overlap values from different systems with factors to scale out the incfluence from the fact that the number of computed groundstates was dependent on the system. The result is shown in figure 6. The distribution covers the whole range $[-1, 1]$ which indicates the complex nature of the spin glass groundstates. For the same procedure in 3d we used 800 independent runs to get enough groundstates, because many of the states we got, lied little obove (mostly 4J/N difference) the lowest states. Here the overlap distributions are much more peaked around the regimes $[-1.0, -0.6]$ and $[0.6, 1.0]$.

IV. DISCUSSION

In conclusion, in this paper we have presented an algorithm for the approximation of groundstates for Ising spin glasses. Its time complexity is approximately cubic in the number of spins. Because at each step a large number of spins is allowed to flip, the algorithm approaches low energies very rapidly.
So it should be interesting to try combinations with other algorithms like genetic algorithms or Monte Carlo simulations, or to use it for other Ising type optimization problems.
In comparison with an earlier cluster algorithm [16] which approximated groundstates of 2d $\pm J$ spin glasses without externel field using clusters surrounded by frustrated plaquettes [17], this algorithm works in any dimension, with arbitray forms of interactions/lattices and allows external fields.
It should possible to get results for $B \neq 0$ in a resonable amount of time in order to characterize better the 3d phase diagramm along the $T = 0$ axis.

ACKNOWLEDGMENTS

We would like to thank D. W. Heermann, K. D. Usadel and G. Reinelt for fruitfull discussions, A. Linke for a critical reading of the manuscript and the Paderborn Center of Parallel Computing for the allocation of computer time on a PARIX–PowerPC System.

---

[1] For reviews see
K. Binder and A. P. Young, Rev. Mod. Phys. **58**, 801–976 (1986);
K. H. Fisher and J. A. Hertz, Spin Glasses, Cambridge University Press (1991)
[2] H. Rieger, in Annual Reviews of Computational Physics **II**, ed. D. Stauffer, World Scientific, Singapore (1995)
[3] B. A. Berg and T. Celik , Phys. Rev. Lett. **69**, 2292–2295 (1992);
B. A. Berg, T. Celik and U. Hansmann, Europhys. Lett. **22**, 63–68 (1993);
U. H. E. Hansmann and B. A. Berg, Int. J. Mod. Phys. C **5**, 263–265 (1994)
[4] P. Sutton, D. L. Hunter and N. Jan, J. Physique I **4**, 1281–1285 (1994)
[5] W. Krauth and O. Pluchery, J. Phys. A **27**, L715–L720 (1994)
[6] F. Barahona, Phys. Rev. B **49**, 12864–12867 (1994)
[7] J. D. Claiborne, Mathematical preliminaries for computer networking, John Wiley & Sons, New York 1990
[8] W. Knödel, Graphentheoretische Methoden und ihre Anwendung, Springer, Berlin 1969
[9] R. E. Tarjan, Data Structures and Network Algorithms, Society for Industrial and Applied Mathematics, Philadelphia, 1983
[10] J.–C. Picard and H. D. Ratliff, Networks **5**, 357–370 (1975)
[11] L. R. Ford and D. R. Fulkerson, Canadian J. Math. **8**, 399–404 (1956)
[12] J.–C. Picard and M. Queyranne, Math. Prog. Study **13**, 8–16 (1980)
[13] A. K. Hartmann and K. D. Usadel, Physica A **214**, 141–152 (1995)
[14] J.–S. Wang and R. H. Swendsen, Phys. Rev. B **38**, 4840–4844 (1988)
[15] H.–F. Cheung and W. L. McMillan, J. Phys C **16**, 7027–7032 (1983)
[16] H. Freund and P. Grassberger, J. Phys. A **22**, 4045–4059 (1989)
[17] I. Morgenstern and H. Horner, Phys. Rev. B **25**, 504–507 (1982)

FIGURE CAPTIONS

Figure 1 Average energy density $e_L(t) = E_L(t)/N$ of 2d $\pm J$ spin glass at different iterations t. Shown are 3 sizes $L = 4, 6, 16$.

Figure 2 Average groundstate energy density $e^0(L) = E^0(L)/N$ of 2d $\pm J$ spin glass as a function of size L.

Figure 3 Average groundstate energy density $e^0(L) = E^0(L)/N$ of 3d $\pm J$ spin glass as a function of size L.

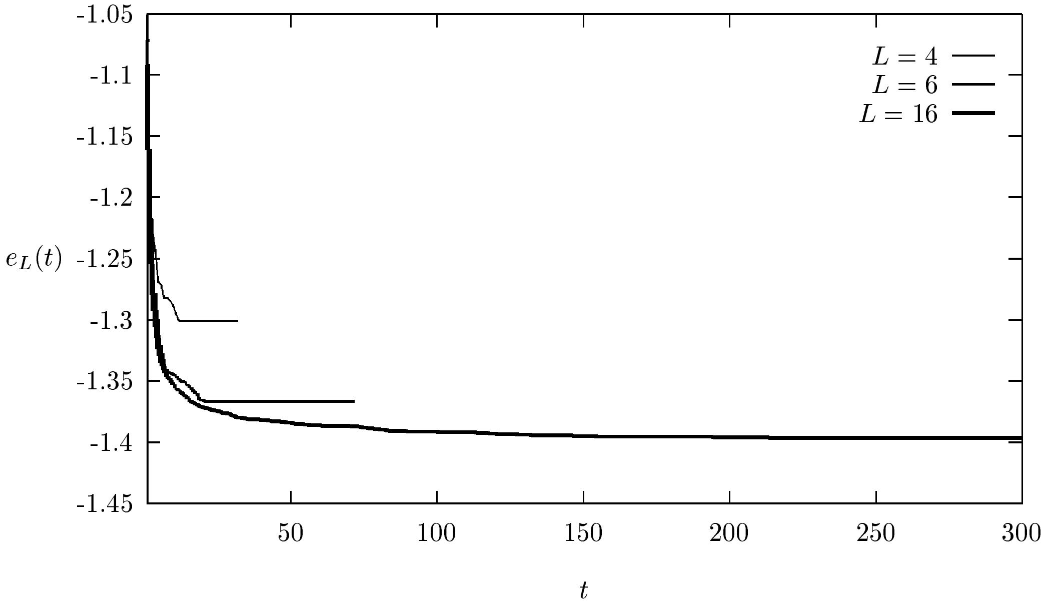
Figure 4 Average number of steps needed to reach a groundstate as function of system size N for 2d system ($L = 4, \ldots, 20$, lowerline) and 3d system ($L = 4, \ldots, 14$, upper line).

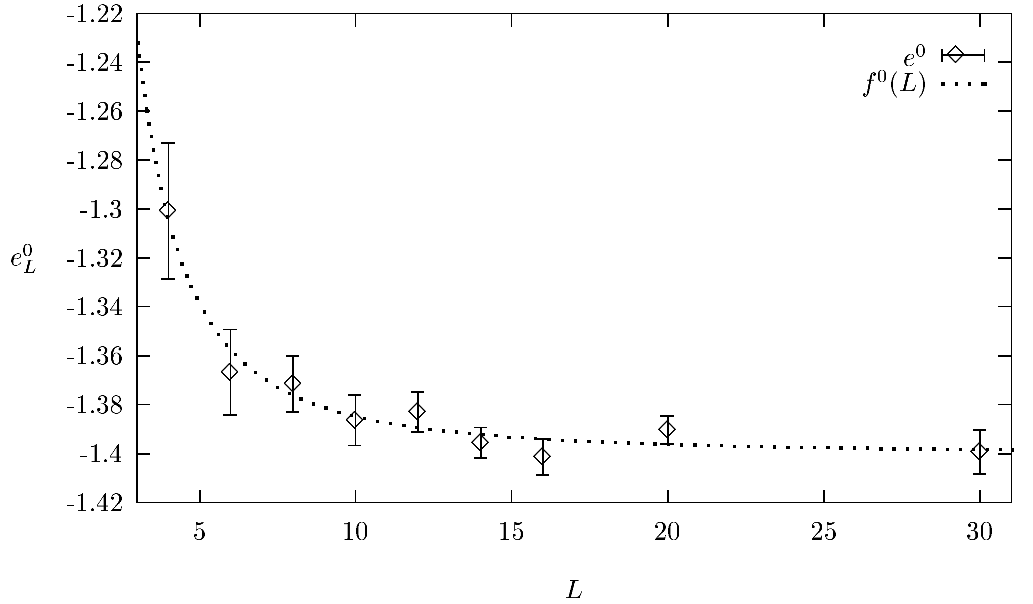
Figure 5 Distribution of overlaps for one single realization (2d, $L = 16$, $e^0 = -1.414$). For different starting configurations with energies $e_s = 0.117$, -1.07, -1.414 in each of 8 independent runs 100 groundstates were calculated. The states with energy density $e^0 = -1.414$ were included in the calculation of the distribution, i.e. 400, 300, 800 states for the three values of e_s. The lines are guides for the eyes only.

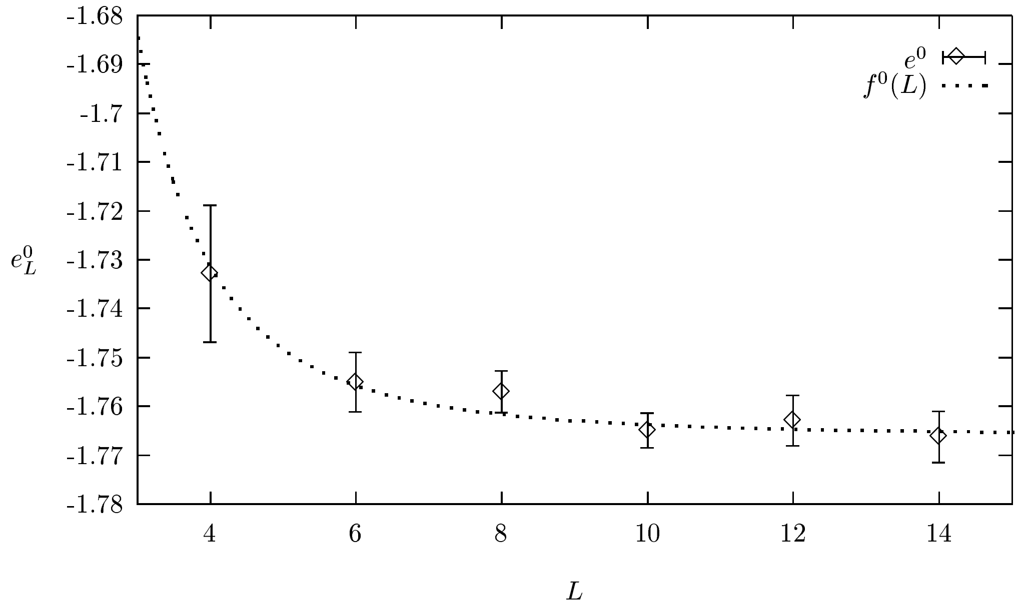
Figure 6 Distribution of overlaps averaged over 20 realizations (2d, $L = 16$) and 80 independent runs. The lines are guides for the eyes only.

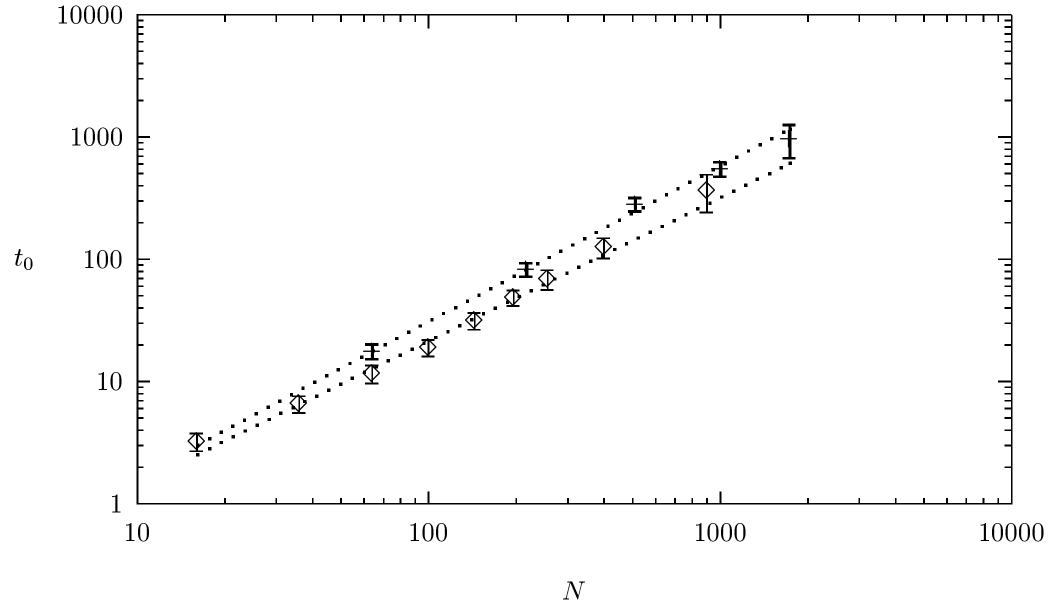
Figure 7 Distribution of overlaps averaged over 20 realizations (3d, $L = 8$) and 800 independent runs. The lines are guides for the eyes only.

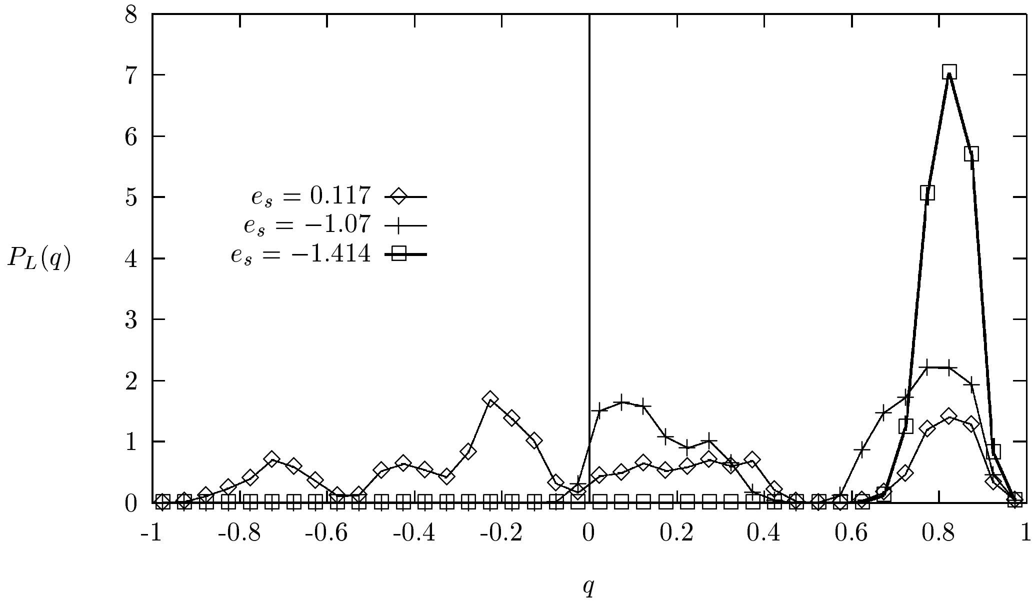
FIGURES



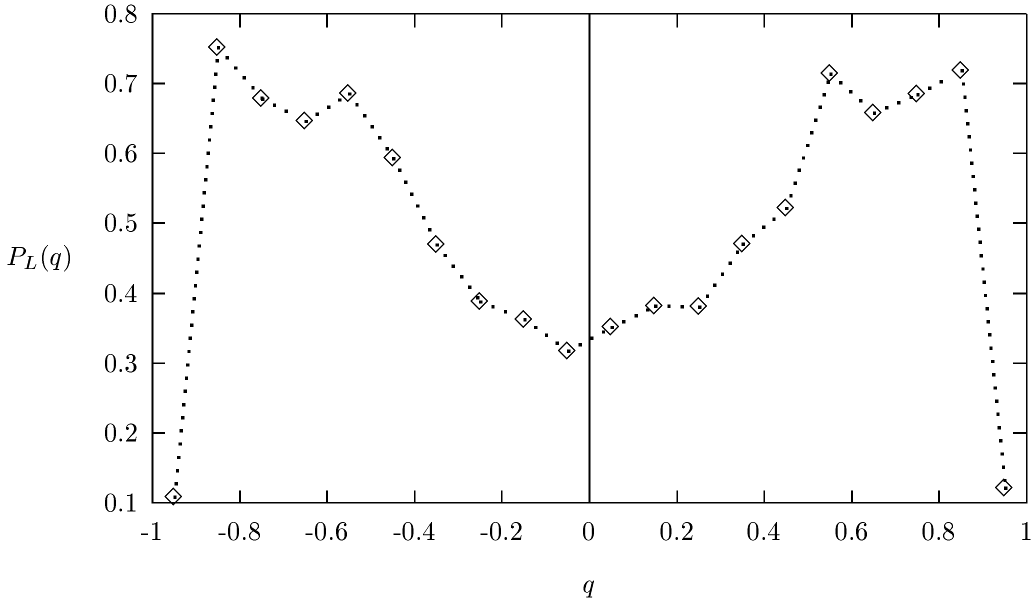



-1.68
-1.69
-1.7
-1.71
-1.72
-1.73
-1.74
-1.75
-1.76
-1.77
-1.78
e_L^0
4
6
8
10
12
14
L
e^0
$f^0(L)$

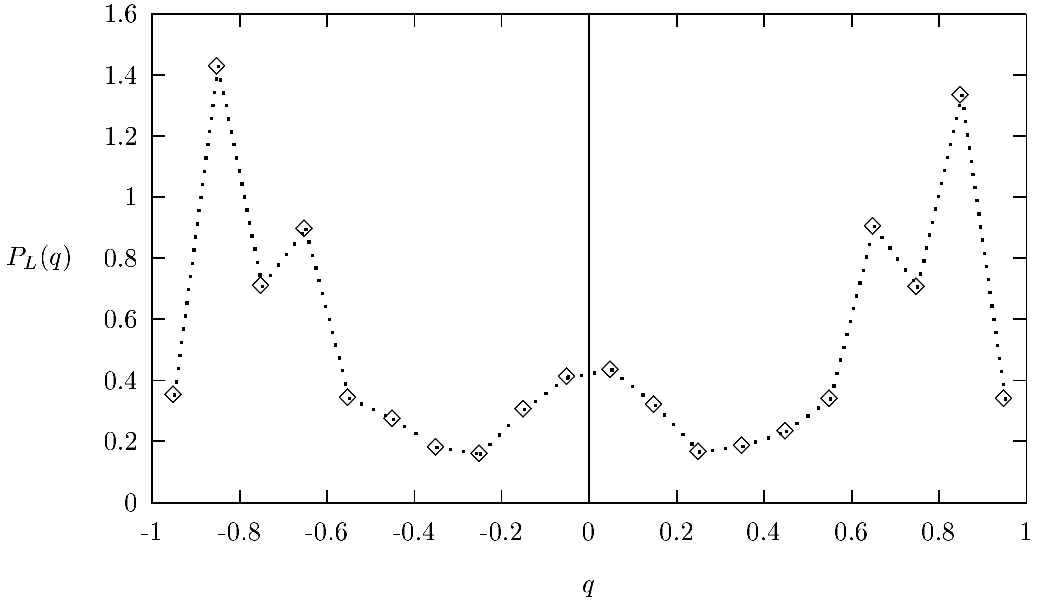

10000
1000
100
10
1
t_0
10
100
1000
10000
N



8
7
6
5
4
3
2
1
0
$P_L(q)$
$e_s = 0.117$
$e_s = -1.07$
$e_s = -1.414$
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
q



0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
$P_L(q)$
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
q


1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
$P_L(q)$
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
q